8-11-2015

# Data Aggregation Scheduling in Wireless Networks

Mingyuan Yan

DATA AGGREGATION SCHEDULING IN WIRELESS NETWORKS

by

MINGYUAN YAN

Under the Direction of Yingshu Li, PhD and Zhipeng Cai, PhD

## ABSTRACT

Data aggregation is one of the most essential data gathering operations in wireless networks. It is an efficient strategy to alleviate energy consumption and reduce medium access contention. In this dissertation, the data aggregation scheduling problem in different wireless networks is investigated. Since Wireless Sensor Networks (WSNs) are one of the most important types of wireless networks and data aggregation plays a vital role in WSNs, the minimum latency data aggregation scheduling problem for multi-regional queries in WSNs is first studied. A scheduling algorithm is proposed with comprehensive theoretical and simulation analysis regarding time efficiency. Second, with the increasing popularity of Cognitive Radio Networks (CRNs), data aggregation scheduling in CRNs is studied. Considering the precious spectrum opportunity in CRNs, a routing hierarchy, which allows a secondary user to seek a transmission opportunity among a group of receivers, is introduced. Several scheduling algorithms are proposed for both the Unit Disk Graph (UDG) interference model and the Physical Interference Model (PhIM), followed by performance evaluation through simulations. Third, the data aggregation scheduling problem in wireless networks with cognitive radio capability is investigated. Under the defined network model, besides a default working spectrum, users can access extra available spectrum through a cognitive radio. The problem is formalized as an Integer Linear Programming (ILP) problem and solved through an optimization method in the beginning. The simulation results show that the ILP based method has a good performance. Although, it is difficult to evaluate the solution theoretically. A heuristic scheduling algorithm with guaranteed latency bound is presented in our further investigation. Finally, we investigate how to make use of cognitive radio capability to accelerate data aggregation in probabilistic wireless networks with lossy links. Lossy links introduce retransmissions that occurs more energy consumption and transmission delay. A two-phase scheduling algorithm is proposed, and the effectiveness of the algorithm is verified through both theoretical analysis and numerical simulations.

INDEX WORDS: Data aggregation, Scheduling, Connected dominate set, Delay analysis, Wireless sensor network, Cognitive radio network, Probabilistic wireless networks

DATA AGGREGATION SCHEDULING IN WIRELESS NETWORKS

by

Mingyuan Yan

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2015

DATA AGGREGATION SCHEDULING IN WIRELESS NETWORKS

by

MINGYUAN YAN

| | |
|---|---|
| Committee Chair: | Yingshu Li |
| | Zhipeng Cai |
| Committee: | Raj Sunderraman |
| | Anu Bourgeois |
| | Xin Qi |

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

August 2015

# DEDICATION

This dissertation is dedicated to my family.

## ACKNOWLEDGEMENTS

I would never have been able to finish my dissertation without the guidance of my committee members, help from my group, and support from my family and my friends. I would like to show my great gratitude to all of them.

I would like to express my deepest gratitude to my advisors, Dr. Yingshu Li and Dr. Zhipeng Cai, for their excellent inspiration, guidance, caring, patience, and supporting me with a great atmosphere for doing research. Dr. Li and Dr. Cai always gave me the greatest support during my graduate studies, they patiently inspired me and financially supported my research.

I would like to thank my committee members, Dr. Raj Sunderraman, Dr. Anu Bourgeois, and Dr. Xin Qi. Dr. Sunderraman and Dr. Bourgeois gave me great support for my PHD study. They also kindly provided me valuable teaching suggestions. Special thanks goes to Dr. Xin Qi, who was willing to participate in my final defense committee at the last moment.

I also would like to thank the professors and staffs at our department, especially Ms. Tammie Dudley, Ms. Adrienne Martin and MS. Venette Rice, their patient help makes my life much more easier and colorful.

I would like to thank my group members and fellow students. Special thanks goes to Dr. Shouling Ji, Dr. Jing He, Guoliang Liu and Meng Han, who provided me help and happiness besides on research.

Last but not least, I would like to thank my family and my friends for their continuing support, understanding, and love.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

- WSNs - Wireless Sensor Networks
- FCC - Federal Communications Commission
- CRNs - Cognitive Radio Networks
- PUs - Primary Users
- SUs - Secondary Users
- DSA - Dynamic Spectrum Access
- UDG - Unit Disk Graph
- CSMA - Carrier Sense Multiple Access
- QoS - Quality of Service
- MRQ - Multi-Regional Query
- RN - Regional Node
- RQT - Regional Query Tree
- MRQF - Multi-Regional Query Forest
- BFS - Breadth-First-Search
- DS - Dominating Set
- CDS - Connected Dominating Set
- SINR - Signal-to-Interference-plus-Noise Ratio
- PhIM - Physical Interference Model

- MIS - Maximal Independent Set
- PCR - Proper Carrier sensing Range
- ILP - Integer Linear Programming
- LP - Linear Programming
- AN - Auxiliary Network
- AU - Auxiliary User
- AUs - Auxiliary Users
- BRT - Balanced Routing Tree
- DNM - Deterministic Network Model
- PNM - Probabilistic Network Model
- PRWN - Probabilistic Regular Wireless Networks
- MLDAS - Minimum Latency Data Aggregation Scheduling

# Chapter 1

# INTRODUCTION

## 1.1 Background and Motivations

Wireless networks are networks in which users communicate with each other through wireless connections. The wireless property considerably reduces the network installation expense by avoiding the costly process of introducing cables and wires. It enables users to achieve equipment portability and location independence. It also makes network expansion simpler. Due to its flexibility, cost-savings and scalability, wireless networks play an vital role in people's life and are widely used in different areas [1][2][3][4][5][6][7][8].

Wireless Sensor Networks (WSNs) are one of the most important types of wireless networks. A WSN consists of multiple wireless sensor nodes. Sensor nodes are usually small in size, consume extremely low energy, and can operate in high volumetric densities. When sensor nodes are deployed in a monitoring area, they can adapt to the environment, self-organize into a network, and operate unattended [9]. Benefiting from all those advantages, WSNs have a variety of applications, *e.g.* environmental monitoring, battlefield surveillance, traffic control, and so on [10].

Data aggregation has been considered an essential data gathering operation in wireless networks. During the data aggregation process, raw readings are aggregated and then transferred within the network. Classic aggregation functions such as *sum*, *maximum*, *minimum*, *average*, and *count* are widely used. Since data is aggregated at intermediate nodes during the transmission process, both data redundancy and the number of transmissions are reduced. Therefore, data aggregation is an efficient strategy to alleviate energy consumption and medium access contention. One major task of WSNs is to collect sensed data from sensor nodes deployed in the monitoring area. Data aggregation plays a vital role in the collecting of summarized information in WSNs, such as tracking critical phenomena in continuous or periodic monitoring applications. Therefore, the minimum latency data aggregation scheduling problem for multi-regional queries in WSNs is first studied.

Second, with the increasing popularity of Cognitive Radio Networks (CRNs), data aggregation scheduling in CRNs is investigated. Wireless networks have been employed in a variety of applications. They provide an economical and convenient way to share wireless bandwidth among multiple users. The explosive growth of population in wireless networks during the last few decades indicates the urgency of improving the inefficient spectrum usage. However, the traditional fixed spectrum assignment policy for wireless networks allows no one but licensed Primary Users (PUs) or services to access the spectrums. Some records show that only certain portions of the spectrums are efficiently utilized whereas the rest remain unutilized most of the time under this policy. Particularly, a report from Federal Communications Commission (FCC) demonstrates that the usage of spectrum assigned to PUs is between 15% and 85%. In order to alleviate the spectrum shortage and underutilization problem, the cognitive radio technology has been introduced and investigated recently. In Cognitive Radio Networks (CRNs), unlicensed Secondary Users (SUs) coexist with PUs. As long as communications among PUs are ensured [11], SUs are able to access and exploit the unoccupied licensed spectrum using cognitive radio which is capable of sensing/accessing available spectrums and switching between spectrums in an opportunistic manner. Due to the precious spectrum opportunity in CRNs, it is meaningful to study data aggregation which reduces redundant information and only focuses on the most valuable aggregated information in network.

In CRNs, SUs can only access spectrum opportunistically. This reality constraints the usage of CRNs, especially in applications with heavy traffic, even though cognitive radio capability is a wonderful technology. It would be a pity if we could not make a full use of it. From this point of view, researchers proposed to introduce cognitive radio capability to conventional wireless networks [12][13][14][15]. Motivated from those works, the data aggregation scheduling problem in wireless networks with cognitive radio capability is investigated. Under the defined network model, besides a default working spectrum, users can access extra available spectrum through a cognitive radio. Scheduling concurrency can be improved via extra transmission opportunity on the extra spectrum. Default working spectrum can be used if no extra spectrum is available or interference on the extra spectrum is too high. Our motivation is to accelerate the data aggregation process by seeking transmission opportunity on an extra spectrum instead of relying on it.

Finally, we investigate how to make use of cognitive radio capability to accelerate data aggregation in probabilistic wireless networks with lossy links. A large number of Researchers verified the existence of *Transitional Region Phenomenon*[16][17], which leads to the existence of lossy links in wireless networks and results in the fact that a transmission between two users who are theoretically connected under the *Deterministic Network Model* cannot be guaranteed. Therefore, we focus on a more practical network model called Probabilistic Network Model (PNM) which can better characterize the lossy links [18] in wireless networks. When lossy links are considered, retransmission may be needed to guarantee that receivers can receive a data package successfully and accurately. This requires more time and energy consumption. By involving cognitive radio technology, users in the wireless networks can seek extra transmission opportunity if other spectrum resource is available. Otherwise, the data aggregation process still can be done on the default working spectrum.

Based on the above mentioned motivations, in this dissertation, we focus on the data aggregation scheduling problem in WSNs, CRNs and wireless networks with cognitive radio capability under both deterministic and probabilistic network models.

## 1.2 Characteristics of WSNs

A WSN consists of wireless sensor nodes. A sensor node usually has a power unit, sensing unit, processing unit, storage, communication unit, and software. These components enable a sensor node to do computing, communication, and store data as a small smart computer. However, since sensor nodes are small in size and limited in power supply, the capability of all those functions are restricted. The important characteristics of WSNs can be summarized as follows.

1. **Large-Scale.** Since sensor nodes are small in size and cheap in price, WSNs are suitable for large scale wireless networks. To deal with large scale WSNs, robustness and scalability will be two important features that researchers must consider.

2. **Constrained Power Supply.** Sensor nodes are usually powered by batteries. Their battery are very limited in order to keep the size small. Hence, how to use the limited precious energy in an efficient way and how to operate the sensor nodes in a more energy saving manner is one of the most important concerns in WSNs.

3. **Restricted Communication Capability.** Based on the limited energy supply, the communication capability of sensor nodes are also limited.

4. **Limited Computing and Storage Capabilities.** As a tiny computer, the computing ability and storage capability of sensor nodes are constrained.

5. **Dynamic Network.** Because WSNs are usually deployed in hostile environment and short in energy, some of them may die due to damage or power depletion and result in topology change.

6. **Redundancy.** In many applications, sensor nodes are deployed in a dense arrangement. Due to the dense deployment, data generated by adjacent sensor nodes may be similar, which introduces redundancy into WSNs. Therefore, redundancy is another important issue that need to be considered in WSNs.

Overall, all the characteristics of WSNs concluded above introduce challenges into WSNs, and make solutions for traditional wireless networks unsuitable for WSNs.

## 1.3 Characteristics of CRNs

CRNs (also known by NeXt Generation (xG) Networks or Dynamic Spectrum Access (DSA)) are motivated by the reality of the limited available spectrum and the inefficient spectrum usage. CRNs have been considered as a promising solution to relieve the disequilibrium. The core technology of CRNs is the cognitive radio. Cognitive radio has the capability of detecting and opportunistically using or sharing spectrums. To be specific, a cognitive radio has cognitive capability and Reconfigurability.

- Cognitive capability. The cognitive capability enables a user to obtain or sense the spectrum information from its radio environment. It contains three processes: spectrum sensing, spectrum analysis and spectrum decision. During the spectrum sensing period, the cognitive radio senses the available spectrum bands, and then detects spectrum holes that are not used by licensed PUs based on the captured information. The spectrum holes and captured network information are analyzed in the spectrum analysis period, and a best decision for spectrum access is made based on the analysis.

- Reconfigurability. The reconfigurability allows the radio to dynamically adjust its operating transmitter parameters without modifying hardware components. Particularly, it can adjust transmitter parameters according to its interaction with the spectrum environment. These parameters include operating frequency, modulation, transmission power and communication technology.

To summary, the cognitive radio technology is capable of distinguishing which portions of the spectrum are available, making the best decision to select the best available spectrum, employing a targeting spectrum hole coordinately with other users, and releasing the spectrum timely when a licensed PU is detected [11].

## 1.4 Characteristics of Data Aggregation

Data aggregation is a process during which information is searched, collected and presented in a summarized format, to achieve the purposes of statistical analysis, reducing network traffic and so on [19]. It aggregates the input information, condenses it into smaller output. It does a great job in processing massive amounts of data in different applications. Data aggregation in wireless networks is mainly used for the following two purposes.

- Reduce network traffic. In wireless networks, bandwidth is a precious network resource, especially with the explosive increase in the number of users. Data aggregation is an efficient way to reduce network traffic by summarizing information and transferring the aggregated information instead of the massive raw data.

- Reduce redundancy. In applications such as monitoring a target area in WSNs, data collected by adjacent users may be quiet similar, which challenges the robustness of communications in WSNs by redundancy. Data aggregation is a useful solution for reducing redundancy by aggregating related raw information into more condense information.

Data aggregation is one of the most important functions in wireless networks due to the above mentioned advantages.

## 1.5 Deterministic and Probabilistic Network Models

In this dissertation, we follow the presentation of [20] and [21]. The ideal Deterministic Network Model (DNM) models a wireless network in which users are either considered as connected or disconnected. For two users that are considered as connected, a deterministic and reliable link is assumed to exist between them, so that data transmission is guaranteed to be successful as long as there is no collision or interference from the environment. The DNM has been extensively employed in existing literatures because of its suitability in deriving theoretical analysis and developing optimization algorithms. Contrary to DNM, another network model named Probabilistic Network Model (PNM) which can characterize the wireless links in a better and practical way has been proposed. In PNM, wireless links are not as reliable as that in DNM, so called lossy links. Due to the existence of lossy links, even without collision or interference from the environment, transmission between users can be successfully conducted with a certain probability instead of fully guaranteed.

In the following chapters, research in chapter 3, 4 and 5 are conducted under the DNM, while chapter 6 is based on the PNM.

## 1.6 Organization

The rest of this dissertation proposal is organized as follows: Chapter 2 summarized the related literature. Chapter 3 studies data aggregation scheduling for Multi-Regional Query in WSNs with Minimum Latency. Chapter 4 investigates the data aggregation scheduling problem in CRNs, where the target is to minimum the overall transmission delay. Chapter 5 investigates the data aggregation problem when cognitive capability is introduced to wireless networks. Chapter 6 focuses on the minimum delay data aggregation scheduling in probabilistic wireless networks with cognitive radio capability. Chapter 7 concludes this dissertation.

# Chapter 2

# RELATED WORK

Although data aggregation scheduling has been widely studied in WSNs, less attention has been paid on it in CRNs. It was hard to find existing works of data aggregation scheduling in wireless networks with cognitive capability. In this section, the research progress and related literature are summarized.

## 2.1 Data Aggregation in WSNs

Considered one of the most important processes in WSNs, data aggregation scheduling has been extensively investigated from many viewpoints, such as time efficiency, energy efficiency, security and others. How to develop energy efficient data aggregation protocols to extend the network lifetime has been discussed in [22][23][24][25][26][27][28][29]. Secure and privacy preserving data aggregation are investigated in [30][31][32][33][34]. In this section, literature working on time efficient data aggregation in WSNs is discussed in detail.

The authors of [35] investigated the time efficient data aggregation scheduling problem in wireless sensor networks under the Unit Disk Graph (UDG) interference model. A scheduling algorithm based on the maximal independent set has been proposed in this paper. Theoretical analysis indicates that the latency bound of the proposed algorithm is $23R + \Delta - 18$, where $R$ is the radius of the network and $\Delta$ is the maximum degree. This paper has been cited many times because $\Delta$ contributes as an additive factor instead of a multiplicative factor as in previous literature. In [36], the authors dedicated their efforts on the aggregation scheduling problem in WSNs when multiple frequency channels are available. The NP hardness of minimizing the scheduling latency in an arbitrary network with respect to multiple channels has been proved. The authors then demonstrated that finding the minimum number of channels required in the network to alleviate all the interference is

NP-hard. To solve the problem, an approximation algorithm with constant factor is introduced for unit disk graphs, and a $\Delta * \log n$ approximation algorithm is proposed for general graphs. The work in [37] formulates the scheduling problem of minimizing the overall data transmission delay. The characteristics of optimal scheduling were studied, followed by the proof of the lower bound of optimal performance. Two scheduling policies were proposed in this paper. Decision in one policy was made based on the current system state. In the other policy, predication of the future system conditions were also taken into consideration. The authors of [38] studied the distributed aggregation scheduling problem in WSNs with respect to minimum latency. Compared with centralized solutions, a distributed scheduling plan has its own advantages. In this paper, an algorithm based on vertex coloring was proposed with a proved delay of $4R + 2\Delta - 2$. In [39], minimum latency aggregation scheduling in WSNs with multiple sinks were investigated. Differentiating from prior literatures which considers only one fixed sink collecting data aggregation result, sensor nodes in the network make their best choice of which sink they should send the data to for the purpose of minimizing the transmission latency. Two approximation algorithms with bounded latency were proposed. The data aggregation problem seeks a schedule with minimum latency in multihop wireless networks subject to different interference constraint was studied in [40]. [40] concentrates on the protocol interference model with both $\rho = 1$ and $\rho \geq 1$, where $\rho$ is the ratio of the communication radius and the interference radius. Three scheduling algorithms with guaranteed latency were introduced for $\rho = 1$, and two scheduling algorithms are presented from the extension of the case when $\rho = 1$.

### 2.2 Data Aggregation in CRNs

Cognitive radio networks (CRNs) have been considered as the next generation communication networks, which can provide high bandwidth to mobile users via heterogeneous wireless architectures and dynamic spectrum access techniques [11]. A cognitive radio can adjust its transmitter parameters based on interchanging information with its operating environment. This capability enables wireless devices with cognitive radio to dynamically access networks. There are hundreds of existing literature working on topics in cognitive radio networks regarding *spectrum sensing*, *spectrum management*, *spectrum mobility* and *spectrum sharing*. In most of the existing research, cognitive radio networks have no spectrum resources that can opportunistically access licensed spectrum in the environment. [11] and [41] summarized the characteristics and challenges in cognitive radio networks, which provided a foundation for most of the following research works. The authors of [42] and [43] investigated the spectrum sensing issue in cognitive radio networks. [42] surveys the methodologies, challenges of spectrum sensing and enabling spectrum sensing methods. It also introduces the statistical modeling of network traffic and utilization of these models for prediction of primary user behavior. To overcome the signal fading, shadowing and receiver uncertainty issues, cooperative spectrum sensing has been demonstrated as an effective method to improve the detection performance by exploiting spatial diversity . [43] concludes the state-of-the-art cooperative sensing methods and open research challenges addressing the issues of cooperation method, cooperative gain, and cooperation overhead. The authors of [44], [45] and [46] surveyed the advances and intelligence in research of recent cognitive radio networks, including the fundamentals of cognitive radio technology, architecture of a cognitive radio network and its applications and important issues in dynamic spectrum allocation and sharing.

Many applications in cognitive radio networks rely on control message exchanges on a common control channel. In [47], the authors summarized the challenges of designing a common control channel, such as the robustness of the mechanism with respect to primary users' activity, limited coverage and so on. Features and design issues of spectrum mobility were surveyed in [48]. [49] and [50] investigate the spectrum sensing and spectrum handoff problem, respectively. In [49], the authors focused on finding the optimal spectrum probing strategy so that SUs can detect PUs' state change effectively and efficiently. In [50], the modeling and performance analysis for the proactive-decision spectrum scheme was discussed. The works in [51][52][53] focus on maximizing the network performance through cross-layer optimizations. The issues of power control and topology control are investigated in [54][55][56].

However, little attention has been paid to data aggregation. The most related literature was presented by [57], which studied the data collection scheduling problem in CRNs.

## 2.3 Wireless Networks with Lossy Links

[16] and [17] verifies the existence of *Transitional Region Phenomenon*. Test on wireless devices shows that the transitional region phenomenon in wireless networks leads to the problem that a large number of links in the network are not reliable. Those unreliable links are called lossy links. The authors of [58] focused on developing wireless network generators that could generate a wireless network instance with lossy links of an arbitrary size, under an arbitrary deployment rule, and considering all important features which might affect the actual deployment. [59] investigates the problem of how lossy links impact the performance of multihop wireless networks. Their attention were paid on the maximum achievable throughput and energy utilization. Optimization methods were used to find optimal solutions for both flow-based and destination-based routing. The authors of [60] studied the existence of uniqueness of fair rate allocation in wireless networks with lossy links, in which the max-min optimization function was used to deal with the fairness. In [18], the authors modeled the lossy links as leaky pipes. They described data flows transmitted through lossy links as water flowing through a leaky pipe, thus, with data rate of a given flow reducing along the link. Cross layer rate control was investigated as a method to maximize the network utility. Authors of [61] were interested in developing efficient geographic routing protocol over lossy links in wireless sensor networks. [62] and [63] studied data aggregation in probabilistic wireless sensor networks, where [62] emphasized energy efficiency and [63] focused on continuous data aggregation and network capacity. The authors in [20] studied the construction of a load-balance tree for data aggregation in probabilistic wireless sensor networks. A Connected Dominating Set (CDS)- based tree has beed proposed through approximation algorithm and max-min optimization function is used to balance the load.

## 2.4 Wireless Networks with Cognitive Capability

Different from conventional CRNs in which a SU can only opportunistically access the licensed spectrum unoccupied by PUs, literatures summarized in this subsection discuss different wireless networks with cognitive radio capability. In all literature, users worked in a mixed spectrum environment of licensed and unlicensed spectrums, and a default unlicensed spectrum is always available with the licensed ones dependant on activities of PUs. The authors of [12] proposed to improve the secondary network performance by allowing SUs operate in the heterogeneous spectrum environment. An analytical model was provided for the Carrier Sense Multiple Access (CSMA) in an unslotted Ad hoc network. Even though two types of spectrums exist, PUs will have absolute priority on the licensed channel. In order to avoid collision, blocked SUs and preempted SUs were forced to leave the system forever if all spectrums are busy, which was questioned as unreasonable in [13]. In [13], the authors focused on the same model as [12] with an adjustment that the blocked or preempted SUs retry with a probability $q$ or leave the system with probability $1-q$. Cognitive technique was used in smart grid by [14]. In this paper, the authors concentrated on obtaining the best sensing time to balance the sensing accuracy and delay. However, the introduced analysis model only applied to single-hop data transmission in smart grid network. The authors of [15] studied the benefits that could be gained by introducing cognitive capability into wireless mesh network. The studied problem was formulated into integer linear programming, after that, the improvement on Qos and resource utilization were analyzed. [64] and [65] as well as literatures surveyed in [64][65] investigated cognitive wireless sensor networks. Challenges, frameworks and protocols have been studied under the condition that users in the wireless sensor networks have cognitive capability. [66][67] focus on the spectrum access and topology control in a cognitive Ad hoc wireless networks. [68] analyzed the issues and challenges of cognitive Ad hoc wireless networks from a more comprehensive level based on existing works on this topic.

# Chapter 3

# MINIMUM LATENCY DATA AGGREGATION SCHEDULING FOR MULTI-REGIONAL QUERY IN WIRELESS SENSOR NETWORKS

## 3.1 Introduction

WSNs have variety of applications due to its low cost and easy deployment [10]. One major task of WSNs is to collect sensed data from sensor nodes deployed in the monitoring area. WSNs are also called *data-centric* networks where the sensed data of all the sensor nodes can be viewed as a database. Due to the fact that sensor nodes are usually randomly deployed in hostile or unreachable areas, in most cases, the only way that users can interact with WSNs and obtain useful information is to disseminate queries from a special node called the sink (or base station). Therefore, query processing is one of the most important issues in WSNs and has been widely studied[69][70][71][72]. Early researches of query processing mainly focuses on how to improve query efficiency [73][74] while minimizing the energy consumption. With the emergence of *real-time query processing* [75], *e.g.* fire monitoring in forests, *minimum latency* becomes another primary concern. Hence, *minimum latency query scheduling*, which aims to derive a collision-free query scheduling plan with minimum latency, shows its advantage in time-sensitive applications recently.

Ever since the query scheduling problem was introduced, tons of efforts have been made to design different scheduling strategies for different query applications. Since limited power is still a severe constraint in WSNs, the energy efficient scheduling problem is investigated in [76][77][78][79]. After that, the authors in [80][38][81][82] investigated the minimum latency query processing problem. Additionally, the authors in [83][84][85][86][87][88] aimed to improve network performance such as throughput and Quality of Service (QoS) when designing a query scheduling.

The previous researches made many contributions to the development of query schedul-

**SELECT** $\mathcal{R}_{1\text{.maxGasdensity}}$, $\mathcal{R}_{2\text{.maxTemperature}}$
**FROM** $\mathcal{R}_1$,$\mathcal{R}_2$
**WHERE** $\mathcal{R}_1$=(11$^{\text{th}}$-floor$\cup$12$^{\text{th}}$-floor) and $\mathcal{R}_2$=(12$^{\text{th}}$-floor$\cup$13$^{\text{th}}$-floor)

Figure 3.1. An MRQ example.

ing technologies. However, consider the following scenario: a WSN is deployed in a building, where the sensors at the $11^{th}$ and $12^{th}$ floors report gas leakages and the sensors at the $12^{th}$ and $13^{th}$ floors report fire alarms. In order to get the detailed information, the query shown in Fig.3.1 is disseminated, which is a query targeting data from different regions (subareas) asking for the maximum gas density of the $11^{th}$ and $12^{th}$ floors and the maximum temperature of the $12^{th}$ and $13^{th}$ floors. The aforementioned techniques ([77][78][79]-[83][84][76][38]) are inefficient when dealing with the above query. The reasons are as follows. First, in most existing works, a user disseminates a query by broadcasting the query to the entire network even though the interested data is from a particular small region. It wastes a large amount of energy, especially in large scale WSNs. Second, the scheduling strategies proposed in the existing works construct a single fixed scheduling tree consisting of all the sensor nodes. Avoiding collisions on a fixed global scheduling tree results in more latency. Hence, It is not suitable for scheduling multiple queries without correlations and targeting multiple regions simultaneously. Last, the assumptions on disseminating time of queries with temporal and spatial considerations in the existing works are no longer reasonable when dealing with queries without those characteristics.

In order to address the aforementioned issues, we investigate the Minimum Latency Data Aggregation Scheduling for Multi-Regional Query(ML-MRQS) problem in this paper. A Multi-Regional Query (MRQ) is defined as a query that targets interested data from multiple regions (overlapping may exist) of a WSN, where each region is a subarea of the WSN. Furthermore, for an MRQ, there is no limitation in a query disseminating time interval or assumptions on temporal or spatial correlations. The traditional multi-query processing can be thought of as a special case of the MRQ processing if the query region is the entire network. Therefore, compared with the traditional query model, the MRQ model is more practical and general. Consequently, we investigate the ML-MRQS problem which aims to get a collision-free scheduling plan for an MRQ with minimum latency. The contributions of this section is summarized as follows:

1. In this paper, we investigate the ML-MRQS problem for WSNs which is NP-Hard, and propose a heuristic collision free scheduling algorithm, called Multi-Regional Query Scheduling Algorithm (MRQSA). In MRQSA, we construct a CDS-based scheduling tree for each region. All those scheduling trees form a scheduling forest serving as the data transmission structure during the data transmission procedure. In addition, in order to improve the parallelism of MRQSA, we propose to assign higher priorities to selected sensor nodes which may cause multiple collisions. During the scheduling procedure, transmissions on different trees are scheduled concurrently. In order to avoid collisions, priority and interference are considered in the entire forest.

2. We theoretically analyze the latency of MRQSA which is upper bounded by $23A + B + C$ for an MRQ with $m$ query regions $\mathcal{R}_1, \mathcal{R}_2, ..., \mathcal{R}_m$, where $A = \max_{i=1}^{m} D_i^{left}$, $B = \max_{i=1}^{m}\{(23D_i + 5\Delta + 21)k_i\}$, $C = \sum_{i=1}^{m} H_i + 5\Delta - m + 17$, $m$ is the number of regions, $\Delta$ is the maximum node degree in the WSN, $D_i$ is the diameter of $\mathcal{R}_i$, $k_i$ is the maximum overlapped degree of sensor nodes in $\mathcal{R}_i$, $H_i$ represents the distance of $\mathcal{R}_i$ with respect to the sink, and $D_i^{left}$ is the diameter of the non-overlapped part of $\mathcal{R}_i$.

3. We also conduct extensive simulations to validate the performance of MRQSA. The simulation results indicate that the proposed MRQSA has a better performance compared with the most recently published multi-query scheduling algorithm C-DCQS [80]. MRQSA reduces latency by 49.3% on average for different number of query regions, 50.7% on average for different network densities, 42.7% on average for different region sizes, and 51.63% on average for different interference/transmission ranges compared with C-DCQS.

## 3.2 Problem Formulation

### 3.2.1 Network Model

We consider a connected dense WSN consisting of $n$ sensor nodes along with one *sink* (*base station*) denoted by $s$. Each node is equipped with a single, *half-duplex* radio. During a time slot, a node $a$ can either send or receive data (not both) to or from all directions. The *transmission/interference range* of any sensor node is defined as a *unit disk* with radius 1 centered at the node. Hence, we can use an undirected Unit Disk Graph (UDG) $G = (V, E)$ to represent the topology graph of a WSN, where $V$ represents the set of all the sensor nodes, then $|V| = n + 1$; $E$ denotes the *bidirectional* link among all the sensor nodes. $\forall a \in V$, $a$'s one-hop neighbor set is denoted by $\mathcal{N}(a) = \{b||a - b| \leq 1, b \in V, a \neq b\}$. An edge $(a, b) \in E$ exists if and only if $b \in \mathcal{N}(a)$.

We assume the network time is synchronized and slotted. Within a time slot, a data package can be sent from a sender and received by a receiver successfully as long as there is no interference. Additionally, we assume the global locations of all the sensor nodes are known.

### 3.2.2 Multi-Regional Query

Let $U$ represent the entire monitoring area of a WSN. A *Region* is defined as a *consecutive* subarea within $U$, *i.e.*, for a region $\mathcal{R}_i$ in a WSN, $\mathcal{R}_i \subseteq U$. Since the considered WSN is dense, the sensor nodes deployed in $\mathcal{R}_i$ is a connected component of $G$. For $\forall a \in V$, $\forall \mathcal{R}_i \subseteq U$, $a$ is called a *Regional Node* (RN) of $\mathcal{R}_i$ if and only if $a$ is deployed in $\mathcal{R}_i$. Furthermore, $\mathcal{S}(\mathcal{R}_i)$ is defined as the set of RNs in $\mathcal{R}_i$.

**SELECT** $\mathcal{R}_1$.ds, $\mathcal{R}_2$.ds, …, $\mathcal{R}_m$.ds
**FROM** $\mathcal{R}_1 \cup \mathcal{R}_2 \cup \ldots \cup \mathcal{R}_m$
**[WHERE** predicates**]**

Figure 3.2. The formal expression of an MRQ structure.

**Definition 3.2.1. Multi-Regional Query** *(MRQ). An MRQ is a query targeting at user interested data from multiple regions in a WSN. The formal expression of an MRQ is given in Fig.3.2 which shows an MRQ with m query regions.* SELECT *denotes the query results, where $\mathcal{R}_i.ds$ means the user interested data in region $\mathcal{R}_i$.* FROM *shows the multiple regions where the MRQ should be disseminated. The query conditions are specified in* WHERE.

When an MRQ is disseminated to regions specified by the *FROM* clause, the RNs deployed in $\mathcal{R}_i$ only have to collect $\mathcal{R}_i.ds$. An example MRQ is shown in Fig.3.1. The MRQ targets at two regions with $\mathcal{R}_1 = \{11^{th} \bigcup 12^{th}\}$ floors and $\mathcal{R}_2 = \{12^{th} \bigcup 13^{th}\}$ floors. The purpose is to obtain the maximum gas density of $\mathcal{R}_1$ and the maximum temperature of $\mathcal{R}_2$. From this example, first, we can see that this MRQ targets at data with no correlations from different regions simultaneously. Secondly, the target regions in an MRQ may overlap, such as $\mathcal{R}_1 \bigcap \mathcal{R}_2 = \{12^{th}\}$ floor. Thirdly, since the target regions of an MRQ are known, the MRQ can be directly disseminated to $\mathcal{R}_1 \bigcup \mathcal{R}_2 = \{11^{th} \bigcup 12^{th}\} \bigcup \{12^{th} \bigcup 13^{th}\} = \{11^{th} \bigcup 12^{th} \bigcup 13^{th}\}$ floors.

According to the above illustration, an MRQ only cares about data sets collected by sensor nodes deployed in the target query regions. Hence, MRQ has the following benefits. The query regions of an MRQ are known. Therefore, an MRQ can be directly disseminated to its corresponding query regions. Besides, for sensor nodes that are deployed outside of the query regions, they do not need to be involved in the MRQ unless they are relay nodes that help to transfer query results to the sink. In this way, energy consumption is reduced through reducing unnecessary data transmissions. Finally, since less sensor nodes are involved in data transmission, the possibility of reducing transmission conflicts is increased.

In this paper, we investigate how to process an MRQ, which is dynamically disseminated into multiple regions of a WSN. In order to conserve network resources, query results are transmitted to the sink with aggregation. That is, on the routing tree, a parent node has to wait for all of its children to finish their data transmissions before starting its own transmission. We make no assumption on the correlations among data collected from different regions. Hence, the data can be aggregated during data transmission only if they are from the same region.

### 3.2.3 Problem Definition

Let $\mathcal{R}_i$ represent a region. A *Regional Query Tree* (RQT) $\mathcal{T}_i$ is a tree constructed within $\mathcal{R}_i$ such that $\forall a \in \mathcal{S}(\mathcal{R}_i) \Leftrightarrow a \in \mathcal{T}_i$. The root of $\mathcal{T}_i$ is denoted by $r_{\mathcal{T}_i}$. Considering an MRQ with $m$ query regions $\mathcal{R}_1$, $\mathcal{R}_2$, ..., $\mathcal{R}_m$, $\forall i, 1 \leq i \leq m$, $\mathcal{T}_i$ denotes the RQT constructed in $\mathcal{R}_i$ with $\mathcal{S}(\mathcal{R}_i)$. The *Multi-Regional Query Forest* (MRQF) denoted by $\mathcal{F}$ of an MRQ is the set of all the RQTs of this MRQ, *i.e.* $\mathcal{F} = \{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_m\}$. The set of all the sensor nodes in $\mathcal{F}$ is denoted by $\mathcal{S}(\mathcal{F})$, *i.e.* $\mathcal{S}(\mathcal{F}) = \mathcal{S}(\mathcal{R}_1) \bigcup \mathcal{S}(\mathcal{R}_2) \bigcup ... \bigcup \mathcal{S}(\mathcal{R}_m)$. The defined MRQF (RQTs) serves as the routing structure in our proposed algorithm.

The *Overlapped Degree* (OD) of an RN $a$, denoted by $od_a$, is defined as the number of RQTs it belongs to. For any region $\mathcal{R}_i$ with $k$ RNs $n_1, n_2, ..., n_k$, let $\mathcal{O} = \{a | od_a = min(od_{n_1}, od_{n_2}, ..., od_{n_k})\}$, which is the set of RNs in $\mathcal{R}_i$ with the smallest OD. $\mathcal{A}$ is the set of RNs in $\mathcal{O}$ that are closest to the sink (the distance of each RN to the sink is measured by hop-distance). An *Accessing Node* (AN) is defined as the RN in $\mathcal{A}$ with the smallest ID.

Given an MRQF $\mathcal{F} = \{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_m\}$, let $\mathcal{I} = \bigcup_{i=1}^{m} \{a | a \in \mathcal{T}_i, \exists b, b \in \mathcal{T}_j, i \neq j, a \in \mathcal{N}(b), 1 \leq i, j \leq m\} \bigcup \{a | od_a \geq 2\}$ represent the set of RNs that have at least one one-hop neighbor in another RQT and RNs whose ODs are no less than 2. $\forall a \in \mathcal{T}_i$, $\mathcal{D}(a)$ is defined as $a$'s descendant set which contains the RNs in a subtree of $\mathcal{T}_i$ rooted at $a$. The *Overlapped Interference Set* (OIS) of an MRQF is denoted by $\mathcal{S}(Ois) = \mathcal{I} \bigcup \mathcal{D}(\mathcal{I})$, where $\mathcal{D}(\mathcal{I})$ is the set of descendants of RNs in $\mathcal{I}$. The reason of including the descendants of $\mathcal{I}$ relies on the transmission constraint of aggregating data within a region during the data transmission.

Fig.3.3 shows an MRQF with two RQTs $\mathcal{T}_1$ and $\mathcal{T}_2$ which are constructed within query regions $\mathcal{R}_1$ and $\mathcal{R}_2$, respectively. An RN is represented by a black circle with its ID inside. The ANs are black and the RNs in the overlapped interference set $\mathcal{S}(Ois)$ are gray. From Fig.3.3, we can see that $od_3 = 1$ since it is only on $\mathcal{T}_1$, while $od_{14} = 2$ because it is on both $\mathcal{T}_1$ and $\mathcal{T}_2$. RN 0 is the AN of $\mathcal{T}_1$ since it has the smallest OD and is closest to

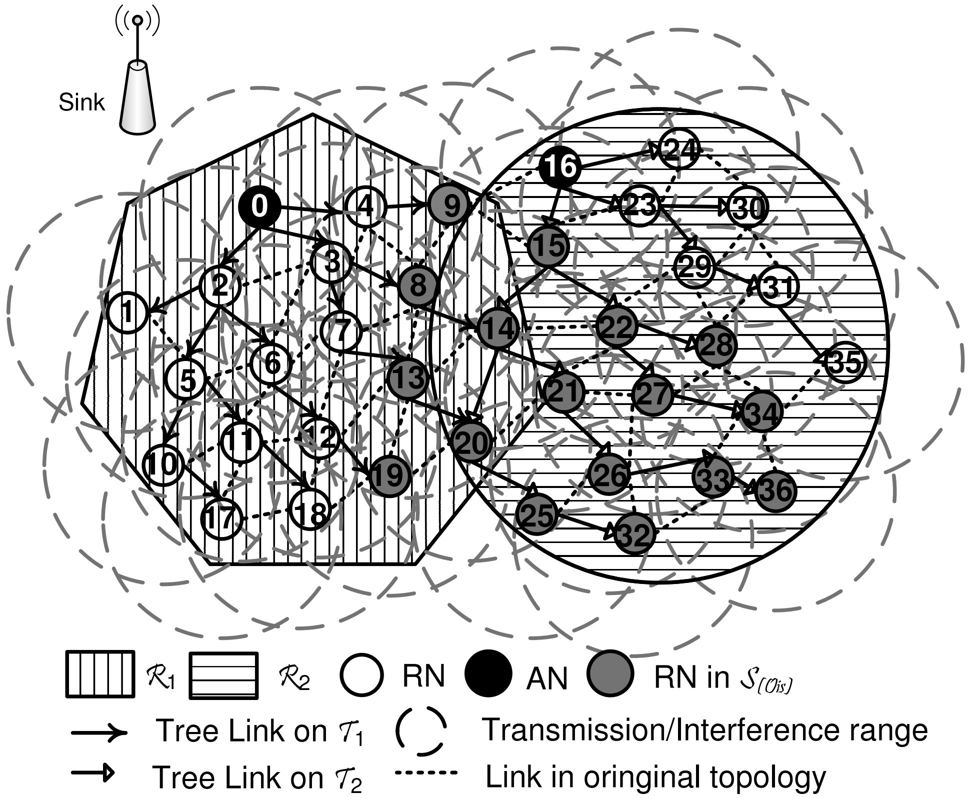



Figure 3.3. An example MRQF.

the sink and the AN of $\mathcal{T}_2$ is RN 16. The OIS of the MRQF is $\mathcal{S}(Ois) = \mathcal{I} \bigcup \mathcal{D}(\mathcal{I}) = \{8, 9, 13, 14, 15, 19, 20, 21, 22, 25, 26, 27, 28, 32, 33, 34\}$, where each RN in $\mathcal{I}$ has at least one one-hop neighbor in another RQT, and the RNs in $\mathcal{D}(\mathcal{I})$ are descendants of RNs in $\mathcal{I}$.

**Interference Model**: In this paper, we employed the classical interference model which is widely used in the previous works [77][84][76][38]. Given two simultaneous transmissions $a \to b$ and $c \to d$ , where $a$ and $c$ are senders, and $c$ and $d$ are receivers. Transmission $c \to d$ is said to conflict with transmission $a \to b$ if and only if $d \in \mathcal{N}(a)$ or $b \in \mathcal{N}(c)$. The the definition of a *Non-Concurrent Set* is given below.

**Definition 3.2.2. Non-Concurrent Set** *(NCS). Given an MRQ with $m$ regions, its corresponding MRQF is $\mathcal{F} = \{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_m\}$. $\forall a \in \mathcal{T}_i$, $\mathcal{P}_i(a)$ is the parent node of $a$ in $\mathcal{T}_i$. Then, transmission $a \to \mathcal{P}_i(a)$ 's NCS is defined as $\mathcal{NCS}(a \to \mathcal{P}_i(a)) = \mathcal{N}_{\mathcal{F}}(\mathcal{P}_i(a)) \bigcup \{\bigcup_{b \in \mathcal{N}_{\mathcal{F}}(a)} \mathcal{C}h_{\mathcal{F}}(b)\} \bigcup \{a, \mathcal{P}_i(a)\} \bigcup \{\mathcal{C}h_{\mathcal{F}}(a)\}$, where $\mathcal{N}_{\mathcal{F}}(\mathcal{P}_i(a))$ is $\mathcal{P}_i(a)$'s one-hop neighbor set in $\mathcal{F}$, $\mathcal{N}_{\mathcal{F}}(a)$ is $a$'s* *one-hop neighbor set in* $\mathcal{F}$*, and* $\mathcal{C}h(b)$ *is* $b$*'s children node set in* $\mathcal{F}$*.*

Specially, RNs may be in multiple RQTs, for a particular time slot, any RN can participate in only one transmission. Hence, for $a$, $\mathcal{C}h(a)$ and $\mathcal{P}_i(a)$ are also needed to be included in $\mathcal{NCS}(a \rightarrow \mathcal{P}_i(a))$, which is different from the NCS defined in the traditional case based on a single fixed scheduling tree. Then, we define the Minimum Latency Multi-Regional Query Scheduling problem (ML-MRQS) as follows.

**Definition 3.2.3. Minimum Latency Multi-Regional Query Scheduling** *(ML-MRQS). Given an MRQ targets at* $m$ *regions* $\mathcal{R}_1, \mathcal{R}_2, ..., \mathcal{R}_m$*, the RQT constructed in* $\mathcal{R}_i$ *is represented by* $\mathcal{T}_i$*.* $\mathcal{F} = \{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_m\}$ *is the MRQF. Let* $\mathcal{S}ch_{\mathcal{T}_i}$ *be the set of all child* $\rightarrow$ *parent transmissions in* $\mathcal{T}_i$*,* $\mathcal{S}ch_{\mathcal{F}} = \mathcal{S}ch_{\mathcal{T}_1} \bigcup \mathcal{S}ch_{\mathcal{T}_2} \bigcup ... \bigcup \mathcal{S}ch_{\mathcal{T}_m}$*, and* $\mathcal{S}ch^{t_j}_{\mathcal{T}_i}$ *represents the set of transmissions in* $\mathcal{T}_i$ *that are scheduled at time* $t_j$*. An ML-MRQS is a sequence of transmission sets denoted by* $\mathcal{SCH} = \{\{\mathcal{S}ch^{t_1}_{\mathcal{T}_1}, \mathcal{S}ch^{t_1}_{\mathcal{T}_2}, ..., \mathcal{S}ch^{t_1}_{\mathcal{T}_m}\}, \{\mathcal{S}ch^{t_2}_{\mathcal{T}_1}, \mathcal{S}ch^{t_2}_{\mathcal{T}_2}, ..., \mathcal{S}ch^{t_2}_{\mathcal{T}_m}\}, ..., \{\mathcal{S}ch^{t_L}_{\mathcal{T}_1}, \mathcal{S}ch^{t_L}_{\mathcal{T}_2}, ...,$ *satisfying the following conditions:*

(i). $\mathcal{S}ch^{t_i}_{\mathcal{T}_k} \bigcap \mathcal{S}ch^{t_j}_{\mathcal{T}_k} = \emptyset$*,* $\forall i \neq j$*,* $1 \leq k \leq m$*,* $1 \leq i, j \leq L$*.*

(ii). $\forall a \rightarrow \mathcal{P}(a) \in \mathcal{S}ch^{t_i}_{\mathcal{T}_k}$*,* $\forall b \rightarrow \mathcal{P}(b) \in \mathcal{S}ch^{t_i}_{\mathcal{T}_j}$*,* $a \notin \mathcal{NCS}(b \rightarrow \mathcal{P}(b))$ *and* $b \notin \mathcal{NCS}(a \rightarrow \mathcal{P}(a))$*, where* $\forall k \neq j$*,*$1 \leq k, j \leq m$*,* $1 \leq i \leq L$*.*

(iii). $\bigcup_{i=1}^{L} \mathcal{S}ch^{t_i}_{\mathcal{T}_k} = \mathcal{S}ch_{\mathcal{T}_k}$*, where* $1 \leq k \leq m$*.*

(iv). $\bigcup_{i=1}^{L} \bigcup_{k=1}^{m} \mathcal{S}ch^{t_i}_{\mathcal{T}_k} = \mathcal{S}ch_{\mathcal{F}}$*.*

(v). *Data are aggregated from* $\bigcup_{k=1}^{m} \mathcal{S}ch^{t_i}_{\mathcal{T}_k}$ *to* $\mathcal{S}ch_{\mathcal{F}}$ - $\bigcup_{t=1}^{t_i} \bigcup_{k=1}^{m} \mathcal{S}ch^{t}_{\mathcal{T}_k}$ *at time slot* $t_i$*, for all* $t = 1, 2, ...L$*. Data are aggregated to* $\bigcup_{k=1}^{m} r_{\mathcal{T}_k}$ *in* L *time slots, where* $L$ *is the latency of the MRQS.*

(vi). $L$ *is minimized.*

In Definition 5.3.1, constraint $(i)$ specifies that a particular transmission in an RQT can only be scheduled once. Constraint $(ii)$ guarantees that at a particular time, the transmissions scheduled in different RQTs should have no collisions. Constraint $(iii)$ forces all the transmissions in an RQT getting their scheduling time slots, while constraint (*iv*) guarantees that all the transmissions in an MRQF are scheduled. Constraint (*v*) denotes the transmission set scheduled at time $t_i$, and it also guarantees that the query result of each region is collected to the AN. Constraint (*vi*) donates that the latency should be minimized.

The ML-MRQS problem under the UDG model is NP-hard. This is based on the fact that the *minimum latency data aggregation scheduling* problem is NP-hard [89] under the UDG model, which can be considered as a special case of ML-MRQS when the target region of an MRQ is the entire network.

## 3.3 Multi-Regional Query Scheduling

In this section, we propose a scheduling algorithm for ML-MRQS which consists of two phases. First, an MRQF is constructed which serves as the data transmission structure during the data transmission procedure. Subsequently, a collision-free scheduling plan is generated.

### 3.3.1 Construction of MRQF

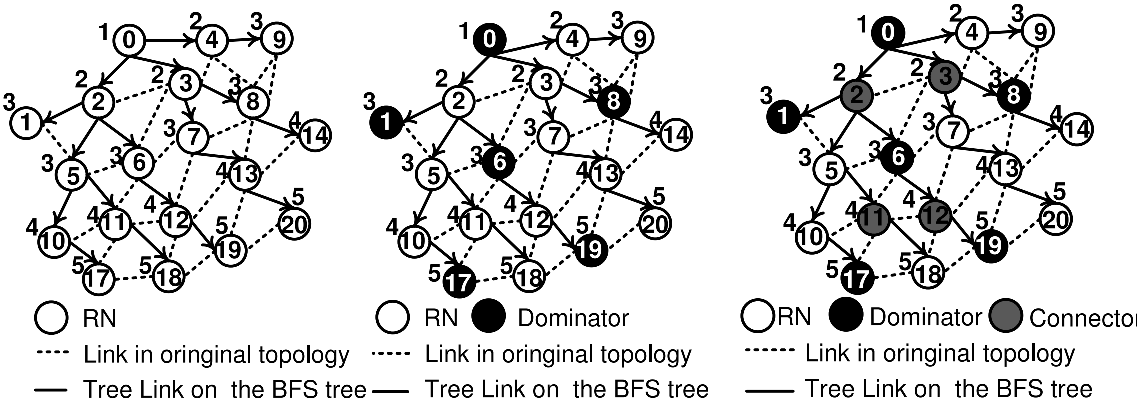
In a WSN represented by $G = (V, E)$, a *Dominating Set* (DS) is defined as a subset of sensor nodes in the WSN with the property that for each sensor node in the WSN, it is either in the subset or adjacent to at least one node in the subset. If the induced subgraph of a DS is connected, we call this DS a Connected Dominating Set (CDS). Nodes in a CDS are called dominators or connectors and others are called dominatees. Ever since the CDS was introduced, it has been widely used in many applications as a virtual backbone for efficient routing [90][91].

In order to obtain an efficient routing tree for the data transmission of an ML-MRQS, we use a CDS-based tree as the routing tree. There are many existing works studying how to construct a CDS [92][93][94][95]. In this paper, the method proposed in [40] is adopted. Note that, other algorithms can also be used. The RQTs constructed in all the regions form an MRQF, and are used as the data transmission structure of the ML-MRQS.

The construction of $\mathcal{T}_i$ in $\mathcal{R}_i$ can be briefly described as follows:

Step 1: Find the AN $a$ of $\mathcal{R}_i$.

Step 2: Build a *Breadth First Searching* (BFS) tree rooted at $a$ with all the RNs in $\mathcal{R}_i$. During the BFS tree construction procedure, obtain a *Maximal Independent Set* (MIS) $\mathcal{M}_i$. The RNs in $\mathcal{M}_i$ colored black are dominators.




(a) (b) (c)

Figure 3.4. The construction of a CDS-based routing tree.

Step 3: Find a set of sensor nodes $\mathcal{C}_i$ to connect the nodes in $\mathcal{M}_i$ to form a CDS. The RNs in $\mathcal{C}_i$ colored gray are connectors and satisfy the fact that for any RN in $\mathcal{C}_i$, its parent node and children nodes are dominators.

Step 4: Color RNs in $\mathcal{S}(\mathcal{R}_i) \backslash (\mathcal{M}_i \bigcup \mathcal{C}_i)$, which are dominatees, as white, and pick each dominatee $b$ a dominator from $\mathcal{N}(b) \bigcap \mathcal{M}_i$ as its parent. This step allocates each dominatee to a dominator, such that every dominatee can transmit data to its allocated dominator during the data transmission process.

Fig.3.4 shows the procedure of constructing a CDS-based routing tree using $\mathcal{T}_1$ in Fig.3.3, where the number inside each circle is the node's ID, the number besides the circle is the depth, and links are represented by lines or dashed lines. Fig.3.4(a) shows the construction of a BFS tree rooted at the AN (RN 0). In Fig.3.4(b), a dominating set is selected with which the RNs in the WSN is either in the dominating set or have a one-hop neighbor in the dominating set, where the selected RN in this step are dominators. In order to guarantee data transmission, a connector set is chosen to make the dominating set connected as shown in Fig.3.4(c), where the RNs selected in this step are connectors. The next step is to allocate each dominatee (RNS except dominators and connectors) to a dominator, so that the dominatee only needs to forward data to its allocated dominator as shown in Fig.12(a), where the data structures of the RNs are updated based on the constructed CDS-based routing tree, *e.g.* the depth of a dominatee is updated to one more than the depth of its parent (dominator).

Particularly, due to the existence of overlapped regions, RQTs in an MRQF may overlap with each other (as shown in Fig.3.3), which is different from a traditional forest. Therefore, the ML-MRQS problem is much more challenging than the traditional scheduling problem because RNs in $\mathcal{S}(Ois)$ may cause collisions more than once.

### 3.3.2 MRQSA

**Scheduling Initialization** At the beginning of the initialization, the RNs in $\mathcal{F}$ are divided into two parts $\mathcal{S}(\mathcal{O}is)$ and $\mathcal{F}^{left}$, where $\mathcal{S}(\mathcal{O}is)$ is the OIS of an MRQF, and $\mathcal{F}^{left} = \mathcal{F}\backslash\mathcal{S}(\mathcal{O}is)$. Let $\mathcal{T}_i^{left} = \mathcal{T}_i\backslash\mathcal{S}(\mathcal{O}is)$, then $\mathcal{F}^{left}$ can be represented by $\{\mathcal{T}_1^{left}, \mathcal{T}_2^{left}, ..., \mathcal{T}_m^{left}\}$ satisfying $\forall i \neq j$, $\mathcal{T}_i^{left} \bigcap \mathcal{T}_j^{left} = \emptyset$. The behind-the-scene meaning of separating $\mathcal{S}(\mathcal{O}is)$ from $\mathcal{F}^{left}$ is to improve the parallelism of MRQS. The RNs in $\mathcal{S}(\mathcal{O}is)$ may influence the data transmissions of multiple RQTs because their interference range may overlap with multiple regions. Therefore, the RNs in $\mathcal{S}(\mathcal{O}is)$ are given a higher priority in MRQSA. It follows that the RNs in $\mathcal{S}(\mathcal{O}is)$ may finish their scheduling first. After that, the scheduling among the $\mathcal{T}^{left}$s can be paralleled without collision.

The $\mathcal{S}(\mathcal{O}is)$ can be specified by checking all the RNs in the MRQF: from $\mathcal{T}_1$ to $\mathcal{T}_m$. If one RN in $\mathcal{T}_i$ has a one-hop neighbor who is in $\mathcal{T}_j$ $(i \neq j)$ or the OD of the RN is no less than 2, mark it as an RN in $\mathcal{S}(\mathcal{O}is)$, *i.e.*, if $\forall a \in \mathcal{T}_i$, $\exists b, b \in \mathcal{T}_j, a \in \mathcal{N}(b), i \neq j$, or $od_a \geq 2$, put $a$ in $\mathcal{S}(\mathcal{O}is)$. Subsequently, find the descendants of all the elements in $\mathcal{S}(\mathcal{O}is)$.

**Scheduling Algorithm** After the initialization phase, all information needed for scheduling is collected. For simplicity, let $\mathcal{S}ch^l$ represent the schedule plan in the $l$-th iteration, which is a set of transmissions scheduled in the $l$-th time slot, $l = 1, 2, ..., L$. $\mathcal{NCS}(\mathcal{S}ch^l)$ represents the NCS of the transmissions in $\mathcal{S}ch^l$. Let $\mathcal{T}_i^{Ois} = \mathcal{T}_i \bigcap \mathcal{S}(Ois)$.

A transmission $a \to b$ on $\mathcal{T}_i$ is said ***ready*** to be scheduled if RN $a$ is a leaf node or $a$'s children nodes in $\mathcal{T}_i$ have already obtained their scheduling time slots and the priority of $a$ is the highest compared with other scheduling candidates in $\mathcal{T}_i$. The priority is given to an RN in the order of "RN in the OIS", "color is white", "depth is larger", and "ID is smaller". The scheduling algorithm, Multi-Regional Query Scheduling Algorithm (MRQSA), is shown in *Algorithm 1*.

---

**Algorithm 1:** MRQSA

---

**input** : MRQF
**output**: schedule plan $\mathcal{S} = \{\mathcal{S}ch^1, \mathcal{S}ch^2, ..., \mathcal{S}ch^m\}$ and latency $L$

1 $\mathcal{S} \leftarrow \emptyset$;
2 $l \leftarrow 0$;
3 **while** $\mathcal{F}^{left} \neq \emptyset$ *or* $\mathcal{S}(\mathcal{O}is) \neq \emptyset$ **do**
4     $l{+}{+}$;
5     $\mathcal{S}ch^l \leftarrow \emptyset$;
6     $\mathcal{NCS}(\mathcal{S}ch^l) \leftarrow \emptyset$;
7     Schedule $\mathcal{S}(\mathcal{O}is)$ at the $l$-th time slot, find the conflict-free transmissions in $\mathcal{S}(\mathcal{O}is)$ and add these transmissions to $\mathcal{S}ch^l$ followed by updating $\mathcal{NCS}(\mathcal{S}ch^l)$ (as shown in *Algorithm 2*) ;
8     Schedule $\mathcal{F}^{left}$ at the $l$-th time slot, find the conflict-free transmissions in $\mathcal{F}^{left}$ and add these transmissions to $\mathcal{S}ch^l$ (as shown in *Algorithm 2*);
9     Add $\mathcal{S}ch^l$ to $\mathcal{S}$;
10 return $\mathcal{S}$, $L = l$;

---

*Algorithm 1* shows that MRQSA takes the MRQF as an input. It runs iteratively from line 3 to line 9 and finally outputs the scheduling plan $\mathcal{S}$ and the latency $L$ of $\mathcal{S}$. During the $l$-th iteration ($l = 1, 2, ..., L$), the first step is to initialize $\mathcal{S}ch^l$ and $\mathcal{NC}(\mathcal{S}ch^l)$ to be empty sets (line 5-6). Subsequently, schedule the transmissions whose senders are RNs in $\mathcal{S}(Ois)$ and can be scheduled in the $l$-th iteration as line 7 shows. Finally, the transmissions whose senders are RNs in $\mathcal{F}^{left}$ can be scheduled in the $l$-th iteration as shown in line 8. MRQSA generates a scheduling plan iteratively until all the *child* $\rightarrow$ *parent* transmissions in each RQT are scheduled. The method of scheduling $\mathcal{S}(\mathcal{O}is)$ or $\mathcal{F}^{left}$ is presented in *Algorithm 2*.

---

**Algorithm 2:** Schedule $\mathcal{S}(\mathcal{O}is)$ or $\mathcal{F}^{left}$ in the $l$-th slot

---

**input** : $\mathcal{OS}$(Objective Set)-$\mathcal{S}(\mathcal{O}is)$ or $\mathcal{F}^{left}$, $\mathcal{NCS}(\mathcal{S}ch^l)$
**output**: schedule plan $\mathcal{S}ch^l$ for time slot $l$

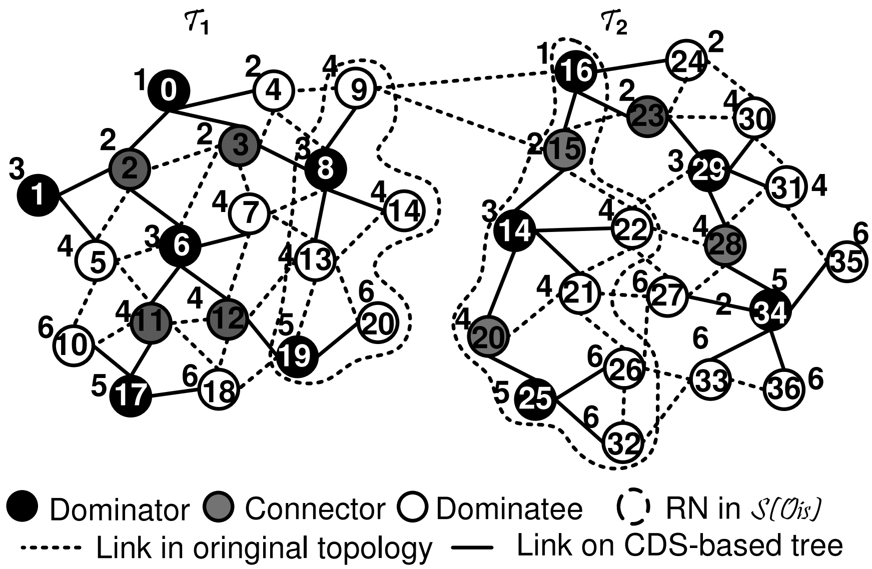
1 $flag = 1$;
2 **while** $flag \neq 0$ **do**
3     $flag \leftarrow 0$;
4     **for** $i : 1 \rightarrow m$ **do**
5         **if** $a \in \mathcal{OS}$ *and* $a \rightarrow P_i(a))$ *is ready and* $a \notin \mathcal{NCS}(\mathcal{S}ch^l)$ **then**
6             $flag \leftarrow 1$;
7             $\mathcal{S}ch^l \leftarrow \mathcal{S}ch^l \bigcup a \rightarrow \mathcal{P}_i(a)$;
8             $\mathcal{NCS}(\mathcal{S}ch^l) \leftarrow \mathcal{NCS}(\mathcal{S}ch^l) \bigcup \mathcal{NCS}(a \rightarrow \mathcal{P}_i(a))$;
9             $od_a - -$;
10             **if** $od_a == 0$ **then**
11                 $\mathcal{OS} \leftarrow \mathcal{OS} \backslash \{a\}$;

12 return $\mathcal{S}ch^l$, $\mathcal{NCS}(\mathcal{S}ch^l)$;

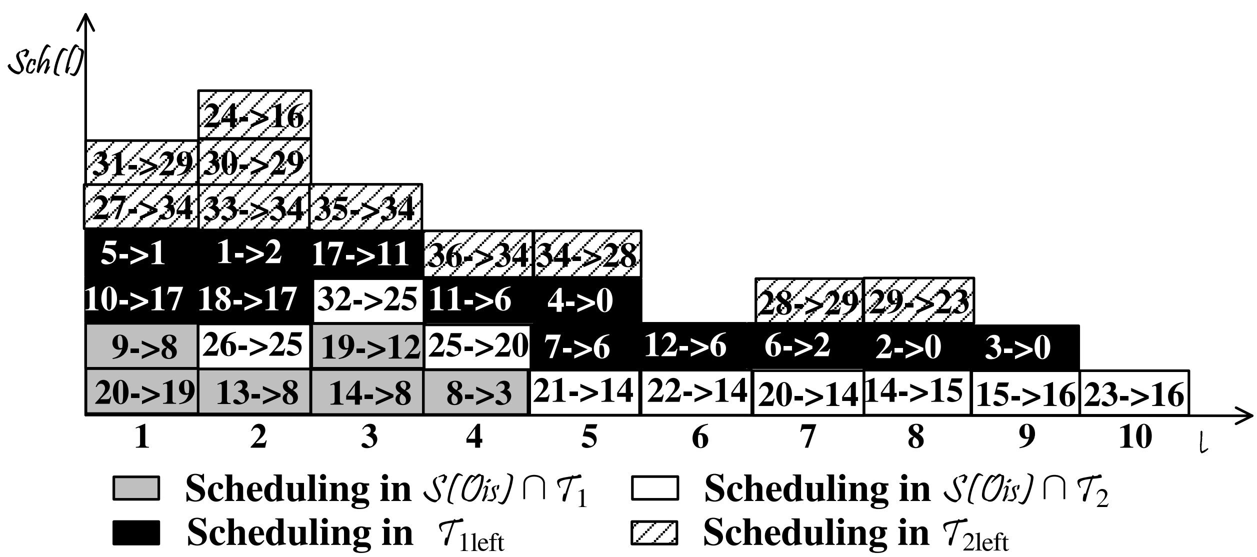
---

*Algorithm 2* takes the $\mathcal{OS}$ (Objective Set) as an input, where $\mathcal{OS} = \mathcal{S}(\mathcal{O}is)$ when scheduling $\mathcal{S}(\mathcal{O}is)$ and $\mathcal{OS} = \mathcal{F}^{left}$ when scheduling $\mathcal{F}^{left}$, respectively. The output of *Algorithm 2* is the transmissions whose senders are in the considered $\mathcal{OS}$ that can be scheduled in the $l$-th iteration. From $\mathcal{T}_1$ to $\mathcal{T}_m$ (line 4), *Algorithm 2* runs iteratively to find a ready transmission $a \rightarrow \mathcal{P}_i(a)$ with the highest priority and $a \notin \mathcal{NCS}(\mathcal{S}ch^l)$ as line 5 shown. If such a transmission is found, add it to $\mathcal{S}ch^l$ (line 7). Subsequently, the $\mathcal{NCS}(\mathcal{S}ch^l)$ is updated to be $\mathcal{NCS}(\mathcal{S}ch^l) \bigcup \mathcal{NCS}(a \rightarrow \mathcal{P}_i(a))$ (line 8). If all the transmissions in the MRQF involving $a$ are scheduled, $a$ should be removed from the $\mathcal{OS}$ (line 10-11). The procedure will continue until no more ready transmission that can be scheduled without collision is found in this iteration.

Fig.12(a) shows the CDS-based RQTs derived from Fig.3.3, where the numbers beside the RNs represent the depth while the numbers within the circles represent their IDs. At the beginning of the 1-st slot, $\mathcal{S}ch^1$ and $\mathcal{NCS}(\mathcal{S}ch^1)$ are initialized to be emp-




(a) RQTs.




(b) Scheduling plan.

ty sets. Starting from $\mathcal{T}_1$, we first search for ready transmissions $a \rightarrow \mathcal{P}_i(a)$ where $a \in \mathcal{T}_1^{Ois}$. RN 14 is selected since it is ready and its priority is the highest. Then, transmission $20 \rightarrow 19$ is added to $\mathcal{S}ch^1$ and $\mathcal{NCS}(\mathcal{S}ch^1) = \emptyset \bigcup \mathcal{NCS}(20 \rightarrow 19) = \{12, 13, 18, 20\} \bigcup \{20, 21, 22, 26, 32\} \bigcup \{19, 20\} \bigcup \emptyset = \{12, 13, 18, 19, 20, 21, 22, 26, 32\}$, where $\{12, 13, 18, 20\}$ is 19's one-hop neighbor set, the second set $\{20, 21, 22, 26, 32\}$ is 20's one-hop neighbors' children set in the MRQF, the third set $\{19, 20\}$ is the set of sender and receiver of the transmission, and $\emptyset$ is 20's children in $\mathcal{F}$. Similarly, the next ready transmission $9 \rightarrow 8$ is found and added to $\mathcal{S}ch^1$. Then $\mathcal{NCS}(9 \rightarrow 8) = \{3, 4, 7, 8, 9, 13, 14, 15, 22, 23, 24\}$ is added to $\mathcal{NCS}(\mathcal{S}ch^1)$, *i.e.*, $\mathcal{NCS}(\mathcal{S}ch^1) = \{12, 13, 18, 19, 20, 21, 22, 26, 32\} \bigcup \{3, 4, 7, 8, 9, 13, 14, 15, 22, 23, 24\} = \{3, 4, 7, 8, 9, 12, 13, 14, 15, 18, 19, 20, 21, 22, 23, 24, 26, 32\}$. Then we check the rest transmissions whose senders are in $\mathcal{T}_1^{Ois}$. It turns out that no more transmissions can be scheduled. Then MRQSA moves to $\mathcal{T}_2^{Ois}$. Unfortunately, there are no transmissions in $\mathcal{T}_2^{Ois}$ that can be scheduled without collision. Subsequently, MRQSA moves back to check $\mathcal{T}_i^{left}$, and continues from $\mathcal{T}_1^{left}$ to $\mathcal{T}_2^{left}$ to check the transmissions whose senders are in $\mathcal{T}_i^{left}$. Transmissions $10 \rightarrow 17$ and $5 \rightarrow 1$ are found in $\mathcal{T}_1$ and transmissions $27 \rightarrow 34$ and $31 \rightarrow 29$ are found in $\mathcal{T}_2$. The 1-st iteration stops here since no more transmissions can be scheduled without collision. MRQSA repeats this procedure until all the transmissions in the MRQF are scheduled. Fig.12(b) shows the entire scheduling plan generated by MRQSA, where the $x$-axis represents the time slot, and $y$-axis represents transmissions scheduled in each corresponding time slot.

After obtaining the scheduling plan, every transmission can be scheduled in its assigned time slot.

In spite of the fact that we divide MRQF into two parts logically, the scheduling of $\mathcal{S}(\mathcal{O}is)$ and $\mathcal{F}^{left}$ are not necessarily be implemented sequentially. When generating a scheduling plan in each iteration, transmissions with senders in $\mathcal{S}(\mathcal{O}is)$ have a higher priority, but transmissions whose senders are in $\mathcal{F}_{left}$ can also be scheduled as long as they are ready and will not cause collisions with the current plan. This strategy aims to improve the parallelism of the MRQSA to minimize the latency. Particularly, the priority strategy guarantees the new added transmissions whose senders are dominators or connectors will not conflict with the scheduled transmissions whose senders are dominatees, and it also guarantees the scheduling of the transmission whose sender has a smaller depth will not conflict with a scheduled transmission whose sender has a higher depth.

### 3.3.3 Performance Analysis

In this subsection, we analyze the latency performance of MRQSA. Compared with traditional query scheduling methods, an MRQ can be directly disseminated to the query regions instead of the entire network. Intuitively, this procedure will reduce latency and energy consumption. Therefore, we only focus on the latency performance of MRQSA consisting of the following three parts: the latency of scheduling $\mathcal{F}^{left}$, the latency of scheduling $\mathcal{S}(\mathcal{O}is)$, and the latency of scheduling ANs to transmit the final aggregated values to the sink.

For convenience, The *diameter* of a region is defined as the maximum distance between any two RNs in this region. Let $D_i^{left}$ represent the diameter of the non-overlapped part of region $\mathcal{R}_i$ ($\mathcal{T}_i^{left}$), $D_i$ be the diameter of $\mathcal{R}_i$, $k_i$ be the maximum overlapped degree of $\mathcal{T}_i^{Ois}$ which is defined as the maximum OD of the RNs in $\mathcal{T}_i^{Ois}$, and $H_i$ represent the distance of $\mathcal{R}_i$ with respect to the sink, which is the hop-distance from the AN of $\mathcal{T}_i$ to the sink.

The following lemma shows some properties of the constructed CDS.

**Lemma 1.** The following statements are true.

*(*i*).* Each connector has at most 5 one-hop neighbors which are dominators *[35].*

*(*ii*).* Each dominator has at most 20 two-hop neighbors which are dominators *[35].*

*(*iii*).* Considering the CDS-based tree excluding the dominatees, the nodes with depth $2i+1$ (odd) are dominators and the ones with depth $2i$ (even) are connectors *[39].*

*(*iv*).* The depth of a CDS-based tree constructed within a region is no more than two times of the diameter of the region, where the diameter of the region is the maximum distance between any two sensor nodes in this region *[76].*

**Lemma 2.** *The latency of scheduling $\mathcal{F}^{left}$ is no more than* $23\max_{i=1}^{m} D_i^{left} + 5\Delta + 17$.

*Proof.* Based on the aforementioned illustration, $\mathcal{F}^{left} = \{\mathcal{T}_1^{left}, \mathcal{T}_2^{left}, ..., \mathcal{T}_m^{left}\}$ satisfies $\forall i \neq j$, $\mathcal{T}_i^{left} \bigcap \mathcal{T}_j^{left} = \emptyset$. Since the $m$ subsets of $\mathcal{F}^{left}$ are pairwise disjoint, the scheduling of each subset is independent with others. As a consequence, the latency of scheduling $\mathcal{F}^{left}$ actually depends on the latency of $\mathcal{T}_i^{left}$ whose latency is the largest among all the subsets. Additionally, since the RNs in $\mathcal{T}_i^{Ois}$ are leaves or subtrees, each $\mathcal{T}_i^{left}$ is still a CDS-based tree based on the construction procedure of an MRQF in Section 3.3.1. According to MRQSA, the scheduling of $\mathcal{T}_i^{left}$ consists of two phases:

(1) Scheduling from dominatees to dominators

For any transmission $a \rightarrow \mathcal{P}_i(a)$, where $a$ is a dominatee and $\mathcal{P}_i(a)$ is a dominator, $\mathcal{NCS}(a \rightarrow \mathcal{P}_i(a)) = \mathcal{N}_{\mathcal{T}_i}(\mathcal{P}_i(a)) \bigcup \{\bigcup_{b \in \mathcal{N}_{\mathcal{T}_i}(a) - \mathcal{C}h(a)} \mathcal{C}h(b)\} - \{a, \mathcal{P}_i(a)\}$. Intuitively, $\|\mathcal{N}_{\mathcal{T}_i}(\mathcal{P}_i(a))\| \leq \Delta - 1$, where $\| \cdot \|$ is the cardinality. Meanwhile, among $a$'s one-hop neighbors, only dominators or connectors may have dominatee children. Based on Lemma 1, the number of children of $a$'s one-hop neighbors which are connectors is bounded by 20, and the number of children of $a$'s one-hop neighbors which are dominators is bounded by $4\Delta$ ($a$'s parent is excluded). Hence, the latency of this phase is at most $20+4\Delta+\Delta-1+1 = 5\Delta+20$.

(2) Scheduling between dominators and connectors

After the first phase, all the transmissions from dominatees to their parents are scheduled. Let $k = 1, 2, ..., ly_{imax}$ represent the depth of the un-scheduled RNs on $\mathcal{T}_i^{left}$, where $ly_{imax}$ is the largest depth. $ly_{imax} \leq 2D_i^{left}$ based on Lemma 1 ($iv$).

*Case 1*: $ly_{imax} = 2j + 1$ is odd.

In this case, the transmissions from the dominators with depth $2j + 1$ to their parents which are connectors with depth $2j$ are scheduled first. For any transmission $a \rightarrow \mathcal{P}_i(a)$, it cannot be scheduled together with transmissions whose senders are dominators with depth $2j+1$ in $\mathcal{NCS}(a \rightarrow \mathcal{P}_i(a))$. Since $\mathcal{P}_i(a)$ is a connector, it has at most five neighbors which are dominators including $a$. However, $\mathcal{P}_i(a)$'s parent will not be scheduled together with $a$. Thus, $\|\mathcal{N}_{\mathcal{T}_i}(\mathcal{P}_i(a))\| = 3$. Moreover, $a$'s one-hop neighbors which are connectors or dominators do not have any extra dominator children with depth $2j + 1$. Hence, $\|\mathcal{NCS}(a \rightarrow \mathcal{P}_i(a))\| = 3$, that is, at most $3+1=4$ time slots are needed to finish the transmission scheduling from depth $2j+1$ to depth $2j$.

After that, we consider the scheduling from depth $2j$ to depth $2j-1$, where for any transmission $a \rightarrow \mathcal{P}_i(a)$, $a$ is a connector with depth $2j$ and $\mathcal{P}_i(a)$ is a dominator with depth $2j-1$. Only the transmissions whose senders are connectors with depth $2j$ in $\mathcal{NCS}(a \rightarrow \mathcal{P}_i(a))$ are considered in this turn. Based on Lemma 1 (*ii*), $\mathcal{P}_i(a)$ has at most 20 one-hop neighbors which are connectors including $a$ and $\mathcal{P}_i(a)$'s parent. Hence, $\|\mathcal{NCS}(a \rightarrow \mathcal{P}_i(a))\| = 18$. Similarly, $a$'s one-hop neighbors which are connectors or dominators do not have any extra connector children in depth $2j$. Therefore, at most 19 time slots are needed to finish the transmission scheduling from the $(2j)$-th to the $(2j-1)$-th depth. Specially, the scheduling from depth 2 to depth 1 needs at most 20 time slots due to the fact that the ANs have no parent in $\mathcal{T}_i$.

The scheduling between dominators and connectors rotates until all the data are collected by the AN in depth 1. The latency from depth $2j+1$ to depth 1 is represented by $l_{odd} \leq 20+4+(19+4)(j-1) = 23(ly_{imax}-1)/2+1 \leq 23D_i^{left}-10$.

*Case 2*: $ly_{imax} = 2j$ is even.

The analysis of case 2 is similar to case 1. Note that the transmission latency from depth $2j+1$ to depth $2j$ is removed. Hence, we denote the latency as $l_{even} \leq 20+(19+4)(j-1) = 23(ly_{imax})/2-3 \leq 23D_i^{left}-3$.

In summary, the latency of scheduling $\mathcal{F}^{left}$ denoted by $l_{\mathcal{F}^{left}} = max\{l_{\mathcal{T}_1^{left}}, l_{\mathcal{T}_2^{left}}, ...., l_{\mathcal{T}_m^{left}}\} \leq 20+5\Delta+23\max_{i=1}^{m} D_i^{left}-3 = 23\max_{i=1}^{m} D_i^{left}+5\Delta+17$. □

**Lemma 3.** *The latency of scheduling $\mathcal{S}(\mathcal{O}is)$ is no more than $\max_{i=1}^{m}\{(23D_i+5\Delta+21)k_i\}$.*

*Proof.* According to the definition of $\mathcal{S}(\mathcal{O}is)$ (Section 3.2.3), $\mathcal{S}(\mathcal{O}is)$ is a set of RNs whose interference range overlapped with other RN (or RNs) in other region (or regions) and their descendants. $\mathcal{S}(\mathcal{O}is)$ can be represented by $\{\mathcal{T}_1^{Ois}, \mathcal{T}_2^{Ois}, ..., \mathcal{T}_m^{Ois}\}$. Let $\{k_1, k_2, ..., k_m\}$ represent the maximum OD of each $\mathcal{T}_i^{Ois}$. For each $\mathcal{T}_i^{Ois}$, let $l_{iomin}$ represent the smallest depth of the RNs in $\mathcal{T}_i^{Ois}$ and $l_{iomax}$ represent the largest depth of the RNs in $\mathcal{T}_i^{Ois}$. Since $\mathcal{T}_i^{Ois}$ may not be consecutive, we extend $\mathcal{T}_i^{Ois}$ to a subset consisting of the RNs in $\mathcal{T}_i$ from depth $l_{iomin}$ to depth $l_{iomax}$. Evidently, the latency of $\mathcal{T}_i^{Ois}$ is no more than the latency of extended $\mathcal{T}_i^{Ois}$. It follows that $l_{iomax} - l_{iomin} \leq 2D_i$.

In order to get the latency of scheduling $\mathcal{T}_i^{Ois}$, we first consider the scheduling of $\mathcal{T}_i^{Ois}$ without considering the overlapping. Then, the scheduling of $\mathcal{T}_i^{Ois}$ contains two parts. The first part is the same as the first phase of $T_i^{left}$ and the second part can be divided into the following four cases. The analysis of each case is similar to that of $\mathcal{T}_i^{left}$.

*Case 1*: Both $l_{iomin}$ and $l_{iomax}$ are odd. Then, the latency is $l_{oo} = (4+19)(l_{iomax} - l_{iomin})/2 \leq 23D_i$. Particularly, if the AN is included, then $l_{oo} = 23D_i + 1$.

*Case 2*: $l_{iomin}$ is odd while $l_{iomax}$ is even. Then, the latency is $l_{oe} = (19+4)((l_{iomax} - l_{iomin})/2 - 1) + 19 \leq 23D_i - 4$. Particularly, if the AN is included, then $l_{oe} = 23D_i - 3$.

*Case 3*: $l_{iomin}$ is even while $l_{iomax}$ is odd. Then, the latency is $l_{eo} = (4+19)((l_{iomax} - l_{iomin})/2 - 1) + 4 \leq 23D_i - 19$.

*Case 4*: Both $l_{iomin}$ and $l_{iomax}$ are even. Then, the latency is $l_{ee} = (19+4)(l_{iomax} - l_{iomin})/2 \leq 23D_i$.

In summary, the latency of scheduling the extended $\mathcal{T}_i^{Ois}$ when the $k_i$ overlappings are considered is $l_{\mathcal{T}_i^{Ois}} \leq k_i * (20 + 5\Delta + 23D_i + 1) = (23D_i + 5\Delta + 21)k_i$. Therefore, the latency of scheduling $\mathcal{S}(\mathcal{O}is)$ is upper bounded by $l_{\mathcal{S}(\mathcal{O}is)} = \max_{i=1}^{m}\{(23D_i + 5\Delta + 21)k_i\}$. □

**Lemma 4.** *The latency of scheduling the ANs is no more than* $\sum_{i=1}^{m}(H_i - 1)$.

*Proof.* Since there are $m$ RQ instances, the number of ANs is at most $m$. In the worst case, no two ANs can be scheduled concurrently. Hence, the latency is at most $l_{ANs} = \sum_{i=1}^{m}(H_i - 1)$. One time slot is deducted since the scheduling of the transmission from an AN to its parent has already been considered in the first two phases. □

In practice, the time for ANs to finish their data collecting may vary. The actual latency may be much less than the bound shown in Lemma 4.

**Theorem 1.** *The upper bound of the latency of MRQSA is no more than* $23A + B + C$, *where* $A = \max_{i=1}^{m} D_i^{left}$, $B = \max_{i=1}^{m}\{(23D_i + 5\Delta + 21)k_i\}$, *and* $C = \sum_{i=1}^{m} H_i + 5\Delta - m + 17$.

*Proof.* According to the illustration of MRQSA in Section 3.3.2, the scheduling of $\mathcal{S}(\mathcal{O}is)$ and $\mathcal{F}^{left}$ can overlap in some extent. In the worst case, the scheduling of $\mathcal{S}(\mathcal{O}is)$ and $\mathcal{F}^{left}$ is sequential. Then, the latency is upper bounded by $L = l_{\mathcal{F}^{left}} + l_{\mathcal{S}(Ois)} + l_{ANs} = 23\max_{i=1}^{m} D_i^{left} + \max_{i=1}^{m}\{(23D_i + 5\Delta + 21)k_i\} + \sum_{i=1}^{m} H_i + 5\Delta - m + 17$. $\square$

## 3.4 Performance Evaluation

In this section, we compare the performance of MRQSA with the centralized version of DCQS denoted by C-DCQS proposed in [80]. C-DCQS is the most recently published algorithm that solves the multi-query scheduling problem. We compare the two algorithms in terms of latency under three different scenarios. Since energy consumption is still one of the primary concerns in WSNs, the performance of our proposed MRQSA in terms of energy consumption is evaluated as well.

### 3.4.1 Simulation Environment

In our simulations, we deploy sensor nodes in a monitoring area of $1000m \times 1000m$, where all the sensor nodes have the same interference/transmission range of $50m$. The sink is deployed in the upper left corner. Moreover, the regions in an MRQ are set as squares. During the result collecting procedure of an MRQ, the results from a particular region will be aggregated and the results from different regions cannot be aggregated. The energy consumption model is that receiving and transmitting a packet both consume 1 unit of energy. 1000 units of energy are assigned to each sensor node during the initialization process. Besides, the latency is defined as the time interval between the first RN beginning to transmit data and the last packet of the results being received by the sink. The sensor nodes involved in MRQSA contain RNs and the sensor nodes on the shortest paths from ANs to the sink. To be fair, we make the following improvements to C-DCQS. Firstly, we use the same algorithm in [40] to construct a CDS-based tree as the routing tree for C-DCQS. Secondly, the sensor nodes involved in an MRQ under C-DCQS is defined as a subtree rooted at the sink which has the least number of sensor nodes but contains all the RNs. In this way, the number of sensor nodes involved in an MRQ is minimized under C-DCQS. The simulation is implemented using C++ in Microsoft Visual Studio 2010 development environment. The results shown in the following are the averaged values by executing the same procedure for 100 times.

### 3.4.2 Simulation Results

We examine the latency of C-DCQS and MRQSA when the number of query regions of an MRQ varies as shown in Fig.3.5(a). We fix each query region as a square of $600m \times 600m$ and the regions are randomly generated in the WSN. The number of query regions increases from 2 to 10 by 1. From Fig.3.5(a), we can see that, with the size of an MRQ increasing (the size of an MRQ means the number of regions in this MRQ), the latency of both C-DCQS and MRQSA increases. This is because more data transmissions are needed when collecting the results from new added regions to the sink. It follows that more time is needed to guarantee collision-free scheduling. MRQSA has a better performance because MRQSA tries to maximize the parallelism of scheduling by concurrently scheduling transmissions of different regions. While C-DCQS calculates the minimum time interval for each query in different regions and schedules transmissions in different regions sequentially according to the single fixed scheduling tree. Thus, more time is needed for C-DCQS. MRQSA reduces latency by 49.3% on average compared with C-DCQS.

We study the influence of network density on the latency of C-DCQS and MRQSA as shown in Fig.3.5(b). In this case, the query region is fixed to be a square of $600m \times 600m$ and the number of query regions is 5. We increase the number of sensor nodes in the WSN from 600 to 2000 by 200 each time. Fig.3.5(b) shows that the latency of both C-DCQS and MRQSA increase with the increasing of node density. This is because the number of RNs in each query region increases in a denser network. Therefore, more time is required to finish the scheduling. However, the increasing of MRQSA is more stable than C-DCQS. Especially, when the number of the sensor nodes increases from 1400 to 2000, the latency of C-DCQS increases shapely. This is because for C-DCQS, with the increasing of the network density,

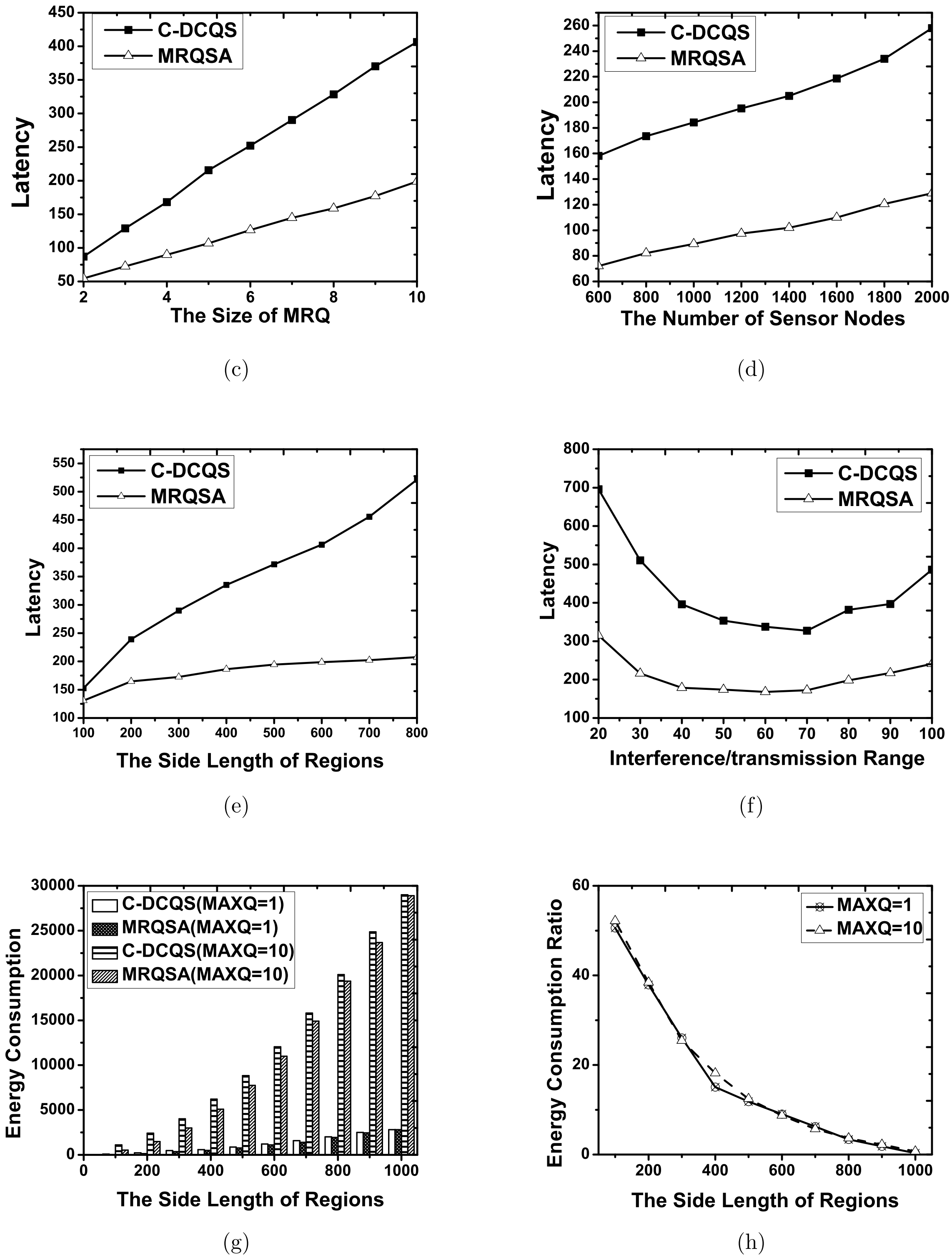



Figure 3.5. Simulation Results.

the intermediate nodes in a fixed single routing tree have to wait longer to collect results from its children. Hence, the minimum interval time required to avoid collisions between the scheduling of different regions increases. MRQSA reduces latency by 50.7% on average when compared with C-DCQS under different network densities.

We study the latency of C-DCQS and MRQSA when changing the size of regions as shown in Fig.3.5(c). In this case, the number of sensor nodes is 1500, the number of query regions is 10, and the size of regions changes from $100m \times 100m$ to $800m \times 800m$. We can see from Fig3.5(c) that with the regions' size increases, the latency of both C-DCQS and MRQSA increase. Especially, the incremental ratio of C-DCQS is higher than MRQSA. This is because with the increasing of region size, the overlapped degree of sensor nodes increases, *i.e.*, more sensor nodes or transmissions are involved in more than one query regions. Hence, in order to guarantee a collision free scheduling for an MRQ, C-DCQS requires longer interval time for scheduling of different regions. By contrast, MRQSA shows its advantage of dealing with an MRQ with overlapping, since it concurrently schedules transmissions in all the regions. Furthermore, the transmissions whose senders are in the overlapped regions are scheduled first which reduces the potential collision possibility. MRQSA reduces the latency by 42.7% on average compared with C-DCQS.

We evaluate the impact of interference/transmission range on the latency of C-DCQS and MRQSA as shown in Fig.3.5(d). In this case, the number of the sensor nodes is 1500, the number of query regions is 10, the size of the query regions is $400m \times 400m$ and the interference/transmission range changes from $20m$ to $100m$. From Fig.3.5(d) we can see that the latency of C-DCQS decreases when the interference/transmission range changes from $20m$ to $70m$ followed by an increase when the interference/transmission range changes from $70m$ to $100m$. The latency of MRQSA shows a similar variation tendency, *i.e.*, the latency of MRQSA decreases when the interference/transmission range changes from $20m$ to $60m$ followed by an increase when the interference/transmission range changes from $60m$ to $100m$. For simplicity, we use the decreasing phase and increasing phase to describe the two phases. Since the in-network data transmission relies on the multi-hop communication among sensor nodes, when the interference/transmission range is small, the delay caused by multi-hop communication plays a main role in the latency. With the increasing of interference/transmission range, although there are more interferences, the delay caused by multi-hop latency drops with a higher ratio. Hence, the total transmission latency decrease as shown in the decreasing phase in Fig.3.5(d). However, when the interference/transmission range exceeds some threshold, such as $70m$ in C-DCQS and $60m$ in MRQSA, the in-network data transmission suffers from interference, which results in the increasing phase. In summary, Fig.3.5(d) shows that MRQSA reduces the latency by 51.63% on average compared with C-DCQS.

In order to study the energy performance of MRQSA, we evaluate the energy consumption of C-DCQS and MRQSA with respect to different sizes of query regions as shown in Fig.3.5(e) and Fig.3.5(f). Fig.3.5(e) shows the total energy cost for all the sensor nodes that participate in a data aggregation process. Fig.3.5(f) shows the energy performance ratio which is evaluated as $\xi = \frac{\overline{CEC} - \overline{CEM}}{\overline{CEC}}$, where $\overline{CEC}$ is the average energy cost of C-DCQS and $\overline{CEM}$ is the average energy cost of MRQSA. To be fair, we only consider the energy consumption during the data aggregation process and ignore the cost of the query dissemination process. In this case, the number of the sensor nodes is 1500, the interference/transmission range is $50m$, 1 and 10 query regions are considered, and the size of the query regions changes from $100m \times 100m$ to $1000m \times 1000m$. From Fig.3.5(e) and Fig.3.5(f) we can see that the performance of MRQSA in terms of energy consumption is better than C-DCQS. Since C-DCQS is based on a global aggregation tree, when the size of query regions is smaller than the monitored target area, more sensor nodes participate in the data aggregation process. Hence, C-DCQS consumes more energy compared with MRQSA. However, with the increasing of the size of query regions, the difference between global aggregation tree and local aggregation tree decreases. Therefore, the difference of energy consumptions of those two methods decreases.

# Chapter 4

# DATA AGGREGATION SCHEDULING IN COGNITIVE RADIO NETWORKS WITH MINIMUM DELAY

## 4.1 Introduction

Even though CRNs have been considered a promising solution to unlicensed spectrum under utilization and licensed spectrum shortage, the opportunistic spectrum accessibility for SUs introduces new challenges which have never been considered in conventional wireless networks. Early studies in CRNs mainly focused on the four most fundamental problems: *spectrum sensing*, *management*, *mobility*, and *sharing*. Subsequently, numerous efforts have been dedicated to studying *network capacity/throughput, network connectivity, routing protocols, etc. Unicast* and *multicast* communication protocols have also been investigated [96][97][53][98][99][100][101][102] [103][104][105][106][107][108]. Another important communication issue, *data aggregation*, has received only limited attention in CRNs.

Due to the scarcity of spectrum opportunity, data aggregation is considered as having a broad prospective in CRNs. Nevertheless, none of the existing works can be intuitively applied to CRNs for many reasons. First, the communication opportunities in CRNs are not symmetric. A potential communication from SU $A$ to SU $B$ does not mean the existence of equivalent opportunity from $B$ to $A$. Most of the current works assume symmetric communication links. Second, PUs in conventional wireless networks only have to consider interference from other PUs. While, a SU has to compete for spectrum resources with both PUs and other SUs in a CRN. To prevent that communications among PUs are interfered, a SU may have to switch among spectrums or terminate its own ongoing transmissions in the worst case. Last, it is very difficult to obtain the overall network information in large scale CRNs because of the opportunistic spectrum availability. Therefore, investigation of distributed data aggregation in CRNs is necessary and meaningful.

The authors of [109] studied a similar problem which focused on convergecast scheduling in CRNs. However, it considered the CRNs with a single licensed spectrum. In this work, a centralized covergecast scheduling algorithm based on cell-based virtual backbone was proposed, but appear that may be not practical in CRNs. Besides, the geometrical partition based virtual backbone reduces the potential parallelism of transmissions among SUs. Furthermore, the SUs considered in [109] is assumed a special distribution which is not suitable for general graphs.

All the above stated reasons and challenges motivate the research of this paper. In this paper, we investigate data aggregation in CRNs under a more general and practical network model. No constraint or assumption is required for the distribution of SUs or PUs. Due to the fact that time sensitivity plays a vital role in many applications, our work mainly focuses on finding a scheduling plan in a distributed manner for data aggregation with the objective of minimizing the total transmission delay. Particularly, we consider a CRN consisting of SUs and PUs with multiple available licensed spectrums. PUs can access the licensed spectrums freely. While the SUs have to conduct a spectrum sensing process before accessing the spectrums. The accessing opportunity exists for a SU *iff* no PU's transmission is interrupted. During a data aggregation process, if a spectrum is available (not being used by PUs), a SU has to decide "to transmit or to wait" considering whether any ongoing transmission will be interfered and also the transmission latency. The contributions of this section can be summarized as follows:

- We formalize the Minimum Latency Data Aggregation Scheduling (MLDAS) problem in CRNs. The formal definitions of SU-PU collision and SU-SU collision are given for the purpose of generating a collision-free scheduling plan.

- Subsequently, the MLDAS problem under the UDG interference model is studied. The idea of using multiple relays to help SUs to forward data is considered in this paper. First, to increase SU's potential transmission opportunity and reduce interference simultaneously, a practical distributed scheduling algorithm based on a CDS routing hierarchy is introduced. After that, for the purpose of maximizing the "gain" and minimizing the “pain”, an improved solution based on a general routing hierarchy is proposed.

- Considering the fading property of wireless signal, the Physical Interference Model (PhIM) is considered in this paper. Based on the conclusion of [57], a distributed data aggregation algorithm for PhIM is introduced.

- In order to evaluate the performance of our proposed methods, extensive simulations were conducted. The simulation results show that both UDAS and PDAS have acceptable delay performance.

## 4.2 System Model and Problem Formulation

### 4.2.1 Network Model

In this paper, we consider a CRN consisting of a dense secondary network and a primary network. The SUs in the secondary network and PUs in the primary network are deployed in the same area and share the same time, space, and spectrums. Assuming the time is slotted with slot length $T$, where $T$ is the time needed for a SU to sense an available spectrum and finish a transmission successfully, but only long enough for a PU to conduct one data transmission.

**Primary network**. In the primary network, we consider $N$ randomly deployed PUs, labeled as $P_1, P_2, ..., P_N$, operating on $K$ orthogonal parallel licensed spectrums, indexed by $\{C_1, C_2, ..., C_K\}$. The transmission radius and interference radius of a PU are denoted as $R$ and $R_I$, respectively. In each time slot $t$, a PU is either *active* or *inactive*. An active PU either sends or receives data (not both) on one of the $K$ spectrums, and an inactive PU keeps silent (neither sending nor receiving data) for $T$. Let $\overrightarrow{\mathcal{P}_i} = \{\mathcal{P}_i^1, \mathcal{P}_i^2, ..., \mathcal{P}_i^K\}$ denote the probability that a PU becoming active, and $\mathcal{P}_i^k$ represents the probability that $P_i$ becomes active on channel $C_k$. Particularly, an active PU can only stay on one of the $K$ spectrums in a given time slot.

**Secondary network**. The secondary network under consideration consists of $n$ randomly deployed SUs, labeled as $S_1, S_2, ..., S_n$, among which there is a SU called base station (SU-BS) denoted as $S_b$ expecting to collect information from the network. Each SU is equipped with a single, half-duplex cognitive radio, with transmission radius $r$ and interference radius $r_I$, respectively. Due to the radio limitation, a SU can either send or receive data, not both, from all directions at one time. Particularly, in each time slot $t$, a SU opportunistically accesses one of the licensed spectrums unoccupied by active PUs in a sensing-before-transmission manner. For two SUs $S_i$ and $S_j$, let vector $\overrightarrow{\mathcal{P}rob(S_iS_j)} = \{p_{ij}^1, p_{ij}^2, ..., p_{ij}^K\}$ represent the channel accessing probability of the transmission from sender $S_i$ to receiver $S_j$, where $p_{ij}^k$ denotes the probability that $S_i$ and $S_j$ simultaneously get the accessing chance of channel $C_k$ at a particular time $t$. $p_{ij}^k$ is initialized to $(\prod\limits_{P_u \in \{P_u \mid \|S_i - P_u\| \leq r_I\}} (1 - \mathcal{P}_u^k)) * (\prod\limits_{P_u \in \{P_u \mid \|S_j - P_u\| \leq R_I\}} (1 - \mathcal{P}_u^k))$ indicating the expected probability that the transmission will not interrupt the primary network. A *potential* data transmission chance between $S_i$ and $S_j$ $(1 \leq i, j \leq n,\ i \neq j)$ at time $t$ on spectrum $C_k$ $(1 \leq k \leq K)$ exists *iff* communication in the primary network will not be interfered.

A *logical link* exists between $S_i$ and $S_j$ *iff* the *Euclidean Distance* between $S_i$ and $S_j$ denoted by $\| S_i - S_j \| \leq r$. Then, the secondary network can be represented as $G = (S, E_l)$, where $S = \{S_1, S_2, ..., S_n\}$ is the set of all the SUs, and $E_l$ is the set of logical links among all the SUs.

The spectrum accessing opportunities for SUs are constrained by the fact that no collision should be caused to the primary network. For the purpose of ultimately utilizing the idle spectrums, we assume *perfect sensing* and *zero false alarm* since the spectrum sensing is out of the scope of this paper. Furthermore, we assume SUs can exchange limited information through a common control channel, so that each SU can check the interference condition separately.

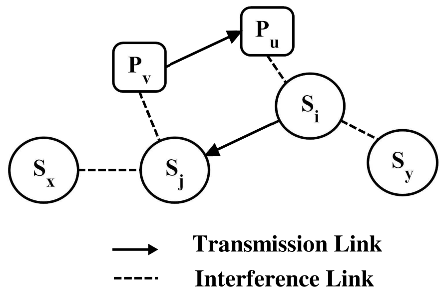



Figure 4.1. An example for SU-SU/SU-PU Collision.

### 4.2.2 Problem Formulation

Let $\mathcal{T}_{ij}^{\{s,k,t\}}$ denote a transmission in the secondary network from sender $S_i$ to receiver $S_j$ ($1 \leq i, j \leq n$ and $i \neq j$) on spectrum $k$ at time $t$, and $\mathcal{T}_{uv}^{\{p,k,t\}}$ ($1 \leq u, v \leq m$ and $u \neq v$) represent a transmission in the primary network from sender $P_u$ to receiver $P_v$ on spectrum $k$ at time $t$.

**Definition 4.2.1.** Secondary User - Primary User ( SU-PU) collision. *Given $\mathcal{T}_{ij}^{\{s,k,t\}}$ and $\mathcal{T}_{uv}^{\{p,k,t\}}$, if $\| S_i - P_v \| \leq r_I$ or $\| S_j - P_u \| \leq R_I$, it is called a SU-PU collision.*

An example is shown in Fig.5.1. At a particular time $t$, if transmission $\mathcal{T}_{vu}^{\{p,k,t\}}$ is ongoing, then transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$ will cause a SU-PU collision because it will interfere $\mathcal{T}_{vu}^{\{p,k,t\}}$. According to the definition of SU-PU collision, a spectrum $C_k$ is said to be *available* to $S_i$ at time $t$ *iff* the data transmission in which $S_i$ is involved (either as a sender or receiver) does not cause any SU-PU collision. Then, a *physical link* on channel $C_k$ from a sender $S_i$ to a receiver $S_j$ at $t$ exists if the following two conditions are satisfied: a logical link exists between $S_i$ and $S_j$, and a channel $C_k$ is available to both $S_i$ and $S_j$. Then, transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$ can be carried out indicating that there is a physical link from sender $S_i$ to receiver $S_j$.

Particularly, referring to the definitions of logical link and physical link, a logical link is symmetric whereas a physical link is not necessarily symmetric. The unsymmetrical and opportunistic accessing property of physical link makes the data aggregation problem in CRNs even more challenging.

**Definition 4.2.2.** Secondary User - Secondary User (SU-SU) collision. *Let $S_i$, $S_j$, $S_x$, and $S_y$ denote four different SUs in the secondary network. Given two transmissions $\mathcal{T}_{ij}^{\{s,k,t\}}$ and $\mathcal{T}_{xy}^{\{s,k,t\}}$, if $\parallel S_i - S_y \parallel \leq r_I$ or $\parallel S_j - S_x \parallel \leq r_I$, then a SU-SU collision occurs.*

Taking Fig.5.1 as an example, given transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$, if $S_x$ also sends data on channel $k$ at $t$, then a SU-SU collision occurs due to $S_x$'s interference to $\mathcal{T}_{ij}^{\{s,k,t\}}$.

Based on the defined network model and aforementioned definitions, the **Minimum Latency Data Aggregation Scheduling** (MLDAS) problem in a CRN can be formalized as follows:

Given a secondary network denoted by $G = (S = \{S_1, S_2, ..., S_n\}, E_l)$, an MLDAS in a CRN can be defined as a set of scheduled links $\mathbb{S} = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_{\mathbb{L}}\}$, where each $\mathbb{S}_t$ $(1 \leq t \leq \mathbb{L})$ is a set of collision-free transmissions among SUs scheduled at $t$. Furthermore, the following constraints exist:

1. $\forall t$ $(1 \leq t \leq \mathbb{L})$, neither SU-PU collision nor SU-SU collision is caused by any transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$ in $\mathbb{S}_t$.
2. $\forall t_1, t_2$ $(1 \leq t_1, t_2 \leq \mathbb{L}$, $t_1 \neq t_2)$, given two transmissions $\mathcal{T}_{ij}^{\{s,k_1,t_1\}}$ and $\mathcal{T}_{xy}^{\{s,k_2,t_2\}}$, $i \neq x$.
3. $\bigcup_{t=1}^{\mathbb{L}} \{S_i | \mathcal{T}_{ij}^{\{s,k,t\}} \in \mathbb{S}_t\} = \{S_1, S_2, ..., S_n\} - S_b$.
4. All data has been aggregated to $S_b$ at $\mathbb{L}$.
5. $\arg\min_{\mathbb{S}=\{\mathbb{S}_1,\mathbb{S}_2,...,\mathbb{S}_{\mathbb{L}}\}} \mathbb{L}$.

Constraint 1 shows that MLDAS should be SU-PU collision-free and SU-SU collision free. SU-PU collision-free guarantees that no interference will be caused to the primary network, and SU-SU collision free avoids extra delay and congestion caused by retransmission. Constraint 2 indicates the property of data aggregation, that is, each SU sends its aggregation result (aggregated data of its own and data received during the MLDAS) only once. The data integrity property is ensured by constraint 3, where all SUs have to transmit data to a designated receiver. Finally, the aggregation result of all the data is received by the

SU-BS according to constraint 4. Constraint 5 indicates that the objective of the MLDAS $\mathbb{S}$ is to minimize the total transmission latency.

The MLDAS problem in CRNs is NP-hard. This conclusion is based on the fact that the minimum latency data aggregation scheduling problem is NP-hard in WSNs [109][39], which can be considered as a special case of MLDAS in CRNs when the number of PUs is fixed to be zero, and the number of channels is $K = 1$. Considering the NP-hardness of the MLDAS problem, two heuristic distributed algorithms under two widely used inference models are presented in this paper.

## 4.3 Data Aggregation Scheduling under the UDG model

In this section, we investigate the MLDAS problem under the *Unit Disk Graph* (UDG) interference model.

### 4.3.1 UDG Interference Model

UDG interference model has been widely employed in the literatures. Under this model, the interference range and transmission range of wireless devices are denoted by equally likely disks. That is, in the UDG model, $R = R_I$ and $r = r_I$.

### 4.3.2 Connected Dominating Set (CDS) based Aggregation Tree

Given a graph $G = \{V, E\}$, where $V$ is the set of vertices and $E$ is the set of edges. A *Dominating Set* (DS) is defined as a subset $V_D \subseteq V$, such that $\forall v \in V$, either $v$ is in $V_D$ or one of $v$'s neighbors is in $V_D$. Let $N(v)$ represent the set of $v$'s neighbors in $G$ ($\forall N(v) \in V \rightarrow (v, N(v)) \in E$). That is, if $V_D$ is a DS, then, $(\forall v \in V) \rightarrow ((v \in V_D) \bigvee \exists N(v)(N(v) \in V_D))$.

The *Connected Dominating Set* (CDS)-based aggregation tree has been used for time sensitive data aggregation in both conventional wireless networks [39] and CRNs [57]. Both centralized and distributed construction algorithms can be found in the existing literatures. Generally speaking, a CDS-based data aggregation tree can be constructed in three steps. In the first step, a *Maximal Independent Set* (MIS), which is also a DS, denoted as $V_D$ is found in $G$. Vertices in $V_D$ are called dominators. Subsequently, another subset $V_C$ known as a set of connectors is chosen from $V - V_D$. The connectors connect dominators and make $G[V_D \bigcup V_C]$ a connected subgraph. The connected subgraph $G[V_D \bigcup V_C]$ is called a CDS. In the last step, for each vertex (named dominatee) in $V - \{V_D \bigcup V_C\}$, some rules are applied to assign a vertex from $V_D$ to be its parent.

The CDS-based aggregation tree (as shown in Fig.4.2(f)) has the following properties: constrained by the characteristics of the tree structure, every vertex only has one parent except for the root. Particularly, dominatee vertices are leaves of the aggregation tree, and every dominatee has a dominator parent. Connectors have dominator parents and all their children are also dominators. A dominator vertex (except the root) has a parent which is a connector and its children are either connectors or dominatees.

### 4.3.3 DA Hierarchy Construction

The CDS-based data aggregation tree shows its advantage in traditional wireless networks. However, both the unsymmetrical link availability and opportunistic link accessibility make the intuitive employment of a CDS-based data aggregation tree in CRNs impractical. On the one hand, if a CDS-based tree is directly used in CRNs, senders and receivers may have to wait for a long time until they both finally get spectrum resources which have not been occupied by extremely active PUs. On the other hand, the property of data aggregation, where a parent node has to wait for data from all of its children, results in an even worse situation when a fixed tree-based virtual backbone is used. Therefore, in order to improve utilization efficiency, a novel CDS-based hierarchy is proposed. In our proposed hierarchy, instead of allocating only one parent to a vertex as a relay node to the upper level, an *aggregation set* is allocated.

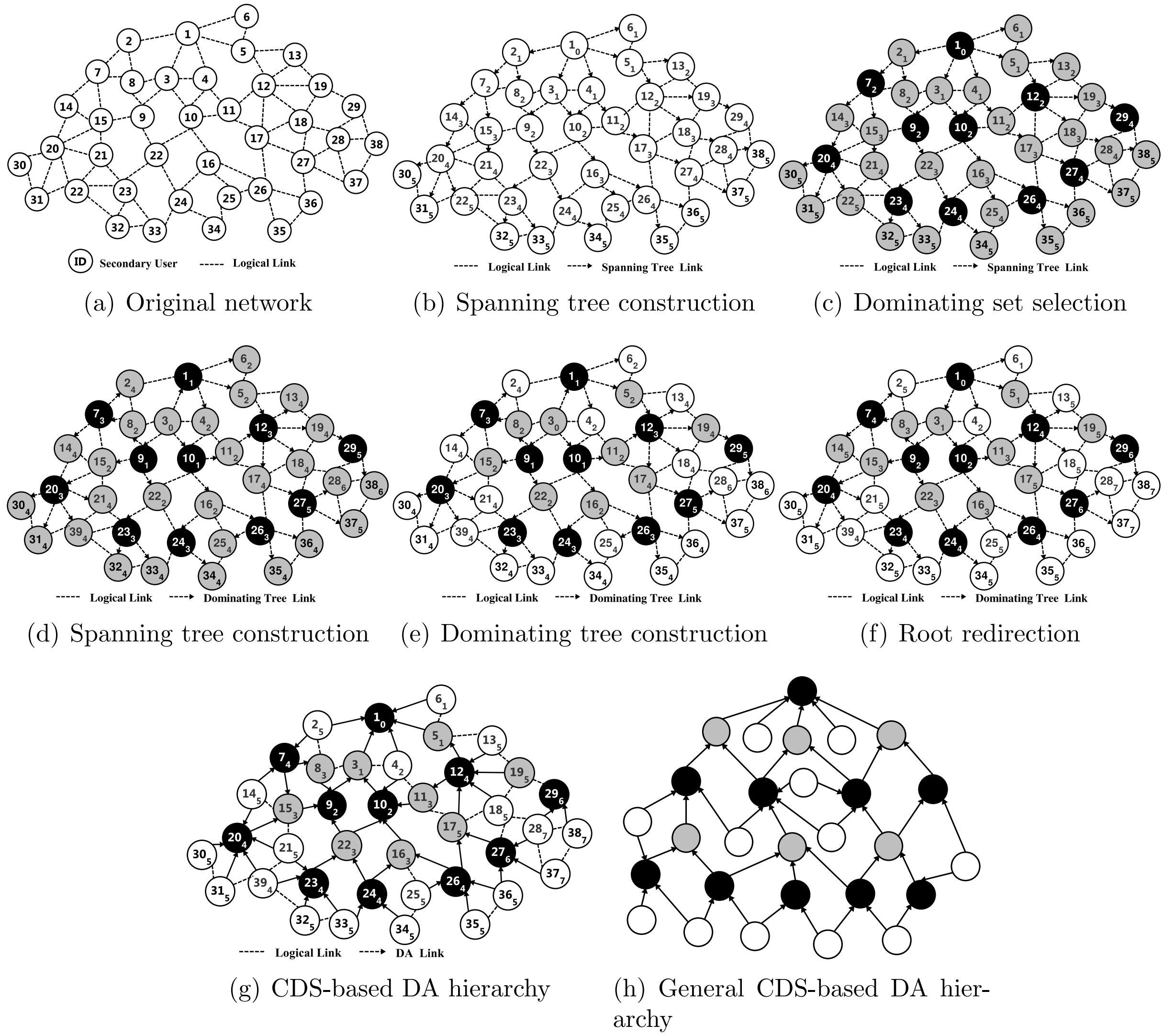
Given a secondary network denoted as $G = (V = \{S_1, S_2, ..., S_n\}\}, E_l)$, among which $S_b$ is the user expecting information from the network, the CDS-based data aggregation hierarchy, whose construction is based on the distributed CDS construction algorithm presented in [110], can be concluded into four steps. During the construction, every SU maintains the following information:

- *level*: initialized to be the number of fewest hops from a SU to $S_b$.
- *color*: SU's role indicator, where white indicates a dominatee, black indicates a dominator, and gray indicates a connector.
- *childrenList* and *neighborList*: to store a SU's children and neighborhood information, respectively.
- Aggregation Set ($\mathcal{A}$): a set of potential parents for a SU. Particularly, $\mathcal{A}(S_i)$ is the aggregation set of $S_i$.

In the first step, every SU gets its LEVEL based on a distributed construction process of a spanning tree $T$ rooted at $S_b$. After that, a DS set $V_D$ is found based on node's ranking (*level* and nodeID) in a distributed way [110]. At the end of this step, every SU in $V_D$ is colored in black and the other nodes in $V - V_D$ are colored in gray. Fig.4.2(a)-4.2(c) show this process, where the number in a circle represents the ID of a SU and the subscript denotes its *level*.

In the second step, let $r_{T^*}$ represent the neighbor of $S_b$ who has the largest number of black neighbors. A dominating tree $T^*$ is generated rooted at $r_{T^*}$ by adding black nodes and gray nodes alternatively. Then, the internal SUs of $T^*$ denoted as $V_D \bigcup V_C$ form a CDS [110] as shown in Fig.4.2(d). However, $T^*$ is not rooted at $S_b$, and the external SUs are still colored in gray which have the same color as SUs in $V_C$. Hence, $T^*$ is not suitable to be directly used as a logical routing structure.

In the third step, external nodes of $T^*$ color themselves white by checking whether their *childrenList* is empty. Subsequently, a message is exchanged between $S_b$ and $r_{T^*}$, so that $S_b$ becomes the root of $T^*$, and $r_{T^*}$ becomes a child of $S_b$. After that, starting from $S_b$, each node updates its *level* information according to the modified $T^*$, where $S_b.level$ is initialized to 0. At the end of this step, a CDS-based $T^*$ rooted at $S_b$ is constructed, including three types of SUs. The dominators in $V_D$ are colored in black and in *even* levels, the connectors




(a) Original network (b) Spanning tree construction (c) Dominating set selection

(d) Spanning tree construction (e) Dominating tree construction (f) Root redirection

(g) CDS-based DA hierarchy (h) General CDS-based DA hierarchy

Figure 4.2. Construction of the CDS-based DA Hierarchy.

in $V_C$ are gray and in *odd* levels, and the white dominatees in $V_E = V - (V_D \bigcup V_C)$ are the leaves of $T^*$. This step is shown in Fig.4.2(e)-4.2(f).

In the last step, each SU updates its neighbors' *color* and *level* information according to its *neighborList*. Subsequently, the aggregation set and *childrenList* of a SU is updated according to the following policies.

- A white SU sets its *childrenList* to be empty and adds all the black SUs with no greater *level* in its neighborhood to its $\mathcal{A}$.

- A black SU adds all the white SUs in its *neighborList* with no smaller *level* and gray SUs with greater *level* to its *childrenList* and updates its $\mathcal{A}$ with all gray SUs having smaller *level* in its *neighborList*.

- A gray SU gets its new *childrenList* by adding all the black SUs with greater *level* on its *neighborList* and generates its $\mathcal{A}$ by including all the black SUs with smaller *level* in its *neighborList*.

For example, as shown in Fig.4.2(g), white node 28 has $\mathcal{A}(28) = \{26, 29\}$, black node 12 has $12.childrenList = \{13, 17, 18, 19\}$ and $\mathcal{A}(12) = \{5, 11\}$, and gray node 22 has $22.childrenList = \{23, 24\}$ and $\mathcal{A}(22) = \{9, 10\}$.

Let $\theta$ denote the maximum number of points whose mutual distance is greater than one in a disk of radius $\mathfrak{R}$ . Then, we have $\theta \leq \lfloor \frac{2\pi}{\sqrt{3}}\mathfrak{R}^2 + \pi\mathfrak{R} \rfloor + 1$ [40]. Based on the selection of $\mathcal{A}$, let $S_i$ and $S_j$ denote different SUs in the secondary network. The following lemma holds.

**Lemma 5.** The following statements are true.

*1.1* $\forall S_i \in V_D - \{S_b\}$, *if* $!\exists S_j((S_j \in V_C) \bigwedge (S_j \in S_i.childrenList))$, *then* $1 \leq |\mathcal{A}(S_i)| \leq 20$, *otherwise,* $1 \leq |\mathcal{A}(S_i)| \leq 19$, *where* $|\mathcal{A}(S_i)|$ *denotes the cardinality of* $\mathcal{A}(S_i)$. *Intuitively,* $|\mathcal{A}(S_b)| = 0$.

*1.2* $\forall S_i \in V_C$, $1 \leq |\mathcal{A}(S_i)| \leq 4$. *Particularly,* $|\mathcal{A}(S_i)| = 1$ *if* $S_i \in S_b.childrenList$.

*1.3* $\forall S_i \in V_E$ *where* $V_E = V - \{V_D \bigcup V_C\}$ *is the set of leaves in the tree,* $1 \leq |\mathcal{A}(S_i)| \leq 5$.

*1.4 Let* $L_{max} = \{S_i.level | \forall S_j \in S, S_i.level \geq S_j.level\}$, *then* $L_{max} \leq 2D$. *D is the diameter of the network which is defined as the maximum hop distance between two SUs in a secondary network [76].*

*Proof*: According to the geometric property of $\theta$, where there are at most five points with mutual distance greater than 1 in a UDG with radius 1, and 21 such points in a UDG with radius 2. For a dominator $S_i \in V_D - \{S_b\}$ satisfying the constraint that $!\exists S_j((S_j \in V_C) \bigwedge (S_j \in S_i.childrenList))$ (has no child who is a connector), it has at least 1 connector parent. Since a connector connects two dominator, and there are at most 20 dominators in a UDG with radius 2 except for $S_i$, $S_i$ has at most 20 connector parents. That is, $1 \leq |\mathcal{A}(S_i)| \leq 20$. Otherwise, if $S_i$ has at least one connector in its *childrenList*, we have $1 \leq |\mathcal{A}(S_i)| \leq 19$. That finishes the proof of Lemma 12.1.

For a connector $S_i$, its aggregation set contains all the dominators in its neighborhood with smaller level. To be a connector, it must have at least one dominator in its *childrenList*. Therefore, if $S_i$ is not a child of $S_b$, it must have at least one parent who is a dominator and at most 4 dominators according to the geometric property. Otherwise, if $S_i$ is a child of $S_b$, then it has only one parent $S_b$. That is, $\forall S_i \in V_C$, $1 \leq |\mathcal{A}(S_i)| \leq 4$. Particularly, $|\mathcal{A}(S_i)| = 1$ if $S_i \in S_b.childrenList$, which proves Lemma 12.2.

The conclusion of Lemma 12.3 can be directly derived from the geometric property of $\theta$. For each dominatee, it has at least one dominator parent and at most five dominators in its neighborhood. Therefore, we have $\forall S_i \in V_E$ where $V_E = V - \{V_D \bigcup V_C\}$ is the set of leaves in the tree, $1 \leq |\mathcal{A}(S_i)| \leq 5$. □

A CDS-based DA hierarchy as shown in Fig.4.2(h) can be used as the logical routing structure in our data aggregation process. It is not a CDS-based tree because every node may have a set of parents, logically. The multiple choices of parents improve the potential spectrum accessibility for a SU. This hierarchy ensures that a child sender seeks for a transmission chance among a set of parents, instead of waiting for an opportunity from a specific one. Furthermore, the properties of CDS naturally limit the maximum size of the aggregation set. Since the bigger the aggregation set the more complicated the interference issue, more nodes have to wait for a child which may also introduce extra delay. The CDS-based DA hierarchy tries to balance the spectrum availability and waiting delay.

### 4.3.4 UDSA Scheduling

When a sender $S_i$ has data to send to a receiver $S_j$, in order to avoid collisions, $S_i$ should tell whether the transmission interferes with the ongoing transmissions among PUs or SUs or not. In this subsection, we propose a distributed scheduling algorithm UDSA, where each SU makes a decision only based on its local collected information.

**Definition 4.3.1.** SU-PU Collision Set *(SPCS). Given a transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$, its SPCS is defined as* $\mathcal{SPC}(\mathcal{T}_{ij}^{\{s,k,t\}}) = \{P_u | (\mathcal{T}_{vu}^{\{p,k,t\}} \bigwedge (\| S_i - P_u \| \leq r_I)) \bigvee (\mathcal{T}_{uv}^{\{p,k,t\}} \bigwedge (\| S_j - P_u \| \leq R_I)\}$, *which is the set of active primary users on spectrum $k$ at time $t$ that will be interfered by $\mathcal{T}_{ij}^{\{s,k,t\}}$.*

The SPCS of $\mathcal{T}_{ij}^{\{s,k,t\}}$ involves all the active PUs conflicting with $\mathcal{T}_{ij}^{\{s,k,t\}}$ in the primary network. As shown in Fig.5.1, for transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$, if we have $\mathcal{T}_{vu}^{\{p,k,t\}}$, then $SPCS(\mathcal{T}_{ij}^{\{s,k,t\}}) = \{P_u, P_v\}$.

**Definition 4.3.2.** SU-SU Collision Set *(SSCS). Given a transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$, its SSCS is defined as* $\mathcal{SSC}(\mathcal{T}_{ij}^{\{s,k,t\}}) = \{S_x | (\mathcal{T}_{yx}^{\{s,k,t\}} \bigwedge (\| S_i - S_x \| \leq r_I)) \bigvee (\mathcal{T}_{uv}^{\{s,k,t\}} \bigwedge (\| S_j - S_x \| \leq r_I)\}$. *$\mathcal{SSC}(\mathcal{T}_{ij}^{\{s,k,t\}})$ involves the set of SUs which will conflict with transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$ if they are scheduled on spectrum $k$ at time $t$. Particularly, according to the data aggregation hierarchy constructed in Section 4.3.3, given a transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$, where $S_j \in \mathcal{A}(S_i)$,* $\mathcal{SSC}(\mathcal{T}_{ij}^{\{s,k,t\}}) = \{S_x | (\mathcal{T}_{xy}^{\{s,k,t\}} \bigwedge (S_x \in S_j.neighborList)) \bigvee (\mathcal{T}_{xy}^{\{s,k,t\}} \bigwedge S_x \in N(S_i).childrenList)\}$.

*Taking the CDS-based hierarchy constructed in Fig.4.2(g) as an example, given transmission $\mathcal{T}_{22,10}^{\{s,k,t\}}$, it will conflict with $\{\mathcal{T}_{3,1}^{\{s,k,t\}}, \mathcal{T}_{4,10}^{\{s,k,t\}}, \mathcal{T}_{11,10}^{\{s,k,t\}}, \mathcal{T}_{16,10}^{\{s,k,t\}}, \mathcal{T}_{21,23}^{\{s,k,t\}}, \mathcal{T}_{39,23}^{\{s,k,t\}}, \mathcal{T}_{32,23}^{\{s,k,t\}}$, $\mathcal{T}_{33,23}^{\{s,k,t\}}$, $\mathcal{T}_{33,24}^{\{s,k,t\}}, \mathcal{T}_{34,24}^{\{s,k,t\}}\}$ if any of those transmissions and $\mathcal{T}_{22,10}^{\{s,k,t\}}$ are scheduled together. Therefore,* $SSCS(\mathcal{T}_{22,10}^{\{s,k,t\}}) = \{3, 4, 11, 16, 21, 32, 33, 34, 39\}$.

The distributed scheduling algorithm UDSA is presented in Algorithm 3. In UDSA, each node $S_i$ first checks if its *childrenList* is empty, that is, whether all of its children have finished data transmission (Line 3). If $S_i.childrenList = \emptyset$, $S_i$ is *ready* to send. Otherwise, $S_i$ has to wait and collect "FINISH" messages from $S_i.childrenList$ (Lines 23). When $S_i$ is ready, a spectrum sensing and selection process will be activated (Lines 4-20). During the spectrum sensing process, $S_i$ tries to find transmission opportunity with all the nodes in $\mathcal{A}(S_i)$ on all the $K$ spectrums. Then, a $ParentSelection()$ process will get the "best" parent $S_j$ and the best communication spectrum $C_{spec}$ dynamically from $\mathcal{A}(S_i)$ (Lines 6-7).

**Algorithm 3:** UDSA Scheduling

**input** : $G = (S = \{S_1, S_2, ..., S_n\}, E_t)$, $S_b \in S$

**output**: schedule $\mathbb{S} = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_{\mathbb{L}}\}$

$spec = \emptyset$, $sprob = 1$, $flag = 0$, $t = 1$;

**for** *each* $S_i : S_i \in \{S_1, ..., S_n\} - \{S_b\}$ **do**

    **if** $S_i.childrenList = \emptyset$ **then**

        **for** $i : 1 \rightarrow K$ **do**

            $SpectrumSensing()$;

            **if** $C_i.availiable$ **then**

                $S_j = ParentSelection()$;

                **if** $\mathcal{SPC}(\mathcal{T}_{ij}^{\{s,spec,t\}}) = \emptyset$ *and* $\mathcal{SSC}(\mathcal{T}_{ij}^{\{s,spec,t\}}) = \emptyset$ *and* $Priority_{MAX}$ *and* $p_{ij}^k < sprob$ **then**

                    $spec = C_i$;

                    $sprob = p_{ij}^k$;

                    $flag = 1$;

        **if** $!flag$ **then**

            $t++$;

            Update $\overrightarrow{\mathcal{P}rob(S_iS_j)}$;

            **go to** 4;

        **else**

            Carry out transmission $\mathcal{T}_{ij}^{\{s,spec,t\}}$;

            Send FINISH to $\mathcal{A}$;

            $\mathbb{S}_t = \mathbb{S}_t \bigcup \mathcal{T}_{ij}^{\{s,spec,t\}}$;

            $S_i.sleep()$;

    **else**

        $t++$;

        Wait for FINISH from $S_i.childrenList$;

        **if** *Received FINISH from* $S_c$, $S_c \in S_i.childrenList$ **then**

            $S_i.childrenList = S_i.childrenList - S_{ch}$;

            **go to** 3;

        **else**

            **if** *no FINISH from* $S_c$, $S_c \in S_i.childrenList$ *for* $T$ **then**

                **go to** 23;

The policy for best parent selection is as following: if more than one node in $\mathcal{A}(S_i)$ has the opportunity to receive data, the one with the fewest children and worst transmission opportunity, that is, shortest *cheildrenList* and lowest $p_{ij}^k$, will be chosen. The purpose for choosing the channel which is the most difficult to get is to leave other spectrums which are easier to get to other SUs. This strategy may also reduce potential SU-SU collisions. Subsequently, a further "conflicting free" verification step will be carried out by checking $\mathcal{SPC}(\mathcal{T}_{ij}^{\{s,k,t\}}) = \emptyset$ and $\mathcal{SSC}(\mathcal{T}_{ij}^{\{s,k,t\}}) = \emptyset$ (Line 8). If both collision sets are empty, $S_j$ is considered as the candidate parent and channel $C_{spec}$ is the candidate channel. (Lines 8-11). After passing all the condition checks, transmission $\mathcal{T}_{ij}^{\{s,spec,t\}}$ can be carried out without conflicting with PUs or SUs (Line 17). Subsequently, $S_i$ broadcasts a "FINISH" message to all the nodes in $\mathcal{A}(S_i)$ to inform them no longer have to wait for it, and then $S_i$ goes to sleep since it already finished its transmission (Line 18-20). Otherwise, $S_i$ has to go back and restart spectrum sensing due to the dynamic spectrum opportunity in CRNs (Lines 13-15).

In UDSA, a spectrum sensing and selection process will be triggered only when $S_i$ is ready. In this way, the energy consumption and collision caused by redundant sensing will be reduced. Besides, since spectrum is the most precious resource in a CRN, a channel which is hard to access for one node may also be difficult for its adjacent nodes to access according to their spatial correlation. Therefore, if multiple channels are available for a transmission, the channel which is the most challenging to be obtained will be chosen, so that others can have higher accessing probability. This strategy may also reduce potential SU-SU collisions. Furthermore, once there is a contention between two transmissions, priority will be given to a transmission according to the following order: sender colored in white, gray then black; sender with larger level; sender/receiver with lower channel accessing probability; receiver with longer *childrenList*; or sender with larger ID ($1 \leq ID \leq n$).

## 4.4 MLDAS under the Physical Interference Model

Since wireless signals can be easily affected by the environment and fade during data transmission, the Physical Interference Model (PhIM) has been considered as another important model in CRNs. In this section, we discuss the MLDAS problem under the PhIM.

### 4.4.1 Physical Interference Model (PhIM)

In the PhIM, the quality of signal received by a receiver is measured by the Signal to Interference and Noise Ratio (SINR) - the ratio of the expected signal strength over the total unwanted (including interference from other users and noise from the environment) signal strength. For simplicity, we ignore the influence of the environment and consider the Signal to Interference Ratio (SIR) instead, which is the ratio of the expected signal strength over the total wireless interference signal strength. Under the PhIM with SIR, a receiver can receive data successfully only if the SIR ratio of the transmission is greater than a threshold. Particularly, according to the network model specified in this paper, the SIR constraints for $P_u$ of transmission $\mathcal{T}_{uv}^{\{p,k,t\}}$ and $S_i$ of transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$ are shown by InEq.4.1 and InEq.4.2, respectively.

$$\frac{\mathbb{P}_p * \parallel P_u P_v \parallel^{-\alpha}}{\sum\limits_{\forall P_x \in \mathcal{T}_{xy}^{\{p,k,t\}}} \mathbb{P}_p * \parallel P_x P_v \parallel^{-\alpha} + \sum\limits_{\forall S_x \in \mathcal{T}_{xy}^{\{s,k,t\}}} \mathbb{P}_s * \parallel S_x P_v \parallel^{-\alpha}} \geq \tau_p. \tag{4.1}$$

$$\frac{\mathbb{P}_s * \parallel S_i S_j \parallel^{-\alpha}}{\sum\limits_{\forall S_x \in \mathcal{T}_{xy}^{\{s,k,t\}}} \mathbb{P}_s * \parallel S_x S_j \parallel^{-\alpha} + \sum\limits_{\forall P_x \in \mathcal{T}_{xy}^{\{p,k,t\}}} \mathbb{P}_p * \parallel P_x S_j \parallel^{-\alpha}} \geq \tau_s. \tag{4.2}$$

where constant $\alpha > 2$ is the path loss exponent. $\mathbb{P}_p$ and $\mathbb{P}_s$ are the transmission power of PUs and SUs, respectively. $\mathcal{T}_{xy}^{\{p,k,t\}}$ (respectively, $\mathcal{T}_{xy}^{\{s,k,t\}}$) is the set of transmissions in the primary network (respectively, secondary network) on channel $k$ at $t$. $\tau_p$ and $\tau_s$ are the respective SIR threshold for a PU and a SU.

According to InEq.4.1 and InEq.4.2, transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$ can be protected under the PhIM *iff* the SIR of $S_i$ satisfies Eq.4.2. Meanwhile, $\mathcal{T}_{ij}^{\{s,k,t\}}$ will not ruin the SIR requirement of all the ongoing transmissions in both the primary and secondary networks.

### 4.4.2 PDSA scheduling

In CRNs, the transmissions among PUs have absolute priorities and are not allowed to be interrupted by SUs. Similarly, the ongoing transmissions among SUs should not be interfered either. Last but not the least, interference from ongoing transmissions should be limited so that the receiver can get the information correctly. Therefore, a SU with data to send needs to perform a carrier-sensing action before a data transmission. A transmission will be carried out only if the increased signal strength from the transmission will not affect the ongoing transmissions (among PUs or SUs), and the noise from ongoing transmissions is limited. On the other hand, the carrier-sensing range is expected to be as small as possible, so that no transmission opportunity will be missed due to over-estimation. Therefore, a proper carrier sensing range which can help with avoiding interference and making the best use of spectrum opportunity is desired. The *Proper Carrier sensing Range* (PCR) $R_s$ derived from [57], which satisfies the requirements as proved, is used in this paper. Based on [57], given a SU $S_i$ with data to send, the concept defined below specifies the collision set of $S_i$, which are the senders that will conflict with $S_i$ if they are scheduled on the same spectrum within the same time slot.

**Definition 4.4.1.** SIR Collision Set *(SIRCS). Given a potential transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$, its SIRCS is defined as $\mathcal{SCS}(\mathcal{T}_{ij}^{\{s,k,t\}}) = \{\{P_u, S_x\}|\mathcal{T}_{uv}^{\{p,k,t\}}, \parallel S_iP_u \parallel \leq R_s$ or $\mathcal{T}_{xy}^{\{s,k,t\}}, \parallel S_iS_x \parallel \leq R_s\}$, where $R_s = max\{(1+\sqrt[\alpha]{\frac{c_2\tau_p}{c_1}})*R, (1+\sqrt[\alpha]{\frac{c_2\tau_s}{c_3}})*r\}$ [57] is the PCSR of $S_i$, $c_1 = \frac{\mathbb{P}_p}{max(\mathbb{P}_p,\mathbb{P}_s)}$, $c_2 = 6+6(\frac{\sqrt{3}}{2})^{-\alpha}(\frac{1}{\alpha-2}-1)$, and $c_3 = \frac{\mathbb{P}_s}{max(\mathbb{P}_p,\mathbb{P}_s)}$ .*

According to Def.4.4.1, a transmission $\mathcal{T}_{ij}^{\{s,k,t\}}$ can be scheduled if no sender of the ongoing transmissions (either a PU or SU) is within $S_i$'s PCR. Then we can get a data aggregation scheduling plan in a distributed manner under the PhIM (denoted as PDAS for simplicity) by modifying our UDSA accordingly. Instead of checking $\mathcal{SPC}(\mathcal{T}_{ij}^{\{s,k,t\}})$ and $\mathcal{SSC}(\mathcal{T}_{ij}^{\{s,k,t\}})$, in PDSA, $\mathcal{SCS}(\mathcal{T}_{ij}^{\{s,k,t\}})$ will be checked. Furthermore, since a spectrum sensing will be conducted first, only interference coming from SUs needs to be verified when a spectrum is available. Thus, we consider $\mathcal{SCS}_s(\mathcal{T}_{ij}^{\{s,k,t\}}) = \{S_x|\mathcal{T}_{xy}^{\{s,k,t\}}, \parallel S_iS_x \parallel \leq R_s$ instead of $\mathcal{SCS}(\mathcal{T}_{ij}^{\{s,k,t\}})$. Then we can get the PDSA by replacing Line 8 of Algorithm 3 by $\mathcal{SCS}_s(\mathcal{T}_{ij}^{\{s,k,t\}}) = \emptyset$.

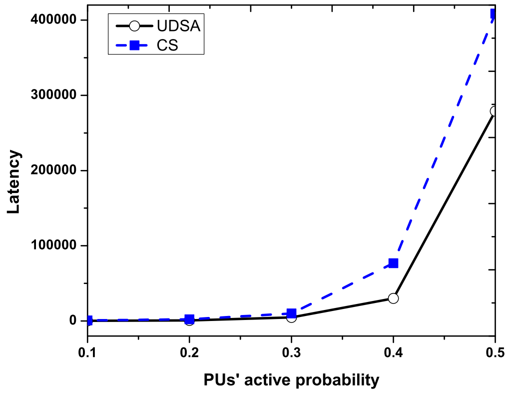
## 4.5 Performance Evaluation

In this section, we examine the performance of our proposed solution through simulations under two scenarios. In the first scenario, time efficiency of our proposed UDSA is evaluated under different spectrum conditions and network configurations. Subsequently, the performance of PDAS is investigated.

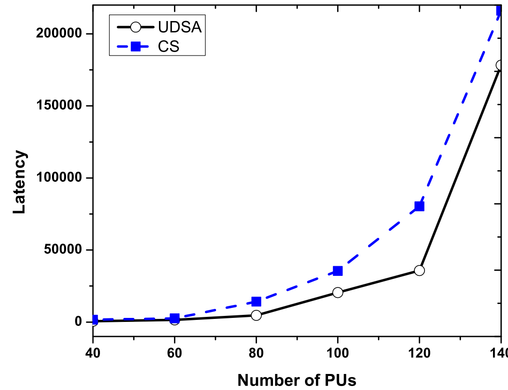
### 4.5.1 Performance of UDSA

We investigate the performance of UDSA under the single spectrum scenario and multiple spectrum scenario, separately. With a single spectrum, a SU seeks a transmission opportunity on a single spectrum. In the multiple spectrum scenario, a SU searches for a chance on multiple spectrums as long as they are not occupied.

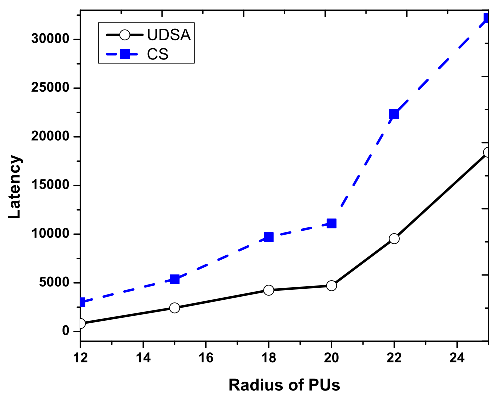
**UDSA vs. CS with Single Spectrum** We evaluate the performance of UDSA by comparing it with CS [109] which is the most state-of-the-art convergecast scheduling algorithm for CRNs. In CS, the area is divided into different cells. Furthermore, SUs in different cells are classified into different levels and concurrent sets. In order to avoid interference, SUs in CS are scheduled from higher level to lower level, and in each level, SUs are scheduled from concurrent set with smaller ID to concurrent set with larger ID. In the simulations, we consider a CRN with a primary network in a region of $100m * 100m$. For clarity, the parameters are consistent with Section 4.3. The numbers of PUs and SUs are denoted as $N$ and $n$, respectively. $R$ represents the radius of a PU and $r$ denotes the radius of a SU. A PU is assumed to be active with probability $p_a$ or keep silent with probability $1 - p_a$ in each time slot. Since only a single spectrum is considered in CS, to be fair, we perform the performance evaluation with only one available spectrum as shown in Fig.4.3(a)-4.3(e), followed by the multi-spectrum performance examination as shown in Fig.4.3(f). According to Fig.4.3, UDSA has a better performance than CS in all the scenarios.



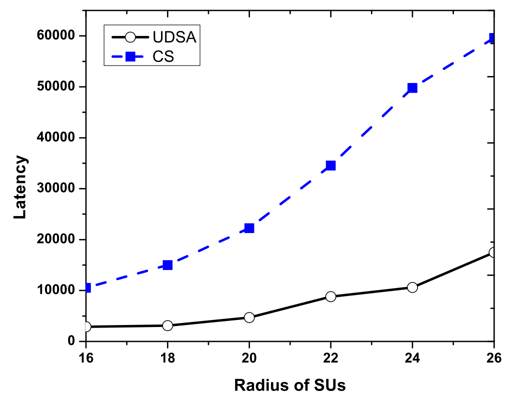

(a) Impact of PUs' activity



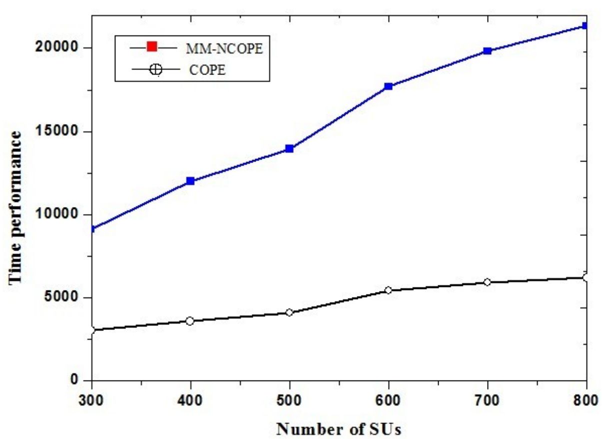

(b) Impact of PU's population



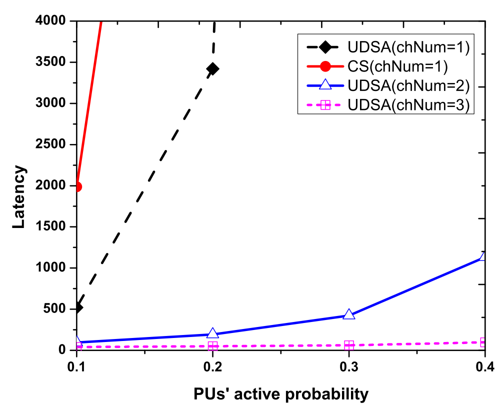

(c) Impact of PU's Interference Range




(d) Impact of SU's Interference Range




(e) Impact of SU's population




(f) Impact of Multi-Spectrums

Figure 4.3. Performance of UDSA with Single Spectrum.

Particularly, the multi-spectrum scenario shows its good performance on data aggregation in CRNs. The simulations are implemented based on the Microsoft Visual Studio 2010 development platform. The data shown in the figures are the averaged results over 100 iterations.

We evaluate the impact of PUs' activities on UDSA and CS by varying $p_a$ from 0.1 to 0.5. The results are shown in Fig.4.3(a). We consider a CRN with $n = 400$ and $N = 80$. The interference radius for both PUs and SUs is fixed to $R = r = 20m$. We can see that the time performance of both the algorithms are closely related to PUs' activities. With the increase of PUs' activities, the scheduling time needed for both the algorithms goes up in order to avoid disturbing transmissions in the primary network. However, the amount of time needed for UDSA is 58.8% less compared with CS. It is because the concurrent sets are scheduled in a fixed order in CS. Even though a spectrum opportunity exists, some SUs may still have to wait if it is not their turns. Therefore, the more active the PUs, the longer time needed for SUs in later concurrent sets to wait for the scheduling of SUs in the former concurrent sets.

The performance of UDSA with varying the number of PUs is examined as shown in Fig.4.3(b). $N$ is from 40 to 140, and $n$ is fixed to be 400. Furthermore, $p_a$ is fixed to 0.3, and both $R$ and $r$ are fixed to $20m$. As shown in Fig.4.3(b), the latency of both UDSA and CS increases with the increasing of the number of PUs. With the increasing of the total number of PUs, the number of active PUs in the primary network grows which results in more activities in the primary network. Hence, in order to avoid conflictions, SUs have to wait for a longer time. Nonetheless, since UDSA seeks transmission opportunity among all its candidate parents, the influence of active PUs' is less in UDSA than in CS. The latency of UDSA is 48.4% less than that of CS in this scenario.

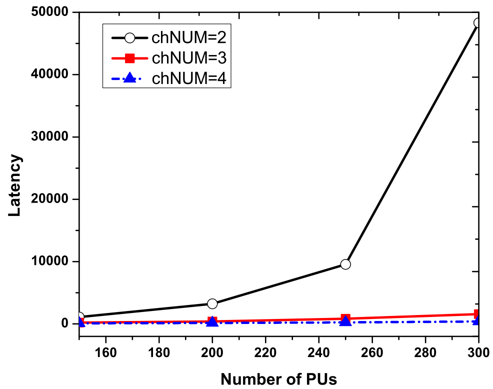
We study the performance of UDSA and CS with respect to the interference radius of PUs. In this case, the parameters are $N = 80$, $n = 400$, $R = 20m$, $p_a = 0.3$, and the radius of a PU varies from $12m$ to $25m$. From Fig.4.3(c) we can see that the delay of both the algorithms decreases with the increasing of the interference radius of PUs. The reason is that the increasing interference radius of active PUs causes interference within a larger area. In order to keep the primary network undisturbed, more SUs have to wait in each time slot. However, even though the reduced concurrency leads to the increasing of latency, UDSA outperforms CS by 57% because of its better concurrency.

We evaluate the influence of SUs' interference radius on the latency of UDSA and CS as shown in Fig.4.3(d). In this case, we adjust the interference radius of SUs from $300m$ to $800m$, and the other parameters are $n = 80$, $N = 400$, $p_a = 0.3$, and $R = r = 20m$. From the results we can see that the increasing of SUs' interference radius leads to an increasing of latency for both UDSA and CS. It is because the competition among SUs becomes more intense with a bigger interference radius. Hence, besides avoiding collisions with PUs, SUs also have to wait for other SUs with higher priorities in its interference range. The bigger the interference radius, the higher chance a SU has to wait. Particularly, the strict order in CS aggravates the negative effect resulting in a 72.8% more latency compared with UDSA.

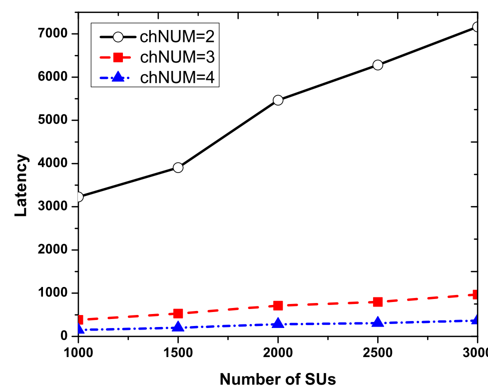
Subsequently, the impact of the number of SUs is presented in Fig.4.3(e). In this scenario, the number of SUs is varied from 300 to 800. The other parameters are $n = 80$, $N = 400$, $p_a = 0.3$, and $R = r = 20m$, which are the same as the former case. The results in Fig.4.3(e) indicate that both UDSA and CS have good performance with respect to scalability. However, due to the fact that the increasing interference among SUs causes more delay in CS, similar to the former scenario, the increasing number of SUs has a worse impact on CS compared with UDSA, that is, cs has 69.3% more latency.

**Performance of UDSA with Multiple Spectrums** When data aggregation is investigated in wireless networks, collision is one of the main factors determining the latency. In this section, we investigate how multiple spectrums affect the performance of UDSA. Particularly, each PU becomes active in the current slot with probability $p_a$, and for an active PU, it has an equal chance to be active on one of the spectrums.

Fig.4.3(f), whose network setting is the same as Fig.4.3(a), shows the delay of UDSA with 2 spectrums and 4 spectrums compared with the performance of UDSA and CS with




(a) Impact of PUs




(b) Impact of SUs

Figure 4.4. Performance of UDSA with Multiple Spectrums.

a single spectrum. From the results we can see that multiple spectrums can significantly reduce the latency of data aggregation in CRNs. When multiple spectrums are available, PUs have more choices when they want to process data transmissions. Hence, the number of active PUs in a specific spectrum is relatively reduced, resulting in the reduction of SU-PU collisions. On the other hand, with PUs' flexible spectrum occupation and proper spectrum accessing policy, SUs have a higher chance to access different spectrums. It leads to the reduction of SU-SU collisions. As a result, the reduction of SU-PU collisions and SU-SU collisions contributes to the improvement of latency.

Furthermore, we examine the influence of PUs and SUs on the multi-spectrum UDSA as shown in Fig.4.4(a) and Fig.4.4(b), respectively. We assume the network is within an area of $100m * 100m$, $R = r = 20m$ and $p_a = 0.3$ in both scenarios. In Fig.4.4(a), $n = 800$ while $N$ varies from 150 to 300. In Fig.4.4(b), $N$ is fixed as 120 while $n$ changes from 1000 to 3000. The results show that the increasing of the number of SUs and PUs results in longer latency, which is accordant with the results shown in Fig.4.3(b) and Fig.4.3(e). Particularly, SU-PU collisions have heavier negative influence compared with SU-SU collisions.

### 4.5.2 Performance of PDSA

In this subsection, we investigate the performance of PDSA under different network scenarios. For simplicity, the parameters are consistent with Section 4.4. $N$ and $n$ denote the numbers of PUs and SUs, respectively, the power of a PU (SU, respectively) is $\mathbb{P}_p$ ($\mathbb{P}_s$, respectively), $\tau_p$ and $\tau_s$ are the respective SIR thresholds for a PU and a SU, and $p_a$ is the probability that a PU becomes active in a time slot. To be specific, in the following scenarios, we consider the CRN and the primary network in an area of $100 * 100m$.

Table 4.1 shows how the number and power of PUs affect the performance of PDSA. In this scenario, we vary the number of PUs from 60 to 140 and the power of PUs from 8 to 16. The other parameters are $n = 400$, $\mathbb{P}_s = 10$, $\tau_p = \tau_s = 8$, $p_a = 0.3$, and $\alpha = 4$. From the results we can see that both the increasing of the number and power of PUs result in the decreasing of delay. The reason can be derived from Eq.4.2. According to Eq.4.2, the SIR for a particular sender is inversely proportional to both the number and the power of PUs. The more PUs or the higher PUs' power, the weaker the SIR received by the sender. Therefore, with the increasing of the number of PUs and PUs' power, a SU has less chance to carry out transmissions. This is also compliant with the constraint that the increasing PU's activities cannot be disturbed.

Table 4.1. Impact of $\mathbb{P}_p$ and $N$

| $N$ | 60 | 80 | 100 | 120 | 140 |
|---|---|---|---|---|---|
| Time | 5486 | 26739 | 99286 | 596201 | 4410070 |

| $\mathbb{P}_p$ | 8 | 10 | 12 | 14 | 16 |
|---|---|---|---|---|---|
| Time | 38261 | 22045 | 37353 | 56404 | 76227 |

Table 4.2. Impact of $\mathbb{P}_s$ and $n$

| $n$ | 300 | 400 | 500 | 600 | 700 | 800 |
|---|---|---|---|---|---|---|
| Time | 19523 | 26059 | 30407 | 32317 | 34246 | 42118 |

| $\mathbb{P}_s$ | 8 | 10 | 12 | 14 | 16 |
|---|---|---|---|---|---|
| Time | 40197 | 27823 | 38442 | 43047 | 79164 |

Table 4.3. Impact of $\alpha$ and $p_{active}$

| $\alpha$ | 3.8 | 3.9 | 4.0 | 4.1 | 4.2 |
|---|---|---|---|---|---|
| Time | 84715 | 47576 | 25820 | 11900 | 3356 |

| $p_{active}$ | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 |
|---|---|---|---|---|---|
| Time | 446 | 2362 | 26168 | 346034 | 5776267 |

We evaluate the performance of PDSA with respect to the number and power of SUs as shown in Table 4.2. In this case, the number of SUs is varied from 300 to 800 and the power varied from 8 to 16. $N = 80$, $\mathbb{P}_p$ is fixed to 8, and the other parameters are the same as the former scenario. The results in Table 4.2 reveal that the latency increases with the growth of the number of SUs, which is because of the increased interference caused by the increased number of SUs. We can derive this based on Eq.4.2, which shows the SIR for a particular sender in both the CRN and the primary network inversely proportional to the number of SUs. To ensure successful data transmissions in the primary network and avoid collisions with other SUs, a SU has to wait for a longer time. However, when it comes to the influence of SUs' power, the conclusion is not intuitive. Table 4.2 indicates that with the increasing of the power of SUs, the latency first decreases and then increases. From Eq.4.2 we can see that for a particular sender, power increasing will enhance its SIR. However, at the same time, power increasing of the other senders in the CRN will weaken it. Furthermore, the increasing power also affects the routing structure. A routing structure with fewer layers will be returned as the power goes up. Therefore, during the latency descending period, both the enhanced SIR and routing structure lead to a better delay. Nevertheless, the higher power of SUs, the more interference in a network. In order to ensure collision-free scheduling and guarantee successful transmissions in the primary network, SUs have to wait for a longer time. It incurs the increased latency in the following period.

The investigation of how $p_a$ and $\alpha$ influence the latency of PDSA is shown in Table 4.3. In this scenario, we adjust $\alpha$ from 3.8 to 4.2 and $p_a$ from 0.1 to 0.5. The other parameters are $N = 400$, $n = 80$, $\mathbb{P}_p = \mathbb{P}_s = 10$, and $\tau_p = \tau_s = 8$. The results demonstrate that with the augmentation of $\alpha$, the induced transmission latency is decreased. It is because the interference of a sender to other ongoing transmissions in the CRN decreases with the increase of path loss ratio. On the other hand, the delay gets worse with the growth of PUs' activities. The reason is obvious, that is, the spectrum opportunity drops quickly with the increase of $p_a$. Therefore, more time is needed for SUs to get transmission chance.

# Chapter 5

# DATA AGGREGATION SCHEDULING IN WIRELESS NETWORKS WITH COGNITIVE RADIO CAPABILITY

## 5.1 Introduction

In CRNs, unlicensed users can dynamically access and exploit licensed spectrum holes when the spectrum is unoccupied by licensed users. Since unlicensed users need to avoid collisions with the ongoing transmissions between both licensed users and other unlicensed users, the spectrum availability is quite limited for unlicensed users. Furthermore, due to the unpredictable activities of licensed users, unlicensed users can only access the licensed spectrum opportunistically. This uncertainty constrains the usage of CRNs on heavy transmission and time sensitive applications. e.g., fast data aggregation in wireless networks.

In this paper, we concentrate on the investigation of data aggregation in wireless networks with cognitive radio capability. Instead of investigating CRNs that data transmissions among unlicensed users can only rely on the unstable spectrum holes, we study time efficient data aggregation in a wireless network, where the users in the network are equipped with cognitive radios. As in conventional wireless networks, an unlicensed spectrum is assigned to the network. It is available to the in-network users all the time. Meanwhile, the cognitive radio enables wireless users searching and exploiting the spectrum holes. Since a default working spectrum is always guaranteed, regular data transmission can still be processed if there is no spectrum hole. Furthermore, when extra idle spectrum exists, some users can move to that spectrum, so that alleviate contention on the default spectrum and speed up the transmission procedure. In this paper, instead of taking unoccupied spectrum holes as our only hope for data transmission, we consider them as our assistant and use them to accelerate the data aggregation process. Literatures on coexistence of heterogeneous wireless systems can be found, such as the coexistence of ZigBee and Wifi studied in [111] and [112].

A large amount of effort has been dedicated to the investigation of CRNs and data aggregation in wireless networks. Several authors did realize the perspective of introducing cognitive radio capability to improve the performance of wireless networks. For example, [12][13][14][15] studied issues of wireless networks with cognitive radio capability. In all the above four articles, users can work on a stable unlicensed spectrum or access licensed spectrum opportunistically. The authors of [12] focused on the network performance when CSMA is employed. [13] studied the similar issue as [12] with an additional retrial phenomenon. "When and how long to perform spectrum sensing" in cognitive radio enabled smart grid is investigated in [14]. The performance of wireless mesh networks with cognitive radio capability is analyzed in [15]. However, the existing literatures rarely concentrate on how to use cognitive radio technique to promote the performance of data aggregation in conventional wireless networks, which is the focus of this paper. The main contributions of this chapter can be concluded into the following aspects:

- We employs the cognitive radio capability in wireless networks to accelerate data transmission. Subsequently, the Minimum Latency Data Aggregation Scheduling problem in wireless networks with Cognitive Radio capability (MLDAS-CR) is formalized and investigated.

- As the first try, the MLDAS-CR problem is formalized as an Integer Linear Programming (ILP) problem. Considering the hardness of solving an ILP, the optimal solution of its linear programming relaxation is obtained, instead. Subsequently, a rounding algorithm is employed to obtain a feasible solution for the ILP from the optimal solution of the relaxed Linear Programming (LP).

- According to the simulation results, we can see that the ILP and LP based method has a good performance, however, it is difficult to theoretically evaluate the solution. Therefore, a heuristic scheduling algorithm with guaranteed latency bound is presented in our further investigation.

- The simulation results verify the performance of the proposed solutions.

## 5.2 System Model and Problem Formulation

We consider a dense wireless network co-exists with another wireless network who are willing to temporarily release idle spectrum holes. The objective of this paper is to find out how cognitive radio capability can contribute to data transmission in conventional wireless networks. Therefore, for the purpose of distinguishing this work from existing literatures on conventional CRNs, we refer to the two wireless networks as the wireless network and the Auxiliary Network (AN) (the network provides extra spectrum opportunity). The Unit Disk Graph Interference Model is used.

### 5.2.1 Network Model

**Auxiliary Network** (AN): Consider an AN consisting of $m$ Auxiliary Users (AUs) denoted by set $\mathbb{U}_a = \{U_1, U_2, ..., U_m\}$. The transmission radius and interference radius of an AU are denoted by $T_a$ and $I_a$, respectively. Let $S_a$ represent the operating spectrum of AUs. Assume the network time is slotted, where the length of a time slot $\tau_a$ is long enough for AUs to finish the transmission of a data package. At the beginning of each time slot, AUs make their decisions to stay active (receive or send data) or inactive in the current time slot according to the network protocol. The inactive AUs remain silent for the rest time of the current slot. During each time slot, the active senders follow a two-dimensional Poisson point process $X_S$ with density $\lambda$. Apparently, the distribution of the active receiver forms another two-dimensional Poisson point process $X_R$ with the same density $\lambda$.

**Wireless Network**: The considering wireless network consists of $n$ users. Let $\mathbb{U}_r = \{u_1, u_2, ..., u_n\}$ denote the set of users, among which user $u_b \in \mathbb{U}_r$ wants to get the aggregated information from the network. For $u_i \in \mathbb{U}_r$, its transmission and interference radii are denoted by $T_r$ and $I_r$, respectively. There is a default working spectrum $S_r$ physically available to all the users all the time. Each $u_i \in \mathbb{U}_r$ is equipped with a single, half-duplex cognitive radio, which is capable of accessing the default spectrum $S_r$ or adapting parameters to access opportunistically appeared spectrum holes on $S_a$. Assume the time in the wireless network is also slotted, where the length of the time slot $\tau_r$ is long enough for a user to monitor the available spectrum conditions and then transmit a data package. Due to the radio limitation, a user can either transmit or receive data, not both, from all directions at one time slot on a specific spectrum. In each time slot $t$, a user who has data to send can either operates on $S_r$ or opportunistically access $S_a$ in a sensing-before-transmission manner, as long as no collision will be caused to both networks. Particularly, users in the wireless network have equal rights to access $S_r$. However, $S_a$ can be used by non-AUs if and only if no on-going transmissions in AN will be interrupted. That is, the AUs have absolute priority on $S_a$.

### 5.2.2 Problem Formulation

**Definition 5.2.1.** Exterior Collision. *At time $t$, given a link $\overrightarrow{u_s u_r}$ $(u_s, u_r \in \mathbb{U}_r)$, where sender $u_s$ has data to send to receiver $u_r$, if the proceeding of this transmission influences or is influenced by at least one on-going transmission in AN, it is said that there exists an exterior collision. Let $U_s$ and $U_r$ be the transmitter and receiver that affect $u_r$ or affected by $u_s$ when an exterior collision is occurred. We have $||U_s - u_r|| \leq T_r$ or $||U_r - u_s|| \leq I_a$, and $||A - B||$ is the Euclidean distance between $A$ and $B$.*

**Definition 5.2.2.** Interior Collision. *Let $\overrightarrow{u_{s_1} u_{r_1}}$ and $\overrightarrow{u_{s_2} u_{r_2}}$ represent two links in the wireless network, where $u_{s_1}$ and $u_{s_2}$ are senders, and $u_{r_1}$ and $u_{r_2}$ are their corresponding receivers. If the concurrent scheduling of the two links at some time $t$ leads to a collision, then the collision is called an interior collision. Similarly, when an interior collision is caused, $||u_{s_1} - u_{r_2}|| \leq T_r$ or $||u_{r_1} - u_{s_2}|| \leq I_r$ can be derived, and vice versa.*

Based on the network model and definitions, the **Minimum Latency Data Aggregation Scheduling** problem in a wireless network with **Cognitive Radio** (MLDAS-CR) can be formalized as follows:

Given a wireless network denoted by $G = (\mathbb{U}_r, E)$, where $\mathbb{U}_r = \{u_1, u_2, ..., u_n\}$ is the set of wireless users, and $E$ is the set of links ($\overrightarrow{u_s u_r} \in E$ if $||u_s - u_r|| \leq T_r$). A user $u_b \in \mathbb{U}_r$ acts as a base station and desires to obtain aggregated data from the network. Users in $\mathbb{U}_r$

are equipped with cognitive radios capable of adapting transmitting parameters as required. A default working spectrum $S_r$ is allocated to $\mathbb{U}_r$. An AN consists of $m$ AUs operating on spectrum $S_a$. $S_a$ is open to $\mathbb{U}_r$ if no transmission in the AN is affected. Initially, each user $u_i \in \mathbb{U}_r \backslash \{u_b\}$ generates a data package $d_i$. For simplicity, let $\mathbb{D} = \{d_1, d_2, ..., d_n\}$ denote the set of data packages generated in the wireless network, where $d_i (i \neq b)$ is the data generated by user $u_i$. An MLDAS-CR problem can be defined as a schedule set $\mathbb{S} = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_{\mathbb{L}}\}$, where each $\mathbb{S}_t$ $(1 \leq t \leq \mathbb{L})$ is a set of collision-free links in $G$ who are scheduled at time slot $t$. Furthermore, to be an MLDAS-CR, the following constraints are required:

1. $\forall t$ $(1 \leq t \leq \mathbb{L})$, neither an exterior collision nor an interior collision is caused by any scheduled links in $\mathbb{S}_t$.

2. $\forall t$ $(1 \leq t \leq \mathbb{L})$, given two links $\overrightarrow{u_{s_1} u_{r_1}} \in \mathbb{S}_t$ $(s_1 \neq r_1)$ and $\overrightarrow{u_{s_2} u_{r_2}} \in \mathbb{S}_t$ $(s_2 \neq r_2)$ scheduled on either $S_r$ or $S_a$, then $s_1 \neq s_2$, $s_1 \neq r_2$, $s_2 \neq r_1$, and $r_1 \neq r_2$.

3. $\forall t_1, t_2$ $(1 \leq t_1, t_2 \leq \mathbb{L}$, $t_1 \neq t_2)$, if $\overrightarrow{u_{s_1} u_{r_1}} \in \mathbb{S}_{t_1}$ and $\overrightarrow{u_{s_2} u_{r_2}} \in \mathbb{S}_{t_2}$, then $s_1 \neq s_2$.

4. $f(\bigcup\limits_{t=1}^{\mathbb{L}} \{d_{\overrightarrow{u_{s_t} u_b}} | \forall \overrightarrow{u_{s_t} u_b} \in \mathbb{S}_t\}) = f_A(\mathbb{D})$, where $d_{\overrightarrow{u_{s_t} u_b}}$ is the data package received by $u_b$ at time $t$ through link $\overrightarrow{u_{s_t} u_b}$, $f_A$ is the aggregate function and $f_A(\mathbb{D})$ is the aggregated result over $\mathbb{D}$.

5. If $t = \mathbb{L}$, the transmission is $\overrightarrow{u_s u_b}$, and $\bigcup\limits_{t=1}^{\mathbb{L}-1} \{u_{s'} | \overrightarrow{u_{s'} u_{r'}} \in \mathbb{S}_t\} \bigcup \{u_s\} \bigcup \{u_b\} = \mathbb{U}_r$.

6. $\arg\min_{\mathbb{S}=\{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_{\mathbb{L}}\}} \mathbb{L}$.

Constraint 1 shows MLDAS-CR should be exterior and interior collisions free. No exterior collision guarantees that no interference will be caused to AN, and interior collision free avoids extra delay and congestion caused by retransmission in the wireless network. Since $u_i$ has only one radio, constraint 2 requires $u_i$ can either be a sender or receiver at a particular time slot $t$, but not both. Constraint 3 indicates the property of data aggregation, that is, each user sends its aggregation result (aggregated data of its own and data received during the MLDAS-CR) only once. The data integrity property is ensured by constraint 4, where data received by the base station should be the aggregated information of the whole network. At the last time slot, the base station should receive the last transmission from a SU as specified in 5. Constraint 6 denotes that the objective of the MLDAS-CR scheduling $\mathbb{S}$ is to minimize the total transmission latency.

It is known that the Minimum Latency Data Aggregation Scheduling (MLDAS) problem in wireless network is NP-hard without considering the cognitive radio capability. It can be considered as a special case of MLDAS-CR when the AUs in AN are so dense and active that no spectrum holes exist. Therefore, the MLDAS-CR problem is NP-hard.

## 5.3 Scheduling Algorithm for MLDAS-CR

In this section, we first introduce the construction of a balanced Connected Dominating Set-based tree, which serves as the routing tree during the data aggregation process. Subsequently, two scheduling algorithms for MLDAS-CR are discussed in detail.

### 5.3.1 Construction of a Balanced Routing Tree

Given a graph, a Connected Dominating Set (CDS) is a connected component with the property that for every vertex on the graph, it is either in the CDS or has some one-hop neighbor in the CDS. This property makes CDS quite suitable for serving as routing infrastructure in wireless networks. However, it is not the case that an arbitrary CDS-based routing tree is efficient for the data aggregation application. In this paper, a Balanced CDS-based Routing Tree (BRT) is employed for the purpose of distributing transmission workload evenly, reducing the delay of users with large degree, and then accelerating the aggregation process.

**Definition 5.3.1.** 2-norm. *Given a vector $X = (x_1, x_2, ..., x_n)$, the 2-norm of $X$ is defined as:* $|X|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2}$.

According to [90], given a vector $X$ as defined in Def. 5.3.1, $|X|_2$ can be used to measure the balance among all variables $x_i$ $(1 \leq i \leq n)$.

Let vector $W = (w_1, w_2, ..., w_n)$ denote the workload, where $w_i$ represents the load allocates to $u_i$. Then, $|W|_2$ can be used to measure how balance the workload is distributed among users in $\mathbb{U}_r$. Especially, the smaller $|W|_2$, the more balance of workload allocation. Initially, $\forall i, w_i = 0$. The construction of BRT can be described as follows:

Step 1: Set the layer of $u_b$ as 0, and build a Breadth First Search (BFS) tree rooted at $u_b$. Then, search the BFS tree from root to leaves, by layer, mark all the users who form a maximal independent set BLACK.

Step 2: Start at the 2nd layer, mark the parent of BLACK nodes GRAY. Subsequently, for each GRAY node, find a BLACK node from the same layer or one upper layer to be its parent. During this process, update $W$ for BLACK nodes according to their number of GRAY children. If multiple choices are available to a GRAY node, then the BLACK gives minimum $|W|_2$ after allocation will be chosen as its parent.

Step 3: In the last step, the unmarked WHITE nodes are balanced allocated to BLACK nodes. In order to obtain a BRT, a WHITE node accepts the BLACK in its one-hop neighborhood who can minimize $|W|_2$ as its parent. The details are illustrated in Alg. 4. Finally, from root to leaves, each node updates its layer according to its parent's layer.

**Algorithm 4:** Balanced Allocation

**input** : The tree gets from step 2
**output**: BRT

**for** *each* $w_i$ **do**
    $w_i = 0$;
**for** *each unmarked node* $u_i$ **do**
    mark in WHITE;
    check all neighbors in BLACK denoted as set $NB(i)$;
    **if** $u_j \in NB(i)$ *and the allocation of* $u_i$ *to* $u_j$ *achieves the minimum increase of* $|W|_2$ **then**
        set $u_j$ as $u_i$'s parent;
        update $w_j$ accordingly.

Since the maximum number of WHITE nodes is $n-1$, and the number of black nodes in a white node's one-hop neighborhood is no more than 5 (Lemma 9), therefore, the running time of Alg. 4 is $O(n)$. Based on the construction of BRT, the following lemma holds:

**Lemma 6.** *All the BLACK users are in even layers. All GRAY users are in odd layers. Each GRAY user has a BLACK parent and at least one BLACK child. The BLACK and GRAY users form a CDS. Any WHITE user is leave on the BRT and has a BLACK parent.*

Particularly, in order to intuitionally show links on BRT, we transfer $G$ to $G^B = \{\mathbb{U}_r, \{E^B, E\}\}$, where $E^B$ contains the links on the BRT. For simplicity, we only consider the directed links from children to parents in $E^B$, while ignore links in the opposite direction. The reason is that data in the network is aggregated only from bottom (children) to top (parent) on the BRT.

### 5.3.2 Scheduling Algorithm Based on LP

In this subsection, a mathematical model is employed to formalize the MLDAS-CR problem. According to the formalization, a scheduling algorithm based on Linear Programming (LP) is discussed in detail.

We define two scheduling variables $R_{ij}^t$ and $A_{ij}^t$ as:

$$R_{ij}^t = \begin{cases} 1, \text{if link } \overrightarrow{u_i u_j} \text{ is scheduled on } S_r \text{ at time } t \\ 0, \text{otherwise} \end{cases}$$

and,

$$A_{ij}^t = \begin{cases} 1, \text{if link } \overrightarrow{u_i u_j} \text{ is scheduled on } S_a \text{ at time } t \\ 0, \text{otherwise} \end{cases}$$

Variable $Y_{ij}$ is used to indicate the AUs' activity around link $\overrightarrow{u_i u_j}$, where

$$Y_{ij}^t = \begin{cases} 1, \text{if schedule } \overrightarrow{u_i u_j} \text{ at } t \text{ cause exterior collision} \\ 0, \text{otherwise} \end{cases}$$

To be specific, at time $t$, $Y_{ij}^t = 1$ if there is at least one receiver in AN active in $u_i$'s interference range or one sending activity is detected within $u_j$'s transmission range at $t$.

Let $Q_i^t = 1$ indicate that $u_i$ has obtained data from all its children at $t$, Otherwise, $Q_i^t = 0$. Apparently, the $Q$ variable of any WHITE node on the BRT is 1.

Due to the constraint of half-duplex radio, a user can active as a sender or receiver but not both on a particular spectrum at a particular time. Therefore, at a specific time $t$, for an arbitrary user $u_a$ on BRT, it may keep silent or play one role on $S_a$ or $S_r$, that is:

$$\sum_{\overrightarrow{u_a u_i} \in E^B} R^t_{ai} + \sum_{\overrightarrow{u_i u_a} \in E^B} R^t_{ia} + \sum_{\overrightarrow{u_a u_i} \in E^B} A^t_{ai} + \sum_{\overrightarrow{u_a u_i} \in E^B} A^t_{ia} \leq 1 \tag{5.1}$$

On the other hand, to prevent re-transmission and unnecessary energy consumption from transmission collision, a scheduled link cannot interrupt or be interrupted by any on-going transmission in both networks. That is, the scheduling of link $\overrightarrow{u_s u_r} \in E^B$ cannot result in exterior or interior collision.

For a particular time $t$, to avoid interior collision, InEq. (5.2) is required.

$$R^t_{sr} + \sum_{u_i \in NB(u_r), \overrightarrow{u_i u_j} \in E^B} R^t_{ij} \leq 1 \tag{5.2}$$

and, on the other hand, to avoid interference with the activities in AN,

$$A^t_{sr} + \sum_{u_i \in NB(u_r), \overrightarrow{u_i u_j} \in E^B} A^t_{ij} \leq 1 \tag{5.3}$$

where $NB(u_i)$ is the set of one-hop neighbor of $u_i$ in the wireless network.

According to InEq. (5.2) and (5.3), the following constraint specifies the property that a confliction-free scheduling plan should have.

$$R^t_{sr} + A^t_{sr} + \sum_{u_i \in NB(u_r), \overrightarrow{u_i u_j} \in E^B} (R^t_{ij} + A^t_{ij}) \leq 1 \tag{5.4}$$

Furthermore, link $\overrightarrow{u_s u_r} \in E^B$ can be scheduled on $S_a$ if and only if $S_a$ is available, i.e.,

$$A^t_{sr} \leq 1 - Y^t_{sr} \tag{5.5}$$

Since a node needs to wait for all its children for aggregating data, for $\overrightarrow{u_s u_r} \in E^B$,

$$R_{sr}^t \leq Q_s^t, A_{sr}^t \leq Q_s^t \tag{5.6}$$

With the purpose of ultimately utilizing spectrum holes on $S_a$ and reducing spectrum competition on $S_r$, the utility function of scheduling $\overrightarrow{u_s u_r} \in E^B$ at $t$ is defined as:

$$f_{sr}^t = R_{sr}^t + (\alpha d_s + \beta l_s + 1)A_{sr}^t \tag{5.7}$$

where $d_s$ and $l_s$ are the degree and layer of $u_s$, respectively. Two variables $\alpha$ and $\beta$ are used to adjust the weight of the two properties according to demand.

Based on the above constraints, we can conclude the MLDAS-CR problem as:

$$\begin{aligned}
\text{Maximize} \quad & \frac{1}{L}\sum_{t=1}^{L} \sum_{\overrightarrow{u_s u_r} \in E^B} f_{sr}^t \\
\text{subject to} \quad & \sum_{\overrightarrow{u_a u_i} \in E^B} R_{ai}^t + \sum_{\overrightarrow{u_i u_a} \in E^B} R_{ia}^t \\
& + \sum_{\overrightarrow{u_a u_i} \in E^B} A_{ai}^t + \sum_{\overrightarrow{u_a u_i} \in E^B} A_{ia}^t \leq 1 \\
& R_{sr}^t + A_{sr}^t + \sum_{u_i \in NB(u_r), \overrightarrow{u_i u_j} \in E^B} (R_{ij}^t + A_{ij}^t) \leq 1 \\
& A_{sr}^t \leq 1 - Y_{sr}^t \\
& R_{sr}^t \leq Q_s^t \\
& A_{sr}^t \leq Q_s^t \\
& 1 \leq t \leq L, \in R_{sr} \in \{0,1\}, A_{sr} \in \{0,1\}, L \in N
\end{aligned}$$

where $L$ is assumed to be the length of the scheduling time.

Even though the introduced objective function and constraints formalize the MLDAS-CR problem to a 0-1 Integer Linear Program (ILP), we are in a dilemma to find a scheduling plan based on the formalization. The major difficulty we are facing is that the activity $Y_{sr}^t$ for link $\overrightarrow{u_s u_r} \in E^B$ at time $t$ is unpredictable. There is no way we can get the information of $Y_{sr}^t$ until time $t$.

Therefore, instead of solving the problem considering continuous time, we switch to find optimal scheduling for each time slot. That is, at a particular time $t$, given $Y_{sr}^t$ for any $\overrightarrow{u_s u_r} \in E^B$, how can we make the best decision so that we can get the maximum number of links scheduled? In this case, we only care about links in $E^B$ denoted as $E^{B'}$ that have not been scheduled yet. For simplicity, in the description below, $t$ is removed from the superscript. Then we have:

$$
\begin{aligned}
\text{Maximize} \quad & \sum_{\overrightarrow{u_s u_r} \in E^{B'}} f_{sr} \\
\text{subject to} \quad & \sum_{\overrightarrow{u_a u_i} \in E^{B'}} R_{ai} + \sum_{\overrightarrow{u_i u_a} \in E^{B'}} R_{ia} \\
& + \sum_{\overrightarrow{u_a u_i} \in E^{B'}} A_{ai} + \sum_{\overrightarrow{u_a u_i} \in E^{B'}} A_{ia} \leq 1 \\
& R_{sr} + A_{sr} + \sum_{u_i \in NB(u_r), \overrightarrow{u_i u_j} \in E^{B'}} (R_{ij} + A_{ij}) \leq 1 \\
& A_{sr} \leq 1 - Y_{sr} \\
& R_{sr} \leq Q_s \\
& A_{sr} \leq Q_s \\
& R_{sr} \in \{0,1\}, A_{sr} \in \{0,1\}
\end{aligned}
$$

Solving the ILP for time $t$ is still at least NP-hard. However, a Linear Program (LP) is polynomial-time solvable. Therefore, a natural choice is to derive an LP by relaxing the constraints $R_{sr} \in \{0,1\}, A_{sr} \in \{0,1\}$ to $0 \leq R_{sr}, A_{sr} \leq 1$. Instead of solving the ILP, the optimal solution of LP can be obtained by an LP solver. After that, a rounding algorithm (as shown in Alg. 5) is employed to get a feasible solution for the ILP. Based on the output of Alg. 5, all the links with $A_{sr} = 1$ can be scheduled on $S_a$, and links with $R_{sr} = 1$ should be scheduled on $S_r$. Then, we can obtain a data aggregation scheduling for MLDAS-CR by iteratively solving the LP problem according to the dynamic network condition, the details are presented in Alg. 6.

**Algorithm 5:** Rounding Algorithm

**input** : Optimal solution from LP
**output**: Feasible solution for ILP

Sort the input by non-descending order denoted as $\mathcal{L} = \{\overrightarrow{u_{s_1}u_{r_1}}^*, \overrightarrow{u_{s_2}u_{r_2}}^*, ...\}$;
**while** $\mathcal{L} \neq \emptyset$ **do**
  **for** *the ordered unmarked links in* $\mathcal{L}$ **do**
    **if** $Y_{sr} = 0$ **then**
      mark $\overrightarrow{u_s u_r}$, set $A_{sr} = 1$, $R_{sr} = 0$;
      **for** *all the links conflict with* $\overrightarrow{u_{s'}u_{r'}}$ *in* $\mathcal{L}$ **do**
        mark $\overrightarrow{u_{s'}u_{r'}}$, set $A_{s'r'} = R_{s'r'} = 0$;
    **else**
      mark $\overrightarrow{u_s u_r}$, set $A_{sr} = 0$, $R_{sr} = 1$;
      **for** *all the links conflict with* $\overrightarrow{u_{s'}u_{r'}}$ *in* $\mathcal{L}$ **do**
        mark $\overrightarrow{u_{s'}u_{r'}}$, set $A_{s'r'} = R_{s'r'} = 0$;

**Algorithm 6:** SLP (Scheduling based on LP)

**input** : $G^B = \{\mathbb{U}_r, \{E^B, E\}\}$
**output**: Schedule $\mathbb{S} = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_{\mathbb{L}}\}$

$t = 0$;
**for** *each user* $u_i \in \mathbb{U}_r$ **do**
  **if** $u_i$ *is WHITE* **then**
    set $Q_i^t = 1$;
  **else**
    set $Q_i^t = 0$;
**while** $E^B$ *is not empty* **do**
  $t$++;
  **for** *each link* $\overrightarrow{u_s u_r} \in E^B$ **do**
    sense spectrums, and set $Y_{sr}^t$ accordingly;
  solve the formulated LP;
  call Alg. 5;
  **for** *each* $R_{sr}^t = 1$ *or* $A_{sr}^t = 1$ **do**
    **if** $R_{sr}^t = 1$ **then**
      schedule $\overrightarrow{u_s u_r}$ on $S_r$;
    **else**
      schedule $\overrightarrow{u_s u_r}$ on $S_a$;
    $\mathbb{S}_t = \mathbb{S}_t \bigcup \{\overrightarrow{u_s u_r}\}$;
    remove $\overrightarrow{u_s u_r}$ from $E^B$;
  **for** *each user* $u_i \in \mathbb{U}_r$ **do**
    update $Q_i^t$ accordingly;

### 5.3.3 Scheduling with Expected Delay Guarantee

As shown in Section 5.3.2 and 6.4, a feasible scheduling policy based on the LP can be derived and its performance is good. However, it is difficult to theoretically show that how well the feasible solution is . In this subsection, we focus on scheduling algorithm with expected delay guarantee. Meanwhile, the algorithm should be easy to implement.

According to Lemma 12, links in $E^B$ can be classified into three types: the sender $u_s$ is WHITE and the receiver $u_r$ is BLACK, $u_s$ is BLACK and $u_r$ is GRAY, and $u_s$ is GRAY and $u_r$ is BLACK. For simplicity, let $l_{wb}$, $l_{bg}$, and $l_{gb}$ denote the three kinds of links, respectively.

**Definition 5.3.2.** *Interior Interference Link Set. Given $\overrightarrow{u_s u_r} \in E^B$, the Interior Interference Link Set (IILS) of $\overrightarrow{u_s u_r}$, denoted as $\mathbb{I}_{sr}$, is defined as all the links in $E^B$ active on $S_r$ or $S_a$ which will cause interior collision if $\overrightarrow{u_s u_r}$ is scheduled on the same spectrum ($S_r$ or $S_a$, accordingly). Furthermore, $|\mathbb{I}sr|$ is defined as the* interference degree *of link $\overrightarrow{u_s u_r}$, which is the total number of links that may interfere with $\overrightarrow{u_s u_r}$ in the wireless network.*

Given link set $\mathcal{L}$, the *conflict graph* of $\mathcal{L}$ denoted by $C[\mathcal{L}]$ is an undirected graph on $\mathcal{L}$ in which there is an edge between two links if they cannot be scheduled simultaneously without collision. Then, given a link set $\mathcal{L}$, the interference degree of $\overrightarrow{u_s u_r}$ is equal to its degree on $C[\mathcal{L}]$, and $\mathbb{I}_{sr}$ is $\overrightarrow{u_s u_r}$'s one-hop neighbor on $C[\mathcal{L}]$.

The FFS algorithm (Alg. 7, Alg. 8), which based on the BRT constructed in Section 5.3.1, can be concluded into two stages:

Stage 1: All the links of type $l_{wb}$ are scheduled. A link $\overrightarrow{u_s u_r}$ in $E^B$ is said *ready* to be scheduled if the sender $u_s$ has received data from all of its children. Since the WHITE users have no children, they are ready for transmission. Firstly, links of type $l_{wb}$ are sorted in a non-decreasing order according to their interference degrees. For simplicity, let $\mathcal{L}_{wb} = \{\overrightarrow{u_{s_1} u_{r_1}}, \overrightarrow{u_{s_2} u_{r_2}}, ..., \overrightarrow{u_{s_w} u_{r_b}}\}$ denote the set of sorted links. Subsequently, a first-fit scheduling policy is employed to schedule the sorted links in $\mathcal{L}_{wb}$. To be specific, links in $\mathcal{L}_{wb}$ are considered in order from $\overrightarrow{u_{s_1} u_{r_1}}$ to $\overrightarrow{u_{s_w} u_{r_b}}$. For a link $\overrightarrow{u_{s_i} u_{r_i}}$, check its spectrum availability and collision status, and then schedule the link whenever a transmission opportunity exist.

Alg. 7 shows this stage in detail.

**Algorithm 7:** FFS-S1 (First-Fit Scheduling Stage 1)

**input** : $G^B = \{\mathbb{U}_r, \{E^B, E\}\}$
**output**: Schedule $\mathbb{S} = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_t\}$

$t = 0$;
Sort links in $E^B$ of type $l_{wb}$ in non-decreasing order according to their interference degree;
$\mathcal{L}_{wb} = \{\overrightarrow{u_{s_1}u_{r_1}}, \overrightarrow{u_{s_2}u_{r_2}}, ..., \overrightarrow{u_{s_w}u_{r_b}}\}$ denote the set of sorted links;
**while** $\mathcal{L}_{wb} \neq \emptyset$ **do**
    $t{+}{+}$;
    **for** *each link in* $\mathcal{L}_{wb}$ **do**
        **if** $\mathbb{E}_{s_i r_i} = \emptyset$ **then**
            schedule $\overrightarrow{u_{s_i}u_{r_i}}$ on $S_a$;
        **else if** $\mathbb{I}_{s_i r_i} = \emptyset$ **then**
            schedule $\overrightarrow{u_{s_i}u_{r_i}}$ on $S_r$;
    **for** *each scheduled link* $\overrightarrow{u_{s_i}u_{r_i}}$ **do**
        $\mathbb{S}_t = \mathbb{S}_t \bigcup \{\overrightarrow{u_{s_i}u_{r_i}}\}$;
        remove $\overrightarrow{u_{s_i}u_{r_i}}$ from $E^B$ and $\mathcal{L}_{wb}$;

Stage 2: Iteratively schedule links of type $l_{bg}$ and $l_{gb}$. Let $E^{B'}$ denote the set of unscheduled links in $E^B$ after Stage 1. The scheduling conducts iteratively, where the ready $l_{bg}$ links are scheduled in even iterations, and the ready $l_{gb}$ links are scheduled in odd iterations. During each iteration, ready links are sorted in non-decreasing order according to their interference degree. After that, a first-fit scheduling policy is applied to arrange the scheduling plan. The detailed pseudocode is shown in Alg. 8. According to Alg. 8, we can see that each "iteration" may consist of several time slots, and the length of different "iterations" may be different. That depends on the required time for scheduling the ready links under consideration.

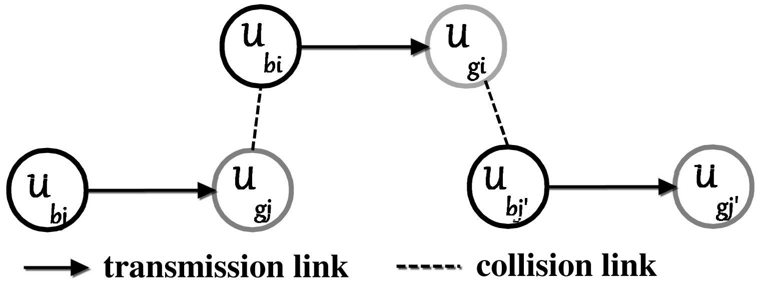
In the following part, we analyze the latency of the proposed FFS algorithm.

**Lemma 7.** *The expected number of spectrums available to a link in the wireless network is* $1 + e^{-\pi\lambda(T_r^2+T_a^2)}$*, where* $T_r$ *and* $T_a$ *are the transmission radius for users in the wireless network and AN, respectively.*

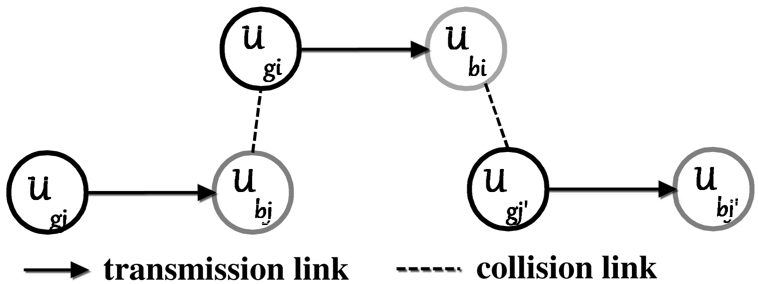
**Algorithm 8:** FFS-S2 (First-Fit Scheduling Stage 2)

**input** : $G^B = \{\mathbb{U}_r, \{E^{B'}, E\}\}$
**output**: Schedule $\mathbb{S} = \{\mathbb{S}_{t+1}, \mathbb{S}_{t+2}, ..., \mathbb{S}_{\mathbb{L}}\}$

1 $t = t + 1, iter = 0$;
2 **while** *Not all scheduled* **do**
3     **if** $iter\%2 = 0$ **then**
4         Sort *ready* links in $E^{B'}$ of type $l_{bg}$ in non-decreasing order according to their interference degree;
5         let $\mathcal{L}_{bg} = \{\overrightarrow{u_{s_1}u_{r_1}}, \overrightarrow{u_{s_2}u_{r_2}}, ..., \overrightarrow{u_{s_b}u_{r_g}}\}$ denote the set of sorted links;
6         **while** $\mathcal{L}_{bg} \neq \emptyset$ **do**
7             **for** *each link in* $\mathcal{L}_{bg}$*: from* $\overrightarrow{u_{s_1}u_{r_1}}$ *to* $\overrightarrow{u_{s_b}u_{r_g}}$ **do**
8                 **if** $\mathbb{E}_{s_i r_i} = \emptyset$ **then**
9                     schedule $\overrightarrow{u_{s_i}u_{r_i}}$ on $S_a$;
10                 **else if** $\mathbb{I}_{s_i r_i} = \emptyset$ **then**
11                     schedule $\overrightarrow{u_{s_i}u_{r_i}}$ on $S_r$;
12             **for** *each scheduled link* $\overrightarrow{u_{s_i}u_{r_i}}$ **do**
13                 $\mathbb{S}_t = \mathbb{S}_t \bigcup \{\overrightarrow{u_{s_i}u_{r_i}}\}$;
14                 remove $\overrightarrow{u_{s_i}u_{r_i}}$ from $E^{B'}$;
15             $t{+}{+}$;
16     **else**
17         repeat step 4 to 15 but replace links of $l_{bg}$ with $l_{gb}$;
18     $iter{+}{+}$;




(a) Collision with link $l_{bg}$




(b) Collision with link $l_{gb}$

Figure 5.1. Example of Collision.

Given a connected graph $G = \{V, E\}$, let $G[U]$ denote a subgraph of $G$ induced by $U \subseteq V$, $\Delta$ and $\delta$ are the maximum and minimum degree of $G$, respectively. The inductivity of $G$ is defined as $\delta^*(G) = MAX_{U \subseteq V}(G[U])$.

**Lemma 8.** *Given an non-decreasing ordering $O =< o_1, o_2, ..., o_n >$, let $d_i$ denote the degree of $o_i$, it is proved that a first-fit coloring policy in smallest-degree-last ordering uses at most $1 + \delta^*$ colors, where $\delta^* = MAX_{1<i<n}|d_i|$ [40].*

**Lemma 9.** *Let $C$ represent a disk of radius $r$, and $U$ is a set of points with mutual distance at least 1, then the number of points with mutual distance at least 1 on the disk is upper bounded by $\frac{2\pi}{\sqrt{3}}r^2 + \pi r + 1$, that is, $|U \bigcup C| \leq \frac{2\pi}{\sqrt{3}}r^2 + \pi r + 1$ [40].*

**Lemma 10.** *The expected latency for FFS-S1 is upper bounded by $\frac{5\Delta}{1+e^{-\pi\lambda(T_r^2+T_a^2)}}$, where $\Delta$ is the maximum degree of $G$.*

*Proof.* According to Alg. 7, in FFS stage 1, all links of type $l_{wb}$ are scheduled. The algorithm employ a first-fit scheduling based on the ordering of links' interference degree. Let $\mathcal{L}_{wb}$

denote the set of $l_{wb}$ links. Then, the latency is upper bounded by $\delta^*(C[\mathcal{L}_{wb}]) + 1$ (Lemma 8), where $\delta^*(C[\mathcal{L}_{wb}])$ is the inductivity of $\mathcal{L}_{wb}$'s conflict graph. Furthermore, $\delta^*(C[\mathcal{L}_{wb}]) = MAX_{\overrightarrow{u_s u_r} \in \mathcal{L}_{wb}} |\mathbb{I}_{w_i b_i}|$, which is the maximum degree of $C[\mathcal{L}_{wb}]$. Assume the abstraction of $l_{w_i b_i}$ is the vertex with maximum degree in $C[\mathcal{L}_{wb}]$, where the degree is equvilent to the number of links that conflict with $l_{w_i b_i}$. Based on the network model and interference model specified in Section 5.2.1, link $\overrightarrow{u_s u_r}$ that cannot be scheduled simultaneously with $l_{w_i b_i}$ have the property that $||l_{w_i} - u_r|| \leq T_r$ or $||l_{b_i} - u_s|| \leq T_r$. Since only $l_{wb}$ links are scheduled in FFS stage 1, for any conflicting link $l_{w_j b_j}$, we have $||l_{w_i} - l_{b_j}|| \leq T_r$ or $||l_{b_i} - l_{w_j}|| \leq T_r$, where $||l_{w_i} - l_{b_j}|| \leq T_r$ contains the BLACK users in $u_{w_i}$'s one-hop neighborhood, and $||l_{b_i} - l_{w_j}|| \leq T_r$ specifies $u_{b_i}$'s one-hop WHITE neighbors. According to Lemma 9, a WHITE user may have at most 5 BLACK one-hop neighbors, where one of them is its parent based on the construction of the BRT. If $\Delta$ denotes the maximum degree of $G$ and $G^B$, then, for each BLACK node, it has at most $\Delta$ WHITE neighbors. Therefore, $|\mathbb{I}_{w_i b_i}| \leq 4 * \Delta + \Delta - 1 = 5\Delta - 1$. According to Lemma 8, $5\Delta$ spectrums are needed to color the links. Since we only have $1 + e^{-\pi\lambda(T_r^2 + T_a^2)}$ spectrum available, the iteration we need to finish the scheduling is $\frac{5\Delta}{1+e^{-\pi\lambda(T_r^2+T_a^2)}}$. □

**Lemma 11.** *The expected latency for FFS-S2 is at most* $\frac{44D+1}{1+e^{-\pi\lambda(T_r^2+T_a^2)}}$*, where* $D$ *is the diameter of the wireless network.*

*Proof.* The proof of stage 2 is similar to Lemma 10, we concentrate on finding the inductivity of the conflict graph for the link set in each iteration. Since the algorithm performs iteratively, the latency can be derived based on the following two propositions.

**Proposition 1:** The latency for the even iteration is at most $\frac{22}{1+e^{-\pi\lambda(T_r^2+T_a^2)}}$.

**Proposition 2:** The latency for the odd iteration is upper bounded by $\frac{22}{1+e^{-\pi\lambda(T_r^2+T_a^2)}}$ if $u_{b_i} \neq u_b$, and $\frac{23}{1+e^{-\pi\lambda(T_r^2+T_a^2)}}$, otherwise.

Finally, based on the construction of the BRT, the maximum number layer of the BRT is upper bounded by $2D$, where $D$ is the diameter of the wireless network (the hops between the farthest two users in the wireless network). According to Alg. 8, Case 1 and case 2 will alternatively run at most $D$ iterations, respectively. Therefore, the expected latency for stage 2 is $\frac{44D+1}{1+e^{-\pi\lambda(T_r^2+T_a^2)}}$. □

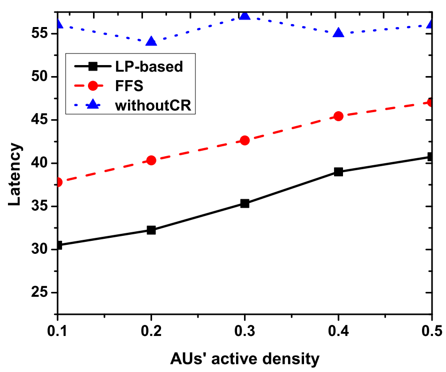
**Theorem 2.** *The expected latency for the proposed first-fit scheduling algorithm is* $\frac{5\Delta+44D+1}{1+e^{-\pi\lambda(T_r^2+T_a^2)}}$, *where* $\Delta$ *and* $D$ *are the maximum degree and diameter of the wireless network, respectively.*

The proof of theorem 3 can be directly derived from Lemma 10 and Lemma 11.

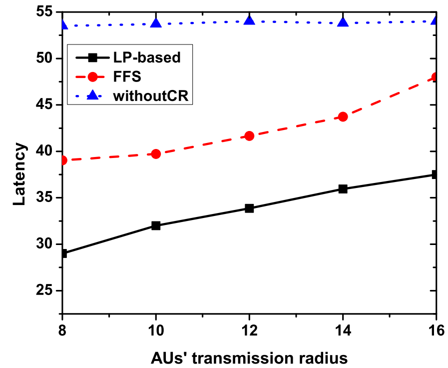
### 5.4 Performance Evaluation

In this section, we evaluate the performance of our proposed scheduling algorithms with respect to different network parameters. To keep consistency with Section 5.2, the same notations are used in this section. To be specific, let $m$, $n$ denote the number of AUs and wireless users, respectively; $T_a$ (respectively, $T_r$) is the transmission radius of AUs (respectively, wireless users). At each time slot, the active senders and receivers in the AN follow Poisson Distribution with density $\lambda$. The network configuration is initially set up as: $A = 100 * 100$, $m = 100$, $n = 400$, $T_a = T_r = 1.5$, and $\lambda = 0.3$. For simplicity, the utility function for LP is defined as $f_{sr}^t = R_{sr}^t + 1.2 * A_{sr}^t$, where users are encouraged to use the auxiliary spectrum if allowed. In the simulation, in order to verify the influence of different parameters on the proposed algorithms, we adjust one of the parameters per time while keep the rest unchanged. Particularly, the performance of our proposed algorithms are compared with the SAS algorithm. SAS is a sequential aggregation scheduling algorithm based on CDS- aggregation tree and first fit coloring scheduling algorithm proposed in [40]. It is the algorithm we can find in existing literature with the best latency bound $15R + \Delta - 4$, where $R$ is the network radius and $\Delta$ is the maximum degree. For comparison, in SAS, we assume only $S_r$ is available.

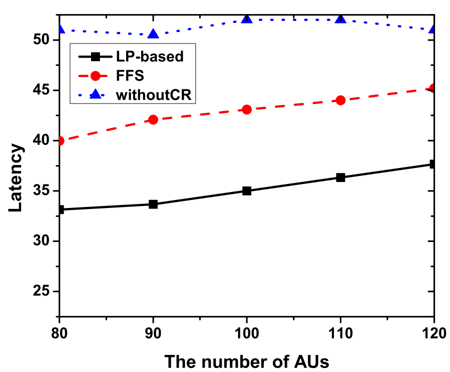
The results are shown in Fig. 5.2, where the impacts of AUs are evaluated in Fig. 5.2(a) and Fig. 5.2(c), and the influence of RUs are tested in Fig. 5.2(d) and Fig. 5.2(e). For simplicity, “LP-based” is used to represent the algorithm proposed in Section 5.3.2, “FFS” refers to the first-fit scheduling algorithm introduced in Section 5.3.3, and “withoutCR” refers to the SAS algorithm. The performance of “LP-based” and “FFS” is compared with the performance of “withoutCR”. According to the results shown in Fig. 5.2, the performance of the proposed algorithms outperform the comparison algorithm in all aspects, which clearly



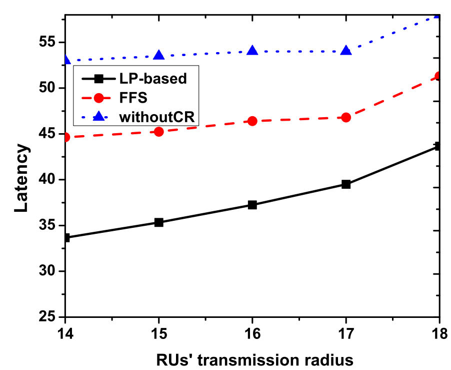

(a) Impact of AUs' active density



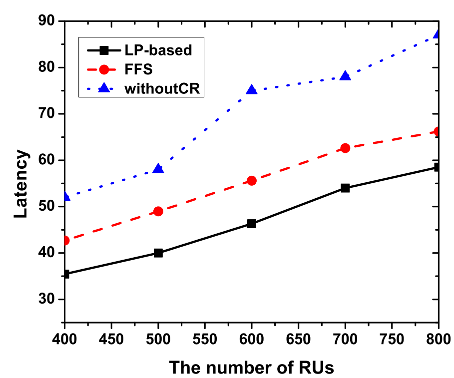

(b) Impact of AUs' transmission radius




(c) Impact of AUs' population




(d) Impact of RUs' transmission radius




(e) Impact of RUs' population

Figure 5.2. Performance Evaluation.

shows the advantage of cognitive radio capability. Since “LP-based” and “FFS” seek transmission opportunity on both $S_a$ and $S_r$, the delay for the two algorithms is shorter than “withoutCR” which only relies on $S_r$. Particularly, “LP-based” generates the scheduling plan based on an optimum algorithm, which achieves a better time performance compared with “FFS”. Furthermore, because the scheduling of “withoutCR” has nothing to do with $S_a$, so that changes on the AN does not affect the time performance of “withoutCR”.

The performance of the three algorithms with respect to the change of AUs’ active density is evaluated in Fig. 5.2(a). With the increasing of active AUs’ density, more senders and receivers are active in the AN at each time slot, which results in a higher risk of exterior collisions. In order to avoid collision, the number of links which are scheduled on $S_a$ at the same time slot decreases, therefore, more time is required to finish the data aggregation. Fig. 5.2(b) shows that the augment of AUs’ transmission radius causes a longer scheduling delay. The reason is that, a larger transmission radius forms a bigger interference range. The increased interference range prevents more users scheduled on $S_a$ if an AU is active at a particular time, hence, leads to more delay. Similar to the above two scenarios, AUs’ population has a negative influence on the time performance of “LP-based” and “FFS” (as shown in Fig. 5.2(c)). The growth of AUs’ population introduces more active AUs into AN, so that enlarge the effect of exterior collisions, which results in more delay in the end.

We verify the influence of RUs’ transmission radius and population in Fig. 5.2(d) and Fig. 5.2(e), respectively. In Fig. 5.2(d), we can see that the latency increases with the increasing of RUs’ transmission radius. The reason is similar to the influence of AUs transmission radius. It has no relation with the exterior collision, however, about the interior collision instead. The change of transmission radius may influence BRT, however, the increased transmission radius has more negative effect on enlarging the interference range of RUs, which results in an reduction of the number of RUs that can be scheduled concurrently. Therefore, more time is needed. Since the exterior interference which comes from AN and the interior interference that comes from other wireless users are inevitable, so that the number of RUs that can be scheduled collision-free at each time slot is limited. Therefore, the latency increases with the growth of RUs' population. Particularly, both RUs' transmission radius and population affect the network condition on $S_r$, that is why we can see the same trend on "LP-based", "FFS", and "withoutCR".

# Chapter 6

# DATA AGGREGATION SCHEDULING IN PROBABILISTIC WIRELESS NETWORKS WITH COGNITIVE RADIO CAPABILITY

## 6.1 Introduction

Wireless Networks are one of the most important technologies in our daily life and have been used in a variety of applications. The ideal Deterministic Network Model (DNM) has been extensively employed in the investigation of existing literatures on wireless networks. Under DNM, wireless links are considered to be very stable. Users in a DNM wireless network are considered to be either connected or disconnected. A deterministic and reliable link is assumed to exist between two connected users. Therefore, data transmission is guaranteed to be always successful as long as there is no collision or interference from the environment. That is, a package sent by the sender can always be received successfully by the receiver with full accuracy if all other situations are ideal. In reality, the *Transitional Region Phenomenon* verified in [58][59] and [18], demonstrates a different conclusion. Observations from [58], [59] and [18] reveal that wireless signal fades due to interference, distance, and some other factors leading to lossy links. Particularly, [18] shows that in wireless sensor networks, more than 90% of the links are lossy. Even without collision or interference from the environment, the existence of lossy links make that the success of a transmission between two users who are theoretically connected under the DNM cannot be guaranteed. This means that the DNM cannot fully characterize the property of wireless links.

In order to better characterize the lossy links in wireless networks, a more practical network model called Probabilistic Network Model (PNM) has been proposed. Under P-NM, a transmission between two users will succeed with a certain probability rather than guaranteed in their connectivity. For a sender and a receiver pair, the probability that the receiver can get a data package successfully (package is received and data is correct) relying on both the connectivity and link quality. [18] models lossy links in wireless networks as leaky-pipes, which demonstrates the special property of lossy links in a vivid way. If the link quality is poor, there is a high probability that the receiver cannot obtain a data package accurately even while it may still receive the package. To ensure data accuracy, a lost packet or inaccurate package is usually restored by multiple retransmissions. The retransmissions, however, result in additional latency and energy consumption. From this point of view, it would be really helpful if more spectrum resources are available, especially under the condition that when the link quality on the extra spectrums are better than the original. It can not only improve the chance of transmission concurrency but also has the potential to reduce retransmission. Based on the above mentioned thought and motivations, it becomes an meaningful and interesting topic to investigate how cognitive radio technique ([11][41]) can contribute to transmissions in Probabilistic Wireless Networks(PWNs).

Data aggregation is one of the most important data gathering applications in wireless networks. It is a data gathering process interested in aggregated information rather than the raw data. Data transmitted in the network is summarized so that traffic and energy consumption can be reduced in an efficient way. It has been widely investigated in conventional wireless networks [35][36][37][38][39][40]. Cognitive radio has been considered one of the most important technologies in the next generation wireless networks. It has shown its advantage in accessing spectrum resources not belonging to the original network. Existing literature on data aggregation can be found in cognitive radio networks [113]. However, to the best of our knowledge, none of the existing literatures considered the data aggregation scheduling problem in the network model under investigation. In this paper, we focus on how cognitive radio capability can be used to accelerate the data aggregation process in PWNs. To be specific, we investigate a PWN where users in the network are equipped with cognitive radios. Wireless devices equipped with cognitive radios are enabled to dynamically adjust their operating transmitter parameters. Therefore, the cognitive radio enables a user to work on the default spectrum of the PWN as a regular radio. It can also hunt extra spectrum resources from the environment by adjusting its transmitter parameter accordingly. Our target is to evaluate how the cognitive radio can be used to improve the data aggregation process regarding time efficiency. Therefore, the minimum latency data aggregation scheduling problem in probabilistic wireless networks with cognitive radio capability is formulated and studied.

The contribution of this work can be summarized from the following three aspects:

- We considers a PWN coexisting with another wireless network. Users in the PWN can either work on a default spectrum or access an extra spectrum opportunistically for transmission opportunity. Network models have been clearly illustrated in Section 6.2.1 followed by the mathematically formalization of *the Minimum Latency Data Aggregation Scheduling Problem in Probabilistic wireless network with Cognitive Radio capability* (short as MLDAS-P-CR) problem. Brief illustration shows the formalized problem is NP-Hard.

- Considering the challenges and special characteristics of the MLDAS-P-CR problem under the defined network model, a two phase scheduling algorithm is proposed. The first phase is finding an efficient routing structure under lossy links and extra transmission opportunity. In the second phase, a dynamic scheduling algorithm is introduced which considers both the link quality and dynamic spectrum availability.

- Theoretical analysis and extensive simulations verify the efficiency of the proposed routing hierarchy and scheduling algorithm.

## 6.2 Network Model and Problem Definition

In this section, the network model and our problem under investigation are introduced in detail.

### 6.2.1 Probabilistic Network Model

We consider a wireless network deployed in an area coexisting with another wireless network. For simplicity, the network that we are focusing on is called the *Regular Wireless*

*Network* (RWN) and the other wireless network coexists in the same area is called the *Auxiliary Wireless Network* (AWN).

**Auxiliary Wireless Network**: Assume the AWN under consideration has $m$ Auxiliary Users (AUs) denoted by set $\mathbb{U}_a = \{U_1, U_2, ..., U_m\}$. Let $T_a$ and $I_a$ represent the transmission radius and interference radius of an AU, respectively. In the AWN, there is a spectrum $S_a$ available for AUs to conduct in-network communication. Assume the time in the AN is slotted with lenght $\tau$, which is long enough for the AUs to sense spectrum and transmit once. For a transmitter and a receiver pair in the AWN, at each time slot, they become active and decide to do a transmission with probability $\mathcal{P}_a$, and stay silent with probability $1 - \mathcal{P}_a$. Particularly, the AUs do not care about the usage of $S_a$ when they stay silent, as long as $S_a$ is available when they need to communicate or transmit.

**Regular Wireless Network**: The considering RWN consists of $n$ users. Let $\mathbb{U}_r = \{u_1, u_2, ..., u_n\}$ denote the set of Regular Users (RUs), among which a RU $u_b \in \mathbb{U}_r$ wants to get the aggregated information from the network. For $\forall u_i \in \mathbb{U}_r$, its transmission and interference radii are denoted by $T_r$ and $I_r$, respectively. There is a default working spectrum $S_r$ assigned to RUs on which they can carry out transmission. Each $u_i \in \mathbb{U}_r$ is equipped with a single, half-duplex cognitive radio, which is capable of accessing the default spectrum $S_r$ or adapting parameters to access spectrum holes in the deployed area (such as spectrum holes on $S_a$). Limited by the radio capability, a RU can either transmit or receive data (not both) on one spectrum from all directions at a specific time. Assume the time in the RWN is also slotted, where the length of the time slot $\tau$ is long enough for a user to sense spectrum and proceed a data transmission. In each time slot $t$, a user who has data to send can either operates on $S_r$ or access $S_a$ opportunistically by following the "sensing-before-transmission" rule. RUs have equal chance to operate on $S_r$. However, $S_a$ can be occupied by RUs if and only if no on-going transmissions in the AWN will be affected. Particularly, at a specific time $t$, when a sender $u_s$ wants to send a message to a receiver $u_r$, the probability for successful transmission denoted by $p_{sr}$ is a non-negative number less or equal to 1 due to the lossy link. The success probability between a sender and a receiver can be obtained by historical communication log or predicting from LQI (Link Quality Index)[114]. If $u_s$ has $p_{sr}$ chance to send a message to $u_r$ successfully in the ideal case (no interference and collision), then on average $u_s$ is expected to send $\frac{1}{p_{sr}}$ times to guarantee $u_r$ can receive the message successfully. For convenience, the RWN with lossy links is called a *Probabilistic Regular Wireless Network* (PRWN), and the success transmission probability on a link is called *success probability* for short.

*Assumptions:* We assume that the slotted time in both the AWN and the RWN are synchronized. The RUs can snoop $S_a$ to achieve the synchronization. The problem under unsynchronized time model will be studied in our future work. Besides the network model, the Unit Disk Graph (UDG) interference Model will be employed. Under this model, the transmission range and interference range of users can be modeled as disks with equal-radius, that is, $T_a = I_a$ and $T_r = I_r$. Furthermore, the success probability for link $\overrightarrow{u_s u_r}$ on $S_a$ is assumed to be at least as good as that on $S_r$ for simplicity. However, the proposed solution is not constrained by this assumption.

### 6.2.2 MLDAS-P-CR Problem

According to the network model and assumptions given in Section 6.2.1, the following definition are introduced to help us mathematically formalize the problem under investigation.

**Definition 6.2.1.** Collision Free Links on Spectrum $S$ *($\rightleftharpoons^S$): Let $\overrightarrow{u_s u_r}$ represent a directed link whose sender is $u_s$ and receiver is $u_r$. Given two links $\overrightarrow{u_{s_1} u_{r_1}}$ and $\overrightarrow{u_{s_2} u_{r_2}}$, if a transmission from $u_{s_2}$ to $u_{r_2}$ does not conflict with an ongoing transmission from $u_{s_1}$ to $u_{r_1}$ on the same spectrum $S$, they are considered as collision free links on $S$. The collision free relationship on $S$ is denoted by notation $\rightleftharpoons^S$. Then, $\overrightarrow{u_{s_1} u_{r_1}} \rightleftharpoons^S \overrightarrow{u_{s_2} u_{r_2}}$ means $\overrightarrow{u_{s_1} u_{r_1}}$ and $\overrightarrow{u_{s_2} u_{r_2}}$ are collision free links on $S$. Particularly, if $\overrightarrow{u_{s_1} u_{r_1}}$ is on spectrum $S$ and $\overrightarrow{u_{s_2} u_{r_2}}$ is on spectrum $S'$, then both $\overrightarrow{u_{s_1} u_{r_1}} \rightleftharpoons^S \overrightarrow{u_{s_2} u_{r_2}}$ and $\overrightarrow{u_{s_1} u_{r_1}} \rightleftharpoons^{S'} \overrightarrow{u_{s_2} u_{r_2}}$ are true.*

According to Definition 6.2.1, under the UDG interference model, $\overrightarrow{u_{s_1} u_{r_1}} \rightleftharpoons^{S_r} \overrightarrow{u_{s_2} u_{r_2}}$ should satisfy $\| u_{s_1} - u_{r_2} \| \geq T_r$ and $\| u_{s_2} - u_{r_1} \| \geq T_r$, where $\| x - y \|$ is the Euclidean distance between x and y. Furthermore, if $\overrightarrow{u_{s_1}u_{r_1}}$ and $\overrightarrow{U_sU_r}$ are active at the same time, $\overrightarrow{u_{s_1}u_{r_1}} \rightleftharpoons^{S_a} \overrightarrow{U_sU_r}$ holds when $\parallel u_{s_1} - U_r \parallel \geq T_r$, $\parallel U_s - u_{r_1} \parallel \geq T_a$. Thus, $\overrightarrow{u_{s_1}u_{r_1}} \rightleftharpoons^{S_a} \overrightarrow{u_{s_2}u_{r_2}}$ requires $\parallel u_{s_1} - u_{r_2} \parallel \geq T_r$ and $\parallel u_{s_2} - u_{r_1} \parallel \geq T_r$, and for all active $\overrightarrow{U_sU_r}$ in the AWN, $\overrightarrow{u_{s_1}u_{r_1}} \rightleftharpoons^{S_a} \overrightarrow{U_sU_r}$ and $\overrightarrow{u_{s_2}u_{r_2}} \rightleftharpoons^{S_a} \overrightarrow{U_sU_r}$.

The Minimum Latency Data Aggregation Scheduling Problem in Probabilistic wireless network with Cognitive Radio capability (short as MLDAS-P-CR) can be formalized as follows: Given a PRWN $G_r = \{\mathbb{U}_r, \mathbb{E}_r, \mathbb{P}(\mathbb{E}_r)\}$ and an AWN $G_a = \{\mathbb{U}_a, \mathbb{E}_a\}$. $\mathbb{U}_r$ and $\mathbb{E}_r$ are the vertices (set of RUs) and edges (lossy links among RUs) in the PRWN. $\mathbb{P}(\mathbb{E}_r)$ denotes the set of success probability on all the lossy links. $\mathbb{U}_a$ and $\mathbb{E}_a$ are the set of AUs and links among AUs in the AWN. For a link $\overrightarrow{u_su_r} \in \mathbb{E}_r$, $\mathbb{P}(\overrightarrow{u_su_r})$ is the probability that a transmission from $u_s$ to $u_r$ can be success. $\mathbb{U}_a$ and $\mathbb{E}_a$ are the set of AUs and links among AUs in the AWN, respectively. In the PRWN, a user $u_b \in \mathbb{U}_r$ wants to collect aggregated information from the network. For simplicity, let $\mathbb{D} = \{d_1, d_2, ..., d_n\}$ denote the set of data packages generated in the PRWN, where $d_i (i \neq b)$ is the data observed by user $u_i$. The objective of MLDAS-P-CR is to find a schedule plan $\mathbb{S} = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_L\}$ who requires the minimum transmission delay $L$, where each $\mathbb{S}_t$ $(1 \leq t \leq \mathbb{L})$ is a set of collision-free links who are scheduled at time slot $t$. We can think in the way that $\mathbb{S}_t$ provids a collision free mapping from a sender set $\mathbb{S}_t^s$ to a receiver set $\mathbb{S}_t^r$. Let $\mathcal{A}(\bullet)$ denotes the aggregation function, where $\mathcal{A}(\mathbb{S}_t)$ is the set of aggregation results at time $t$. Let $X_{u_i,t}^{C,\mathcal{S}}$ and $X_{u_i,t}^{C,\mathcal{R}}$ be variables indicating that a transmission is a success transmission when the RU $u_i$ is active as a sender or receiver on spectrum $C$ at time $t$, separately. $X_{u_i,t}^{C,\mathcal{S}} = 1$ when $u_i$ is active as a sender and the transmission is success, 0, otherwise. Similarly, $X_{u_i,t}^{C,\mathcal{R}} = 1$ when $u_i$ is active as a receiver of a success transmission, otherwise, $X_{u_i,t}^{C,\mathcal{R}} = 0$. The MLDAS-P-CR problem can be formalized as follows:

- Objective: $\arg_{\mathbb{S}=\{\mathbb{S}_1,\mathbb{S}_2,...,\mathbb{S}_{\mathbb{L}}\}} \min L$
- Constraints: Given $0 < t \leq L$ and $C \in \{S_a, S_r\}$

  1. $\forall\ \overrightarrow{u_{s_1}u_{r_1}} \in \mathbb{S}_t$ and $\overrightarrow{u_{s_2}u_{r_2}} \in \mathbb{S}_t$, $\overrightarrow{u_{s_1}u_{r_1}} \rightleftharpoons^{C} \overrightarrow{u_{s_2}u_{r_2}}$ is true.

2. $\forall\ \overrightarrow{u_s u_r} \in \mathbb{S}_t$, if $\overrightarrow{U_{s'} U_{r'}}$ ($U_{s'} \in \mathbb{U}_a$, $U_{r'} \in \mathbb{U}_a$), if $U_{s'}$ and $U_{r'}$ are active at $t$, then $\overrightarrow{u_s u_r} \rightleftharpoons^{S_a} \overrightarrow{U_{s'} U_{r'}}$.
3. $\forall u_i \in \mathbb{U}_r$, $X_{u_i,t}^{C,\mathcal{S}} + X_{u_i,t}^{C,\mathcal{R}} \leq 1$, where $X_{u_i,t}^{C,\mathcal{S}} \in \{0,1\}$, $X_{u_i,t}^{C,\mathcal{R}} \in \{0,1\}$.
4. $\forall u_i \in \{\mathbb{U}_r - u_b\}$, $\sum_{t=1}^{L} X_{u_i,t}^{C,\mathcal{S}} = 1$.
5. $\mathcal{A}(\mathbb{S}_L) = \mathcal{A}(\mathbb{D})$.
6. $X_{u_b,L}^{C,\mathcal{R}} = 1$, $\sum_{i=1}^{n} X_{u_i,L}^{C,\mathcal{S}} = 1$ and $\sum_{i=1,i\neq b}^{n} X_{u_i,L}^{C,\mathcal{R}} = 0$.

Constraint 1) shows that the links scheduled at time $t$ should be collision-free. Since RUs may access $S_a$ or $S_r$, the schedule should be collision-free on both $S_a$ and $S_r$. Constraint 2) declares that for all the links scheduled on $S_a$ at $t$, they should have no collision to on-going links at $t$ in the AWN. Constraint 3) specifies the radio constraint that a RU can only be a sender or a receiver or stay silent at $t$. Constraint 4) limits that a RU is required to send aggregated result only once for the purpose of reducing traffic and energy consumption. Particularly, even though the objective of MLDAS-P-CR is to minimize the transmission latency, Constraint 4) demonstrates that a minimum latency scheduling plan with optimal energy consumption is desirable. Data generated in the PRWN will be aggregated along the scheduling. The aggregated result should match the global result which will be received by $u_b$ at the last step. Those constraints are demonstrated in Constrains 5) and 6).

Obviously, the MLDAS-P-CR problem is NP-hard. The conclusion comes from the fact that the Minimum Latency Data Aggregation Scheduling (MLDAS) problem in wireless network is NP-hard. Considering a special case of our MLDAS-P-CR model where the AUs' density and activity in the AWN is so high that no RU has the opportunity to operate on $S_a$ at all. Furthermore, assuming the link quality and physical environment are so wonderful that as long as two RUs within each other's transmission range, transmissions can always be conducted successfully when the interference condition is ideal (that is, the probability for successful transmission is fixed as 1), which is the MLDAS problem investigated in the traditional wireless networks. From this point of view, the MLDAS problem in traditional wireless network can be considered as a special case of MLDAS-P-CR problem. Therefore, we can conclude that MLDAS-P-CR problem is NP-hard.

## 6.3 Scheduling for MLDAS-P-CR

In this section, how to get an efficient scheduling plan for our proposed MLDSA-P-CR problem is presented. The solution exploring process can be divided into two phases, where the first phase is to find a routing structure which can better contribute to the link scheduling algorithm in the second phase.

### 6.3.1 BPT Construction

According to the existing literature on data aggregation scheduling, tree-based structures have been widely used as the routing structure. Among which, one special type of trees which are based on *Connected Dominating Set (CDS)* has been widely used due to its potential of high concurrency. Given a graph $G = \{V, E\}$, a *Dominating Set (DS)* denoted as $V_D$ has the property that $\forall v \in V$, either $v \in V_D$ or $\mathbb{N}_1(v) \in V_D$ (where $\mathbb{N}_1(v)$ is $v$'s neighbor set within one-hop). A CDS is a connected DS. Former researchers applied different strategies to obtain a CDS-based routing tree. The category that derived from *Maximal Independent Set* is very popular due to its efficiency and nice geographical property. However, solutions for deterministic network model cannot be intuitively applied to the probabilistic network model. For the MLDAS-P-CR problem, on the one hand, we want to employ a routing structure with maximized concurrence. On the other hand, due to the lossy links, we need to consider the probability of success transmission. If the success probability on link $\overrightarrow{u_s u_r}$ is 0.2, the expected number of transmission is $\frac{1}{0.2} = 5$, that means we may need to find five or more time slots to schedule link $\overrightarrow{u_s u_r}$. Therefore, an arbitrary routing tree with lots of "bad" links may result in a long delay.

Before we go to details of the routing tree construction process, two more definitions are presented for simplicity.

**Definition 6.3.1.** One-Hop Neighbor Set *(*$\mathbb{N}_1(\bullet)$*) For a user $u$ in a wireless network, the One-Hop Neighbor set of $u$ is the set of users that can be reached by $u$ directly (with* $1-1=0$

*non-repetitive intermediate user). u's One-Hop Neighbor set is denoted by* $\mathbb{N}_1(u)$*. Similarly,* $\mathbb{N}_h(\bullet)$ *is defined as the set of u's* $h$-Hop Neighbor Set*, where* $\mathbb{N}_h(u)$ *is the set of users that can be reached by u through* $h-1$ *non-repetitive intermediate users and finally reach users in* $\mathbb{N}_h(u)$ *that are h hop away from u.*

Under the network model discussed in Section 6.2.1, $\forall u_i \in \mathbb{E}_r$, $\mathbb{N}_1(u_i) = \{u_j | u_j \in \mathbb{E}_r, \parallel u_i - u_j \parallel \leq T_r\}$, and $\mathbb{N}_h(u_i) = \{u_j | u_j \in \mathbb{E}_r, u_k \in \mathbb{N}_{h-1}(u_i), \parallel u_k - u_j \parallel \leq T_r\}$.

**Definition 6.3.2.** *Average Transmission Delay* ($\widehat{d(\bullet)}$) *Given a RU* $u_s \in \mathbb{E}_r$*, the average transmission delay of* $u_s$ *(denoted by* $\widehat{d(u_s)}$*) is the expected delay for a transmission from* $u_s$ *to one of the RU in* $\mathbb{N}_1(u_s)$*. Let* $p_{sr}$ *represent the success probability of transmission* $\overrightarrow{u_s u_r}$*,* $\widehat{p_a}$ *denotes the average availability of spectrum* $S_a$ *for transmission* $u_s$ *to any RU in* $\mathbb{N}_1(u_s)$*. Then* $\widehat{d(u_s)} = \frac{\sum_{\forall u_r \in \mathbb{N}_1(u_s)} \frac{1}{p_{sr}(1+\widehat{p_a})}}{|\mathbb{N}_1(u_s)|}$*, where* $\mid \mathbb{N}_1(u_s) \mid$ *is the cardinality of* $\mathbb{N}_1(u_s)$*.*

Based on the defined definitions, we propose a CDS-based *Balanced Probabilistic routing Tree (BPT)* considering *Balanced* workload allocation and *Probabilistic* transmissions. The process of finding the BPT can be divided into three stages (as shown in Alg. 9). In the first stage (line $1-6$), a DS denoted as set $\mathbb{D}$ is selected according to RUs' average transmission delay. Initially, $u_b$ is added to $\mathbb{D}$. Then all other RUs are sorted in non-decreasing order based on their average transmission delay. After that, RUs are iteratively selected as a user in $\mathbb{D}$ when they are the node with the highest priority and have no one hop neighbor in $\mathbb{D}$. The disjoint $\mathbb{D}$ provides the potential of concurrent link scheduling, where RUs in $\mathbb{D}$ are called *dominators*. In order to proceed data transmission in the PRWN, in the second stage (line $7-17$), RUs called *connectors* are selected to make the dominators obtained in the first stage a connected component (CDS). To achieve this goal, each RU in $\mathbb{D}$ is initialized as a single component. Subsequently, RUs not in $\mathbb{D}$ are sorted in non-decreasing order based on their average transmission delay. Since connectors mainly perform as a bridge among dominators, the average transmission delay is calculated only based on its one hop dominator neighbor set. The order is used as the priority in the following selection process. Starting from the base station, a RU is added to the connector set $\mathbb{C}$ if it has the highest priority and it connects at least two separate components. Once a connector is confirmed, the separate components that connected by the connector join each other and become a bigger component. This process terminates when there is only one component (CDS). At the end of this process, we actually get a tree whose root is $u_b$ and contains all dominators and connectors. In the last stage (line $18-21$), the remaining RUs referred as *dominatees* $\mathbb{W}$ are added to the BPT with consideration of a balanced workload allocation. Here the workload indicates the total expected transmission delay needed for data transmissions. The 2-norm is used to be the metric that evaluates the balance of workload allocation. A dominatee will find a dominator parent that leads to the minimum increase on the 2-norm of the total expected transmission delay. If a RU $u_i$ chooses $u_j \in \mathbb{D}$ as its parent, the expected increase of $u_j$'s transmission delay is estimated by $\frac{1}{p_{u_i u_j}(1+\widehat{p'_a})}$, where $p_{u_i D_j}$ is the successful transmission probability between $u_i$ and $u_j$, and $\widehat{p'_a}$ is the estimated spectrum availability between them on $S_a$.

According to the generation process, the following properties can be derived from the BPT:

- If the root of BPT is considered as in layer 1, then all the dominators are in the odd layers, all the connectors are in the even layers, and all the dominatee are the leaves of the tree.

- Based on [40], for a dominator $u_d \in \mathbb{D}$, $|\mathbb{N}_1(u_d) \bigcap \mathbb{D}| \leq 5$ and $|\mathbb{N}_1(u_d) \bigcap \mathbb{C}| \leq 21$. That is, a dominator has a maximum of 5 one-hop neighbors who are also dominators, and no more than 21 neighbors within one-hop of a dominator can be connectors.

- Since RUs are allowed to access $S_a$, for a $u_i \in \mathbb{U}_r$, $\widehat{p_a}$ can be obtained through historical detection or roughly estimated by $(1-\mathcal{P}_a)^{|\mathbb{N}_1(u_i) \bigcap \mathbb{U}_a|}$, where $|\mathbb{N}_1(u_i) \bigcap \mathbb{U}_a|$ is the number of AUs within $u_i$'s one-hop neighborhood (this estimation is used in the simulation).

**Algorithm 9:** Construction Alg. for BPT (Balanced Probabilistic routing Tree)

**input** : $G_r = \{\mathbb{U}_r, \mathbb{E}_r, \mathbb{P}(\mathbb{E}_r)\}$,$G_a = \{\mathbb{U}_a, \mathbb{E}_a\}$

**output**: $G_r = \{\mathbb{U}_r, \mathbb{E}_r, \mathbb{E}_r^T, \mathbb{P}(\mathbb{E}_r)\}$, where $\mathbb{E}_r^T$ is the set of links on the BPT

$\mathbb{V} = \{\mathbb{U}_r - u_b\}$, $\mathbb{D} = \emptyset$;

For all $u_i \in \mathbb{V}$, sort in non-decreasing order based on $\widehat{d(u_i)} = \frac{\sum_{\forall u_r \in \mathbb{N}_1(u_i)} \frac{1}{p_{ir}(1+\widehat{p_a})}}{|\mathbb{N}_1(u_i)|}$;

Add $u_b$ to $\mathbb{V}$ as the first element;

**while** $\mathbb{V} \neq \emptyset$ **do**

    $\mathbb{D} = \mathbb{D} \bigcup \{u_f\}$, where $u_f$ is the first element in $\mathbb{V}$;

    $\mathbb{V} = \mathbb{V} - \{u_f\} - \mathbb{N}_1(u_f)$;

$\mathbb{V} = \mathbb{U}_r - \mathbb{D}$, $\mathbb{D} = \{D_1, D_2, ...\}$, $\mathbb{C} = \emptyset$;

For all $u_i \in \mathbb{V}$, sort in non-decreasing order based on $\widehat{d(u_i)} = \frac{\sum_{\forall u_r \in \mathbb{D} \bigcap \mathbb{N}_1(u_i)} \frac{1}{p_{ir}(1+\widehat{p_a})}}{|\mathbb{N}_1(u_i)|}$;

**for** $u_d \in \mathbb{D}$: *d from* 1 *to* ... **do**

    **for** $u_i$ *in the sorted list of* $\mathbb{V}$ **do**

        **if** $u_i \in \mathbb{N}_1(u_d)$ *and* $u_i.marked = NO$ **then**

            **if** $u_i \in \mathbb{N}_1(\mathbb{D}')$ *where* $u_d \in \mathbb{D}'$ *and* $|\mathbb{D}'| \geq 2$ *and* $\mathbb{D}'.group \geq 2$ **then**

                $\mathbb{C} = \mathbb{C} \bigcup \{u_i\}$, $\mathbb{V} = \mathbb{V} - \{u_i\}$;

                $parent(u_i) = u_d$, $u_i.marked = YES$;

                **for** $\forall u_{d'} \in \mathbb{D}'$ **do**

                    $parent(u_{d'}) = u_i$;

                $\forall u_{d'} \in \mathbb{D}'$, Merge into one group;

$\mathbb{W} = \mathbb{V} - \mathbb{D} - \mathbb{C}$, $w_1 = w_2 = ... = w_{|\mathbb{D}|} = 0$;

**for** *each* $u_i \in \mathbb{W}$ **do**

    **if** $D_j \in \mathbb{D} \bigcap \mathbb{N}_1(u_i)$ *and* $D_j$ *brings minimum increase of* $\sqrt{\sum_{k=1}^{|\mathbb{D}|} w_k^2}$, *where* $w_j += \frac{1}{p_{u_i D_j}(1+\widehat{p'_a})}$ **then**

        Add $u_i$ to BPT where $parent(u_i) = D_j$;

### 6.3.2 Scheduling Algorithm for MLDAS-P-CR

In this subsection, we introduce the scheduling algorithm for MLDAS-P-CR based on the constructed routing tree BPT. Our objective is to schedule all the links on the BPT so that aggregated result can be sent to $u_b$ with the shortest latency. Since packages are aggregated on the intermediate nodes, so only nodes that have no children to wait or have received packages from all its children are allowed to transmit. Furthermore, $S_r$ is the default spectrum that is always available while $S_a$ is only available opportunistically. With all those concerns, we propose a two step scheduling algorithm. In the first step, only $S_r$ is concerned to derive an initiatory scheduling plan. In the second step, the final scheduling plan which takes $S_r$, $S_a$, and the lossy links into consideration is obtained based on the initiatory scheduling plan found in the first step.

**Initiatory Scheduling** For convenience, we classify the links on the BPT denoted as set $\mathbb{E}_r^T$ into three categories: $\mathbb{L}_{wd}$, $\mathbb{L}_{dc}$ and $\mathbb{L}_{cd}$. $\mathbb{L}_{wd}$ contains all the links $\overrightarrow{u_w u_d} \in \mathbb{E}_r^T$ whose senders are dominatees ($u_w \in \mathbb{W}$) and corresponding receivers are dominators ($u_d \in \mathbb{D}$). $\mathbb{L}_{dc}$ are the links $\overrightarrow{u_d u_c} \in \mathbb{E}_r^T$ that sender $u_d \in \mathbb{D}$ and receiver $u_c \in \mathbb{C}$, and $\mathbb{L}_{cd}$ is the set of links with sender $u_c \in \mathbb{C}$ and receiver $u_d \in \mathbb{D}$, respectively. Furthermore, a link $\overrightarrow{u_s u_r}$ is said *READY* to be scheduled on the BPT when $u_s$ has no children to wait.

**Definition 6.3.3.** Total Expected Waiting Delay $(\widehat{d_t(u_p)})$ *Given a parent node $u_p$ on the tree, let $Children(u_p)$ denote the set of children $u_p$ has on the BPT, we define the* Total Expected Waiting Delay *of $u_p$ on the BPT as* $\widehat{d_t(u_p)} = \sum_{u_c \in Children(u_p)} \frac{1}{p_{pc}(1+\widehat{p_a})}$.

Alg. 10 shows how to derive an initiatory scheduling plan based on the BPT. The scheduling plan is generated iteratively until all the links in $\mathbb{E}_r^T$ have been scheduled (line 2). In each iteration, links in $\mathbb{L}_{wd}$ have the highest priority to schedule since they are the leaves of the tree so that always READY (line $3-4$). If it is an odd iteration, READY $\mathbb{L}_{dc}$ has higher priority to be scheduled than that of links in READY $\mathbb{L}_{cd}$ (line $5-7$), otherwise, the priority is reversed (line $8-10$). The scheduling within each link set ($\mathbb{L}_{wd}$, $\mathbb{L}_{dc}$ or $\mathbb{L}_{cd}$) is a simple greedy call based on the $\widehat{d_t(\bullet)}$ of the receiver, the details are shown in Alg. 11.

**Algorithm 10:** Initiatory Scheduling Plan

**input** : $G_r = \{\mathbb{U}_r, \mathbb{E}_r, \mathbb{E}_r^T, \mathbb{P}(\mathbb{E}_r)\}$, $\mathbb{L}_{wd}$, $\mathbb{L}_{dc}$ and $\mathbb{L}_{cd}$
**output**: $\mathbb{S}^I = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_{L'}\}$

```
t = 1, S_1 = S_2 = ... = S_L' = ∅;
while L_wd ⋃ L_dc ⋃ L_cd ≠ ∅ do
    if L_wd ≠ ∅ then
        setScheduling(L_wd, E_r^T);
    if t%2! = 0 then
        if L_dc ≠ ∅ then
            setScheduling(L_dc, E_r^T);
        if L_cd ≠ ∅ then
            setScheduling(L_cd, E_r^T);
    if t%2 == 0, L_dc ≠ ∅ and L_cd ≠ ∅ then
        if L_cd ≠ ∅ then
            setScheduling(L_cd, E_r^T);
        if L_dc ≠ ∅ then
            setScheduling(L_dc, E_r^T);
    t + +;
```

**Algorithm 11:** Initiatory Link Set Scheduling on $\mathbb{E}_r^T$-$setScheduling(\mathbb{L}, \mathbb{E}_r^T)$

**input** : $G_r = \{\mathbb{U}_r, \mathbb{E}_r, \mathbb{E}_r^T, \mathbb{P}(\mathbb{E}_r)\}$, $\mathbb{S}_t$, $\mathbb{L}$
**output**: $\mathbb{S}_t$

```
Sort L in non-descending order based on d_t(•)^ denoted by L' = {l_1, l_2, ...};
for each link l_i ∈ L' do
    if l_i ⇌^{S_r} S_t then
        S_t = S_t ⋃ l_i;
        Remove l_i from L and E_r^T;
```

The initiatory scheduling derived above ignores both lossy links and transmission opportunities on spectrum $S_a$. Those two factors are the major concerns for the final scheduling. The concurrency of the link set obtained in the initiatory scheduling has a pretty high chance to be destroyed due to lossy links. Links that can be scheduled at one time slot based on initiatory scheduling may need different time slots to conduct successful transmissions. Because different lossy links have different success probability.

**Final Scheduling** To solve the problems discussed above, we first try to get a collision free scheduling plan considering lossy links. After that, a final dynamic scheduling algorithm is introduced which takes the extra spectrum resource into consideration.

The following definitions are given to simplify our illustration.

**Definition 6.3.4.** Longest Expected Transmission Delay *($\widehat{d_{max}(\bullet)}$) Given a set of links $\mathbb{L}$, the longest expected transmission delay is defined as the expected transmission delay of link $\forall l \in \mathbb{L}$ whose delay calculated as $\frac{1}{p_l(1+p_a^{min})}$ is the biggest. That is, $\widehat{d_{max}(\mathbb{L})} = max(\frac{1}{p_l(1+p_a^{min})})$, where $p_l$ is the transmission success probability of link $l$, and $p_a$ is the access probability on $S_a$.*

**Definition 6.3.5.** Collision Link Set *($\mathbb{CLS}(l)$) Given a link $l$, the collision free link set of $l$ on $E$ is defined as the set of links on $E$ that has no interference with $l$ if they conduct transmissions together. That is, $\mathbb{CFSL}(l) = \{l' | \forall l' \in E, l \rightleftharpoons l'\}$. Then the collision link set is the set that will cause collision when they are scheduled simultaneously, that is, $\mathbb{CLS}(l) = \mathbb{CFSL}^C(l)$, where the superscript $C$ indicates the complement set.*

According to the definitions, RUs are ranked as follows: for a RU $u_s$, its rank is a time interval $[t, t+\Delta t]$, where the rank of $u_s$ indicates that $u_s$ is expected to finish its transmission during the time interval between $t$ to $t + \Delta t$, when the lossy links are considered in the network. The ranking process is pretty lightweight: start from the first collision free link set $\mathbb{S}_1$, ranking all the senders based on $\widehat{d_{max}(\bullet)}$, in the end, all the senders are ranked as $[1, \widehat{d_{max}(\mathbb{S}_1)}]$, which indicates that on average, all the links in $\mathbb{S}_1$ should be scheduled during this time interval. For the following sets $\mathbb{S}_2$ to $\mathbb{S}_{L'}$, all the ranking interval starts from the previous end time plus one, and the end time is the accumulated interval plus its own longest expected transmission delay. This process is roughly shown in Alg. 12. The idea of ranking provides us with a way to obtain concurrent sets with lossy links.

**Algorithm 12:** Ranking-$Ranking(\mathbb{U}_r - u_b)$

**input** : $\mathbb{S}^I = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_{L'}\}$, $\mathbb{E}_r^T$

**output**: $Ranking(\mathbb{U}_r - u_b) = \{rank(u_1), rank(u_2), ...\}$,

**for** *each link* $\overrightarrow{u_s u_r} \in \mathbb{S}_1$ **do**

  $rank(u_s) = [1, \widehat{d_{max}(\mathbb{S}_i)}]$;

**for** $i = 2; i \leq L'; i++$ **do**

  **for** *each link* $\overrightarrow{u_s u_r} \in \mathbb{S}_i$ **do**

    $rank(u_s) = [\sum_{j=1}^{i-1} \widehat{d_{max}(\mathbb{S}_i)} + 1, \sum_{j=1}^{i} \widehat{d_{max}(\mathbb{S}_i)}]$;

Based on the ranking, our dynamic scheduling algorithm is proposed (as shown in Alg. 13). For a $READY$ $u_s \in \overrightarrow{u_s u_r}$ who has not finished transmission, it checks if it is within its default scheduling interval based on $rank(u_s)$ derived from Alg. 12. If the answer is *YES*, it senses spectrums. Transmission preference is given to $S_a$ whenever it is available to leave more chance to remaining RUs on $S_r$. Otherwise, if it is not the expected scheduling time for $u_s$, $u_s$ waits for a very short period $\epsilon \bullet \Lambda$, where $\epsilon$ is a very small positive number and $\Lambda$ is the cardinality of $\mathbb{CLS}(\overrightarrow{u_s u_r})$. After that, $u_s$ detects the availability and contention status on both $S_a$ and $S_r$, a transmission decision is made only when the spectrum is available and no collision with on-going transmissions on the spectrum. Let $\mathbb{CLS}_w(\overrightarrow{u_s u_r})$, $\mathbb{CLS}_d(\overrightarrow{u_s u_r})$ and $\mathbb{CLS}_c(\overrightarrow{u_s u_r})$ be the collision link set of link $\overrightarrow{u_s u_r}$ whose senders are dominatee, dominator and connector, respectively. For a dominatee, $\Lambda$ is a random number in $[0, |\mathbb{CLS}_w(\overrightarrow{u_s u_r})|]$. In an even iteration, $\Lambda$ is a random number in $[|\mathbb{CLS}_w(\overrightarrow{u_s u_r})|+1, |\mathbb{CLS}_w(\overrightarrow{u_s u_r})|+|\mathbb{CLS}_c(\overrightarrow{u_s u_r})|]$ for a connector and $[|\mathbb{CLS}_w(\overrightarrow{u_s u_r})|+|\mathbb{CLS}_c(\overrightarrow{u_s u_r})|+1, |\mathbb{CLS}_w(\overrightarrow{u_s u_r})|+|\mathbb{CLS}_c(\overrightarrow{u_s u_r})|+|\mathbb{CLS}_d(\overrightarrow{u_s u_r})|]$ for a dominator. Otherwise, $\Lambda$ for a dominator and a connector will be swapped.

The implementation of Alg. 13 is based on an assumption illustrated below. First, before each transmission, a sender and a receiver may need to exchange a short becon

**Algorithm 13:** Scheduling for MLDAS-P-CR

**input** : $G_r = \{\mathbb{U}_r, \mathbb{E}_r, \mathbb{E}_r^T, \mathbb{P}(\mathbb{E}_r)\}$, $G_a = \{\mathbb{U}_a, \mathbb{E}_a\}$
**output**: $\mathbb{S} = \{\mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_L\}$

$t = 1$;
**while** $u_s \in \overrightarrow{u_s u_r}$ *is READY but has not finished transmission successfully* **do**
  **if** *$t$ is within $rank(u_s)$* **then**
    **if** *$S_a$ available* **then**
      Transmit on $S_a$;
    **else**
      Transmit on $S_r$;
  **else**
    Random wait for $\epsilon \bullet \Lambda$;
    **if** *$S_a$ available and ACTIVE $\mathbb{CLS}(\overrightarrow{u_s u_r}) = \emptyset$* **then**
      Transmit on $S_a$;
    **else if** *ACTIVE $\mathbb{CLS}(\overrightarrow{u_s u_r}) = \emptyset$ on $S_r$* **then**
      Transmit on $S_r$;
  Update $\mathbb{CLS}(\overrightarrow{u_s u_r})$, $t++$;

message once to confirm the transmission decision. For example, a sender and a receiver need to confirm the availability of $S_a$ before they decide to transmit on $S_a$, or they need to confirm the collision status with ongoing transmissions in either network. Second, after each transmission, RUs are required to broadcast a short message to notice their neighbors and competitors when the transmission is successful. These can be archived by exchanging different short BEACON message. According to [18], lossy links in wireless networks can be modeled as leaky-pipes. For common data package with important information, the accuracy may reduce during the transmission. Therefore, from the point of precision, retransmission are required if data package is not received correctly. However, for simplicity, we assume short BEACON messages can be received successfully. Without this assumption, false alarm and miss detection will occur during spectrum sensing. A better way to deal with this issue will be left to future work.

**Definition 6.3.6.** Lowest Success Probability *($p_0$) Given a BPT and $p_0$ ($0 < p_0 \leq 1$), the lowest success probability is the minimum success transmission probability of links on the*

*BPT. That is, for any $\overrightarrow{u_s u_r}$ on the BPT, $p_{\overrightarrow{u_s u_r}} \geq p_0$.*

**Lemma 12.** *The expected number of spectrums available to a link in the PRWN is $1 + (1 - \mathcal{P}_a)^{\pi(T_a^2+T_r^2)\frac{m}{\mathcal{A}}}$, where $T_r$ and $T_a$ are the transmission radius for users in the PRWN and AWN, respectively.*

*Proof.* For a link $\overrightarrow{u_s u_r}$ in the PRWN, $S_a$ is available if and only if all ongoing transmissions in the AWN are collision free links. That is, no AU in $u_s$'s transmission range $T_r$ active as a receiver, and $u_r$ is not in the range of any active AU sender's transmission range $T_a$. Assume the considered wireless networks are deployed in a square area with area $\mathcal{A}$, then, the expected number of AU in an unit area is $\frac{m}{\mathcal{A}}$. Thus, the probability that $S_a$ is available to $\overrightarrow{u_s u_r}$ is $(1 - \mathcal{P}_a)^{\pi(T_a^2+T_r^2)\frac{m}{\mathcal{A}}}$. Since $S_r$ is always available to $\overrightarrow{u_s u_r}$, the expected number of spectrums available to a link in the PRWN is $1 + (1 - \mathcal{P}_a)^{\pi(T_a^2+T_r^2)\frac{m}{\mathcal{A}}}$. $\square$

**Theorem 3.** *The expected latency of the proposed scheduling algorithm is upper bounded by $\frac{5\Delta+44D+1}{p_0 * [1+(1-\mathcal{P}_a)^{\pi(T_a^2+T_r^2)\frac{m}{\mathcal{A}}}]}$, where $\Delta$ and $D$ are the maximum degree and diameter of the PRWN, respectively.*

*Proof.* Based on the final scheduling, all unfinished READY RUs transmit either on $S_a$ or $S_r$ if they are expected to. For the RUs who are not expected to transmit, they have a chance to try later. In the worst case, only the expected RUs transmit at each time slot. That is, RUs only follow the initial scheduling plan and ranking process to transmit data. According to the initial scheduling plan, in the worst case, all dominatees will be scheduled first, and then, dominators and connectors are scheduled to transmit alternatively. From [115], we can derive that the lower bound for the expected latency from the initiatory scheduling plan is $5\Delta + 44D + 1$, where $\Delta$ is the maximum degree and $D$ is the diameter of the PRWN. Considering the lossy links, in the worst case, all transmissions need to be re-do for at most $\frac{1}{p_0}$ times to guarantee success and correct reception. Together with observation derived from Lemma 12, we can get the conclusion that the expected latency of the proposed scheduling algorithm is upper bounded by $\frac{5\Delta+44D+1}{p_0 * [1+(1-\mathcal{P}_a)^{\pi(T_a^2+T_r^2)\frac{m}{\mathcal{A}}}]}$. $\square$

## 6.4 Performance Evaluation

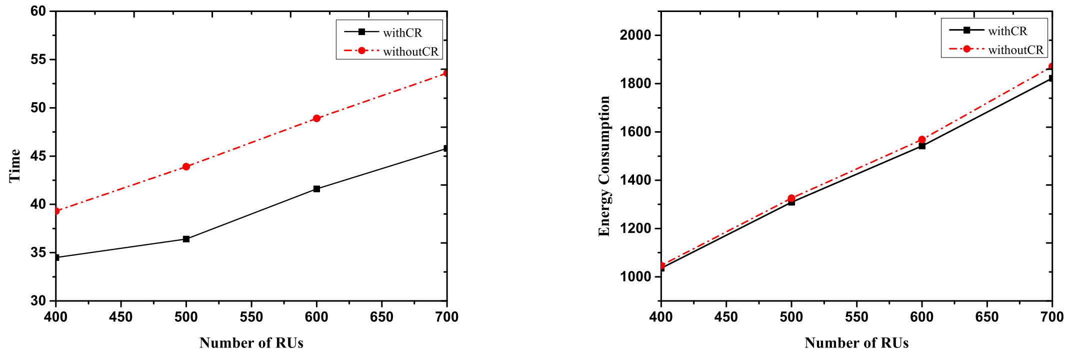
### 6.4.1 Setup

In this section, the effectiveness of the proposed solution is verified through numerical simulations, where the simulation results are obtained from Visual studio 2012 and C++ language. Results from all aspects show that the cognitive capability can reduce delay by about 10% to 20%, while consuming almost the same or even less energy. All the results shown below are the average results over 50 trails. For simplicity, $numRU$ and $numAU$ represent the number of RUs and AUs, respectively. $transAU$ and $transRU$ are the transmission radius for AUs and RUs. $actP$ is the active probability of AUs. $S_a$ and $S_r$ are the lowest success probability of the AWN and the PRWN, which indicates the lowest probability of success transmission. $withCR$ is the proposed scheduling algorithm. Since no existing work focuses on the same topic, we use $withouCR$ as the comparison algorithm, which is a modified version of $withCR$ by ignoring the transmission opportunities on $S_a$.

### 6.4.2 Results and Analysis

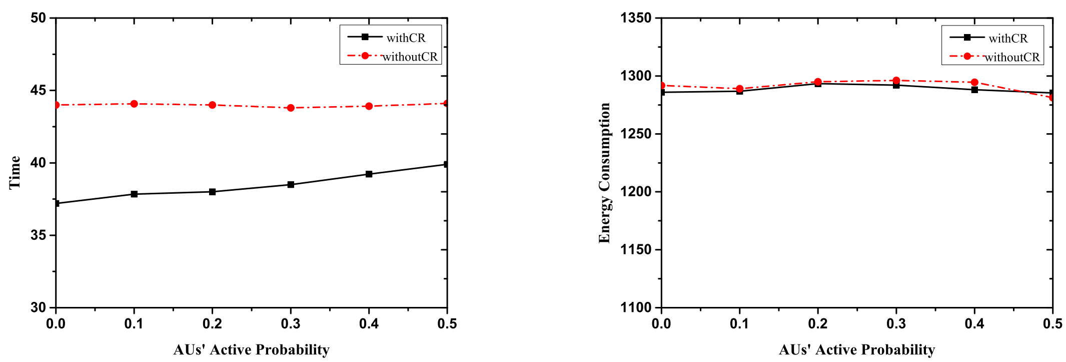
First, the influence of RUs' population on our scheduling algorithm has been evaluated. From the results showed in Fig. 6.1, we can see that because of the increase of RUs' population, competition among RUs becomes more serious. For a sender and a receiver pair, the more RUs, the more competitions coming from collision with, which results in the situation that both latency and energy consumption increase with the increase of RUs' population. However, the extra spectrum opportunity brought by the cognitive radio reduces latency (Fig. 6.1(a)), in the meanwhile, does not introduce extra energy consumption (Fig. 6.1(b)).

Second, we verify the impact of AUs' active probability on the time and energy efficiency. The results are shown in Fig. 6.2. Since $withoutCR$ ignores the existence of $S_a$, the change of AUs' active probability should not affect its performance. As shown in Fig. 6.2(a) and 6.2(b), the results for $withoutCR$ are quiet stable as we expected. With the increase of AUs'



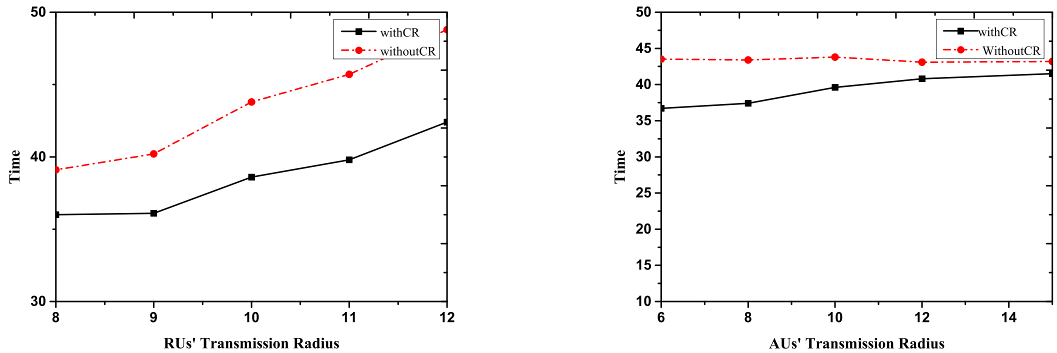

(a) Time efficiency (b) Energy efficiency

Figure 6.1. Impact of RUs' population. ($numAU = 100$, $S_a = S_r = 0.5$, $transAU = 10$, $transRU = 10$, $actP = 0.3$, $area = 80 * 80$.)




(a) Time efficiency (b) Energy efficiency

Figure 6.2. Impact of AUs' active probability. ($numAU = 100$, $numRU = 500$, $S_a = S_r = 0.5$, $transAU = 10$, $transRU = 10$, $area = 80 * 80$.)

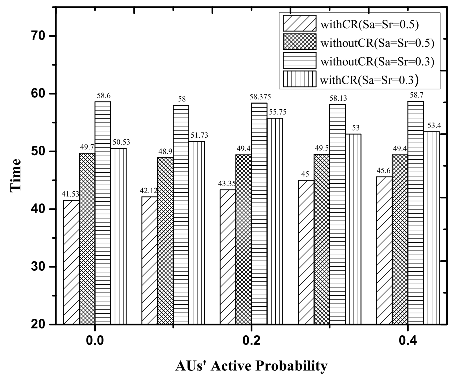



(a) Time efficiency ($transAU = 10$) (b) Time efficiency ($transRU = 10$)

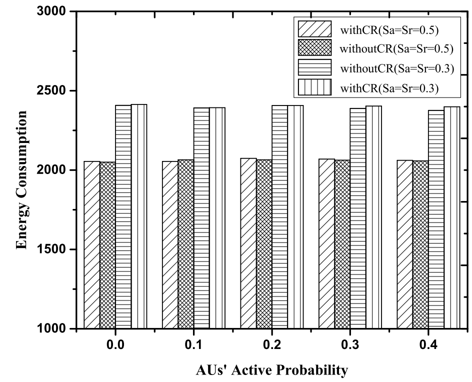
Figure 6.3. Impact of transmission radius. ($numAU = 100$, $numRU = 500$, $S_a = S_r = 0.5$, $transAU = 10$, $actP = 0.3$, $area = 80 * 80$.)

active probability, at each time slot, more AUs have higher chances to be active. That leads to more interference on $S_a$ from AUs. Since AUs have absolute priority than RUs on $S_a$, the increase interference coming from AUs reduces the spectrum availability to RUs, thus resulting in more delay. When $S_a$ is not available, RUs will seek transmission opportunity on $S_r$, that is why the energy consumption obtained from $withCR$ is comparable with that in $withoutCR$.

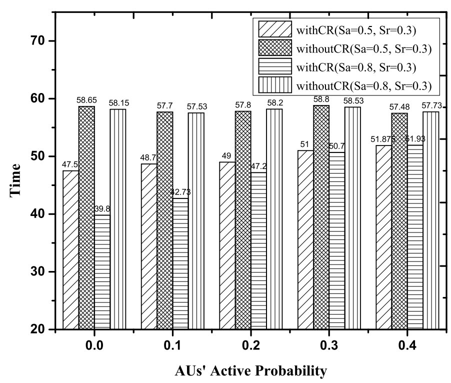
Subsequently, we test how the transmission radius of RUs and AUs influence on the performance of our proposed scheduling algorithm. The results in Fig. 6.3(a) indicate the influence of RUs' transmission radius, and results in Fig. 6.3(b) show the impact of AUs' transmission radius. Fig. 6.3(a) reveals that with the increase of RUs' transmission radius, the delay for both $withCR$ and $withoutCR$ increase, while, $withCR$ outperforms $withoutCR$ due to the existence of $S_a$. For a RU, a bigger transmission radius means a more powerful transmission capability. Particularly, in the PRWN, a RU with larger transmission radius has a potential to find a parent with better success probability within larger transmission range. However, since the transmission constraints concurrent transmissions in a larger range due to interference, the delay increases. Fig. 6.3(b) demonstrates that more interference will brought by AUs when their transmission range expand, thus, results in more delay in



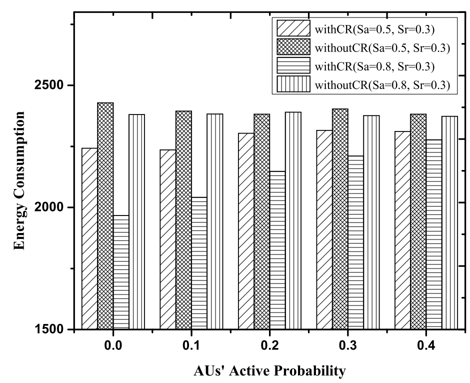

(a) Time efficiency




(b) Energy efficiency




(c) Time efficiency




(d) Energy efficiency

Figure 6.4. Impact of AU's active probability and link quality. ($numAU = 200$, $numRU = 700$, $transAU = 10$, $transRU = 10$, $actP = 0.3$, $area = 100 * 100$.)

$withCR$. However, $withoutCR$ is not affected since $S_a$ has been ignored.

In the end, results in Fig. 6.4 verify the performance regarding to the change of AUs' active probability and the success transmission probability. Fig. 6.4(a) and 6.4(c) show the impact on latency, and Fig. 6.4(b) and 6.4(d) summarize the status of energy consumption. The performance is evaluated under 4 scenarios. For the two scenarios whose results shown in Fig. 6.4(a) and 6.4(b), the success transmission probability on $S_a$ and $S_r$ are assumed to be the same, which simulate the condition that the link quality in the PRWN and AWN are comparable. In one scenario, the lowest connectivity probability is 0.3, and 0.5 in the other scenario. Fig. 6.4(a) further verifies our observation from Fig. 6.2, where the delay of $withCR$ increase with the increases of AUs' active probability, and $withoutCR$ is not be influenced. On the other hand, it also shows that with a better link quality (reflected by the success transmission probability), more RUs can transmit successfully on $S_a$. It improves the scheduling concurrency, so that requires less time compared with that under the condition that link quality is poor. Results shown in Fig. 6.4(c) and 6.4(d) consider the scenarios that $S_a$ has a better link quality compared with that on $S_r$. To be specific, the lowest success probability in the AWN is set as 0.8 and 0.5, respectively, while the lowest success probability in the PRWN is set as 0.3. Under those two scenarios, the link quality of $S_a$ in the AWN is assumed to be better than that in $S_r$. The results shown in Fig. 6.4(c) and 6.4(d) match with our expectation. Compared with Fig. 6.4(a), time needed for scheduling in Fig. 6.4(c) is less because better transmission opportunities are available on $S_a$.

Besides the simulations and results presented above, we have done numerical scenarios which have not been included here due to the space constraint. For all the scenarios, $withCR$ shows its advantage in improving time efficiency while keeping a comparable energy efficiency. However, the improvement comes from $S_a$ is highly related to the degree of interference from the AWN. There are some scenarios during our test, the improvement on time efficiency is only 5% when the interference or link quality in the AWN is extremely poor. The AWN is considered as a probabilistic wireless networks in all scenarios. Therefore, no matter on which spectrum a link is scheduled to transmit, it cannot avoid the potential of retransmission resulting from lossy links. This is the reason for the tiny difference in energy consumption between $withCR$ and $withoutCR$.

# Chapter 7

# CONCLUSION

Data aggregation has been considered as an essential data gathering operation in wireless networks. During the data aggregation process, raw readings are aggregated and then transferred in the network. Since data is aggregated at intermediate nodes during the transmission process, both data redundancy and the number of transmissions are reduced. Therefore, data aggregation is an efficient strategy to alleviate energy consumption and medium access contention. In this dissertation, we focus on the time efficient data aggregation scheduling problem in different wireless networks under varies network models.

First, we investigate the Minimum Latency Data Aggregation Scheduling for Multi-Regional Query(ML-MRQS) problem. A Multi-Regional Query (MRQ) is defined as a query that targets at interested data from multiple regions (overlapping of regions may exist) of a WSN, where each region is a subarea of the WSN. Furthermore, for an MRQ, there is no limitation in query disseminating time interval or assumptions on temporal or spatial correlations. The traditional multi-query processing can be thought of as a special case of the MRQ processing if the query region is the entire network. Therefore, compared with the traditional query model, the MRQ model is more practical and general. Consequently, we investigate the ML-MRQS problem which aims to get a collision-free scheduling plan for an MRQ with minimum latency. A heuristic collision free scheduling algorithm (named Multi-Regional Query Scheduling Algorithm (MRQSA))has been proposed. In MRQSA, we construct a CDS-based scheduling tree for each region. All those scheduling trees form a scheduling forest serving as the data transmission structure during the data transmission procedure. Furthermore, in order to improve the parallelism of MRQSA, we propose to assign higher priorities to the sensor nodes which may cause multiple collisions. During the scheduling procedure, transmissions on different trees are scheduled concurrently. In order to avoid collisions, priority and interference are considered in the entire forest.

Second, with the increasing popularity of Cognitive Radio Networks (CRNs), data aggregation scheduling in CRNs is investigated. The explosive growth of population in wireless networks during the last few decades indicates the urgency of improving the inefficient spectrum usage. The traditional fixed spectrum assignment policy for wireless networks allows no one but licensed Primary Users (PUs) or services to access the spectrums. Particularly, a report from Federal Communications Commission (FCC) demonstrates that the usage of spectrum assigned to PUs is between 15% and 85%. In order to alleviate the spectrum shortage and under-utilization problem, the cognitive radio technology has been introduced and investigated. In Cognitive Radio Networks (CRNs), unlicensed Secondary Users (SUs) coexist with PUs. As long as communications among PUs are ensured [11], SUs are able to access and exploit the unoccupied licensed spectrum using cognitive radio which is capable of sensing/accessing available spectrums and switching between spectrums in an opportunistic manner. Due to the precious spectrum opportunity in CRNs, it is meaningful to study data aggregation which reduces redundant information and only focuses on the most valuable aggregated information in network. Thus, we investigate data aggregation in CRNs under a more general and practical network model. To be specific, we formalize the Minimum Latency Data Aggregation Scheduling (MLDAS) problem in CRNs. Subsequently, the MLDAS problem under the UDG interference model is studied. The idea of using multiple relays to help SUs to forward data is considered. First, to increase SU's potential transmission opportunity and reduce interference simultaneously, a practical distributed scheduling algorithm based on a CDS routing hierarchy is introduced. After that, for the purpose of maximizing the "gain" and minimizing the "pain", an improved solution based on a general routing hierarchy is proposed. Considering the fading property of wireless signal, the Physical Interference Model (PhIM) is considered in this paper. Based on the conclusion of [57], a distributed data aggregation algorithm for PhIM is introduced.

Third, we concentrate on how to use cognitive radio technique to promote the performance of data aggregation in conventional wireless networks. In CRNs, SUs can only access spectrum opportunistically. This reality constraints the usage of CRNs, especially in applications with heavy traffic, even though cognitive radio capability is a useful technology. It would be a pity if we could not make a full use of it. From this point of view, researchers proposed to introduce cognitive radio capability to conventional wireless networks [12][13][14][15]. Motivated from those works, the data aggregation scheduling problem in wireless networks with cognitive radio capability is investigated. Under the defined network model, besides a default working spectrum, users can access extra available spectrum through a cognitive radio. While, scheduling concurrency can be improved via extra transmission opportunity on the extra spectrum, default working spectrum can be used if no extra spectrum is available or interference on the extra spectrum is too high. Our motivation is to accelerate the data aggregation process by seeking transmission opportunity on an extra spectrum without relying on it. We employ the cognitive radio capability in wireless networks to accelerate data transmission. We then formalize and investigate the Minimum Latency Data Aggregation Scheduling problem in wireless networks with Cognitive Radio capability (MLDAS-CR). The MLDAS-CR problem is formalized as an Integer Linear Programming (ILP) problem, then considering the hardness of solving an ILP, the optimal solution of its linear programming relaxation is obtained. Next, a rounding algorithm is employed to obtain a feasible solution for the ILP from the optimal solution of the relaxed Linear Programming (LP).

Finally, we investigate how to make use of cognitive radio capability to accelerate data aggregation in probabilistic wireless networks with lossy links. To be specific, we investigate a probabilistic wireless network where users in the network are equipped with cognitive radios. Wireless devices equipped with cognitive radios are enabled to dynamically adjust their operating transmitter parameters. Therefore, the cognitive radio enables a user to work on the default spectrum of the probabilistic wireless network as a regular radio. It can also hunt extra spectrum resources from the environment by adjusting its transmitter parameters accordingly. Our target is to evaluate how the cognitive radio can be used to improve the data aggregation process regarding time efficiency. Considering the challenges and special characteristics of the problem under investigation, a two phase scheduling algorithm has been proposed. The first phase is finding an efficient routing structure under lossy links and extra transmission opportunity. In the second phase, a dynamic scheduling algorithm is introduced which considers both the link quality and dynamic spectrum availability.

## REFERENCES


[1] "advantage of wireless," http://ipoint-tech.com/wireless-networking-wi-fi-advantages-and-disadvanta

[2] J. Li, S. Cheng, H. Gao, and Z. Cai, "Approximate physical world reconstruction algorithms in sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, 2014.

[3] S. Cheng, J. Li, and Z. Cai, "$O(\epsilon)$-approximation to physical world by sensor networks," in *INFOCOM*, 2013.

[4] J. Gao, J. Li, Z. Cai, and H. Gao, "Composite event coverage in wireless sensor networks with heterogeneous sensors," in *INFOCOM*, 2015.

[5] C. Vu, Z. Cai, and Y. Li, "Distributed energy-efficient algorithms to maximize network lifetime for coverage problem in adjustable sensing ranges wireless sensor networks," *Discrete Mathematics, Algorithms and Applications*, vol. 1, no. 3, pp. 299 – 317, 2009.

[6] S. Cheng, Z. Cai, J. Li, and X. Fang, "Drawing dominant dataset from big sensory data in wireless sensor networks," in *INFOCOM*, 2015.

[7] X. Wang, Z. He, X. Zhao, C. Lin, Y. Pan, and Z. Cai, "Reaction-diffusion modeling of malware propagation in mobile wireless sensor networks," *Science China Information Sciences*, vol. 56, no. 9, pp. 1 – 18, 2013.

[8] L. Guo, C. Ai, X. Wang, Z. Cai, and Y. Li, "Real time clustering of sensory data in wireless sensor networks," in *The 28th IEEE International Performance Computing and Communications Conference*, 2009, pp. 33 –40.

[9] A. Abdelgawad and M. Bayoumi, *Resource-Aware data fusion algorithms for wireless sensor networks*. Springer, 2012.

[10] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *Communications magazine, IEEE*, vol. 40, no. 8, pp. 102–114, 2002.

[11] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006.

[12] H. Al-Mahdi, M. A. Kalil, F. Liers, and A. Mitschele-Thiel, "Increasing spectrum capacity for ad hoc networks using cognitive radios: an analytical model," *Communications Letters, IEEE*, vol. 13, no. 9, pp. 676–678, 2009.

[13] D. B. Zhu and B. D. Choi, "Performance analysis of csma in an unslotted cognitive radio network with licensed channels and unlicensed channels," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, pp. 1–7, 2012.

[14] R. Deng, S. Maharjan, X. Cao, J. Chen, Y. Zhang, and S. Gjessing, "Sensing-delay tradeoff for communication in cognitive radio enabled smart grid," in *Smart Grid Communications (SmartGridComm), 2011 IEEE International Conference on.* IEEE, 2011, pp. 155–160.

[15] N. Bouabdallah, B. Ishibashi, and R. Boutaba, "Performance of cognitive radio-based wireless mesh networks," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 1, pp. 122–135, 2011.

[16] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on.* IEEE, 2004, pp. 517–526.

[17] Y. Liu, Q. Zhang, and L. M. Ni, "Opportunity-based topology control in wireless sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 21, no. 3, pp. 405–416, 2010.

[18] Q. Gao, J. Zhang, and S. V. Hanly, "Cross-layer rate control in wireless networks with lossy links: Leaky-pipe flow, effective network utility maximization and hop-by-hop algorithms," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 6, pp. 3068–3076, 2009.

[19] "dataaggregation," http://searchsqlserver.techtarget.com/definition/data-aggregation.

[20] J. He, S. Ji, Y. Pan, and Y. Li, "Constructing load-balanced data aggregation trees in probabilistic wireless sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 7, pp. 1681–1690, 2014.

[21] S. Ji, R. Beyah, and Z. Cai, "Snapshot/continuous data collection capacity for large-scale probabilistic wireless sensor networks," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 1035–1043.

[22] C. Hua and T.-S. P. Yum, "Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks," *Networking, IEEE/ACM Transactions on*, vol. 16, no. 4, pp. 892–903, 2008.

[23] X. Tang and J. Xu, "Extending network lifetime for precision-constrained data aggregation in wireless sensor networks." in *INFOCOM*, 2006.

[24] Y. Wu, X.-Y. Li, M. Li, and W. Lou, "Energy-efficient wake-up scheduling for data collection and aggregation," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 21, no. 2, pp. 275–287, 2010.

[25] H. Zhang and H. Shen, "Balancing energy consumption to maximize network lifetime in data-gathering sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 20, no. 10, pp. 1526–1539, 2009.

[26] Q. Gao, Y. Zuo, J. Zhang, and X.-H. Peng, "Improving energy efficiency in a wireless

sensor network by combining cooperative mimo with data aggregation," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 8, pp. 3956–3965, 2010.

[27] B. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," in *Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on.* IEEE, 2002, pp. 575–578.

[28] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, "Impact of network density on data aggregation in wireless sensor networks," in *Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on.* IEEE, 2002, pp. 457–458.

[29] T. He, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Aida: Adaptive application-independent data aggregation in wireless sensor networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 3, no. 2, pp. 426–457, 2004.

[30] W. Zhang, S. K. Das, and Y. Liu, "A trust based framework for secure data aggregation in wireless sensor networks," in *Sensor and Ad Hoc Communications and Networks, 2006. SECON'06. 2006 3rd Annual IEEE Communications Society on*, vol. 1. IEEE, 2006, pp. 60–69.

[31] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher, "Pda: Privacy-preserving data aggregation in wireless sensor networks," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE.* IEEE, 2007, pp. 2045–2053.

[32] E. Mlaih and S. A. Aly, "Secure hop-by-hop aggregation of end-to-end concealed data in wireless sensor networks," in *INFOCOM Workshops 2008, IEEE.* IEEE, 2008, pp. 1–6.

[33] R. Bista, K.-J. Jo, and J.-W. Chang, "A new approach to secure aggregation of private data in wireless sensor networks," in *Dependable, Autonomic and Secure Computing, 2009. DASC'09. Eighth IEEE International Conference on.* IEEE, 2009, pp. 394–399.

[34] H. Xiangdong and T. Hui, "Modeling on secure data aggregation of wireless sensor networks based on evaluation of creditability," in *Internet Technology and Applications, 2010 International Conference on.* IEEE, 2010, pp. 1–4.

[35] C.-H. H. Scott, W. Peng-Jun, T. Chinh *et al.*, "Nearly constant approximation for data aggregation scheduling in wireless sensor networks," in *Proceedings of INFOCOM*, 2007, pp. 366–372.

[36] A. Ghosh, O. D. Incel, V. A. Kumar, and B. Krishnamachari, "Multi-channel scheduling algorithms for fast aggregated convergecast in sensor networks," in *Mobile Adhoc and Sensor Systems, 2009. MASS'09. IEEE 6th International Conference on.* IEEE, 2009, pp. 363–372.

[37] C. Joo, J.-G. Choi, and N. B. Shroff, "Delay performance of scheduling with data aggregation in wireless sensor networks," in *INFOCOM, 2010 Proceedings IEEE.* IEEE, 2010, pp. 1–9.

[38] Y. Li, L. Guo, and S. K. Prasad, "An energy-efficient distributed algorithm for minimum-latency aggregation scheduling in wireless sensor networks," in *Distributed Computing Systems (ICDCS), 2010 IEEE 30th International Conference on.* IEEE, 2010, pp. 827–836.

[39] B. Yu and J. Li, "Minimum-time aggregation scheduling in multi-sink sensor networks," in *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011 8th Annual IEEE Communications Society Conference on.* IEEE, 2011, pp. 422–430.

[40] P.-J. Wan, S. C.-H. Huang, L. Wang, Z. Wan, and X. Jia, "Minimum-latency aggregation scheduling in multihop wireless networks," in *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing.* ACM, 2009, pp. 185–194.

[41] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "A survey on spectrum

management in cognitive radio networks," *Communications Magazine, IEEE*, vol. 46, no. 4, pp. 40–48, 2008.

[42] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *Communications Surveys & Tutorials, IEEE*, vol. 11, no. 1, pp. 116–130, 2009.

[43] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Physical Communication*, vol. 4, no. 1, pp. 40–62, 2011.

[44] B. Wang and K. R. Liu, "Advances in cognitive radio networks: A survey," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 1, pp. 5–23, 2011.

[45] P. Steenkiste, D. Sicker, G. Minden, and D. Raychaudhuri, "Future directions in cognitive radio network research," in *NSF workshop report*, vol. 4, no. 1, 2009, pp. 1–2.

[46] S. Kaur, "Intelligence in wireless networks with cognitive radio networks! pushing frontiers with the first lady of emerging technologies," *IETE Technical Review*, vol. 30, no. 1, pp. 6–11, 2013.

[47] B. F. Lo, "A survey of common control channel design in cognitive radio networks," *Physical Communication*, vol. 4, no. 1, pp. 26–39, 2011.

[48] I. Christian, S. Moh, I. Chung, and J. Lee, "Spectrum mobility in cognitive radio networks," *Communications Magazine, IEEE*, vol. 50, no. 6, pp. 114–121, 2012.

[49] Z. Chen and C. Chen, "Adaptive energy-efficient spectrum probing in cognitive radio networks," *Ad Hoc Networks*, vol. 13, pp. 256–270, 2014.

[50] L.-C. Wang, C.-W. Wang, and C.-J. Chang, "Modeling and analysis for spectrum handoffs in cognitive radio networks," *Mobile Computing, IEEE Transactions on*, vol. 11, no. 9, pp. 1499–1513, 2012.

[51] Y. Shi, Y. T. Hou, S. Kompella, and H. D. Sherali, "Maximizing capacity in multihop cognitive radio networks under the sinr model," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 7, pp. 954–967, 2011.

[52] Y. Shi, Y. T. Hou, H. Zhou, and S. F. Midkiff, "Distributed cross-layer optimization for cognitive radio networks," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 8, pp. 4058–4069, 2010.

[53] M. Pan, C. Zhang, P. Li, and Y. Fang, "Joint routing and link scheduling for cognitive radio networks under uncertain spectrum supply," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 2237–2245.

[54] W. Ren, Q. Zhao, and A. Swami, "Power control in cognitive radio networks: how to cross a multi-lane highway," *Selected Areas in Communications, IEEE Journal on*, vol. 27, no. 7, pp. 1283–1296, 2009.

[55] J. Zhao and G. Cao, "Robust topology control in multi-hop cognitive radio networks," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 2032–2040.

[56] Y. Dai, J. Wu, and C. Xin, "Virtual backbone construction for cognitive radio networks without common control channel," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 1456–1464.

[57] Z. Cai, S. Ji, J. He, and A. G. Bourgeois, "Optimal distributed data collection for asynchronous cognitive radio networks," in *Distributed Computing Systems (ICDCS), 2012 IEEE 32nd International Conference on*. IEEE, 2012, pp. 245–254.

[58] A. Cerpa, J. L. Wong, L. Kuang, M. Potkonjak, and D. Estrin, "Statistical model of lossy links in wireless sensor networks," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*. IEEE, 2005, pp. 81–88.

[59] Y. Li, J. Harms, and R. Holte, "Impact of lossy links on performance of multihop

wireless networks," in *Computer Communications and Networks, 2005. ICCCN 2005. Proceedings. 14th International Conference on.* IEEE, 2005, pp. 303–308.

[60] V. G. Subramanian, K. R. Duffy, and D. J. Leith, "Existence and uniqueness of fair rate allocations in lossy wireless networks," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 7, pp. 3401–3406, 2009.

[61] M. Z. Zamalloa, K. Seada, B. Krishnamachari, and A. Helmy, "Efficient geographic routing over lossy links in wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 4, no. 3, p. 12, 2008.

[62] S. Xiao, B. Li, and X. Yuan, "Maximizing precision for energy-efficient data aggregation in wireless sensor networks with lossy links," *Ad Hoc Networks*, vol. 26, pp. 103–113, 2015.

[63] S. Ji, J. S. He, Y. Pan, and Y. Li, "Continuous data aggregation and capacity in probabilistic wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 73, no. 6, pp. 729–745, 2013.

[64] G. Vijay, E. Ben Ali Bdira, and M. Ibnkahla, "Cognition in wireless sensor networks: A perspective," *Sensors Journal, IEEE*, vol. 11, no. 3, pp. 582–592, 2011.

[65] A. S. Zahmati, S. Hussain, X. Fernando, and A. Grami, "Cognitive wireless sensor networks: emerging topics and recent challenges," in *Science and Technology for Humanity (TIC-STH), 2009 IEEE Toronto International Conference.* IEEE, 2009, pp. 593–596.

[66] S. Gao, L. Qian, and D. R. Vaman, "Distributed energy efficient spectrum access in cognitive radio wireless ad hoc networks," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 10, pp. 5202–5213, 2009.

[67] Q. Guan, F. R. Yu, S. Jiang, and G. Wei, "Prediction-based topology control and rout-

ing in cognitive radio mobile ad hoc networks," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 9, pp. 4443–4452, 2010.

[68] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, "Crahns: Cognitive radio ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, 2009.

[69] L. Pan, J. Luo, and J. Li, "Probing queries in wireless sensor networks," in *Distributed Computing Systems, 2008. ICDCS'08. The 28th International Conference on.* IEEE, 2008, pp. 546–553.

[70] A. Meliou, C. Guestrin, and J. M. Hellerstein, "Approximating sensor network queries using in-network summaries," in *Information Processing in Sensor Networks, 2009. IPSN 2009. International Conference on.* IEEE, 2009, pp. 229–240.

[71] X. Liu, Z. Ma, S. Liu, and X. Cao, "Query processing in multi-user scenario for wireless sensor networks," *Sensors Journal, IEEE*, vol. 11, no. 10, pp. 2533–2541, 2011.

[72] Z. Cai, S. Ji, and J. Li, "Data cachingbased query processing in multisink wireless sensor networks," *International Journal of Sensor Networks*, vol. 11, no. 2, pp. 109–125, 2012.

[73] N. Trigoni, Y. Yao, A. Demers, J. Gehrke, and R. Rajaraman, "Multi-query optimization for sensor networks," in *Distributed Computing in Sensor Systems.* Springer, 2005, pp. 307–321.

[74] L. Xie, L. Chen, S. Lu, L. Xie, and D. Chen, "Energy-efficient multi-query optimization over large-scale sensor networks," in *Wireless Algorithms, Systems, and Applications.* Springer, 2006, pp. 127–139.

[75] W. Wang, Y. Wang, X.-Y. Li, W.-Z. Song, and O. Frieder, "Efficient interference-aware tdma link scheduling for static wireless networks," in *Proceedings of the 12th annual international conference on Mobile computing and networking.* ACM, 2006, pp. 262–273.

[76] B. Yu, J. Li, and Y. Li, "Distributed data aggregation scheduling in wireless sensor networks," in *INFOCOM 2009, IEEE*. IEEE, 2009, pp. 2159–2167.

[77] N. Vasanthi and S. Annadurai, "Energy efficient sleep schedule for achieving minimum latency in query based sensor networks," in *Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on*, vol. 2. IEEE, 2006, pp. 214–219.

[78] H. Wu, Q. Luo, and W. Xue, "Distributed cross-layer scheduling for in-network sensor query processing," in *Pervasive Computing and Communications, 2006. PerCom 2006. Fourth Annual IEEE International Conference on*. IEEE, 2006, pp. 10–pp.

[79] Y. Zheng, M. Liu, and J. Cao, "Rsqs: resource-saving multi-query scheduling in wireless sensor networks," in *Asia-Pacific Services Computing Conference, 2008. APSCC'08. IEEE*. IEEE, 2008, pp. 801–806.

[80] O. Chipara, C. Lu, J. Stankovic, and G. Roman, "Dynamic conflict-free transmission scheduling for sensor network queries," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 5, pp. 734–748, 2011.

[81] O. Chipara, C. Lu, and G.-C. Roman, "Real-time query scheduling for wireless sensor networks." in *rtss*, 2007, pp. 389–399.

[82] X. Guan, A. Li, Z. Cai, and T. Ohtsuki, "Coalition graph game for robust routing in cooperative cognitive radio networks," *Mobile Networks and Applications*, vol. 20, no. 2, pp. 147–156, 2015.

[83] S. Gopalakrishnan, "Optimal schedules for sensor network queries," in *Real-Time Systems Symposium (RTSS), 2010 IEEE 31st*. IEEE, 2010, pp. 140–149.

[84] H. Wu, Q. Luo, J. Li, and A. Labrinidis, "Quality aware query scheduling in wireless sensor networks," in *Proceedings of the Sixth International Workshop on Data Management for Sensor Networks*. ACM, 2009, p. 7.

[85] H. Wu and Q. Luo, “Adaptive holistic scheduling for query processing in sensor networks,” *Journal of Parallel and Distributed Computing*, vol. 70, no. 6, pp. 657–670, 2010.

[86] C. Ai, L. Guo, Z. Cai, and Y. Li, “Processing area queries in wireless sensor networks,” in *MSN*, 2009, pp. 1–8.

[87] Z. He, Z. Cai, S. Cheng, and X. Wang, “Approximate aggregation for tracking quantiles in wireless sensor networks,” in *COCOA*, 2014, pp. 161–172.

[88] S. Cheng, Z. Cai, and J. Li, “Curve query processing in wireless sensor networks,” *IEEE Transactions on Vehicular Technology*, 2014.

[89] X. Chen, X. Hu, and J. Zhu, “Minimum data aggregation time problem in wireless sensor networks,” in *Mobile Ad-hoc and Sensor Networks*. Springer, 2005, pp. 133–142.

[90] J. He, S. Ji, M. Yan, Y. Pan, and Y. Li, “Genetic-algorithm-based construction of load-balanced cdss in wireless sensor networks,” in *MILITARY COMMUNICATIONS CONFERENCE, 2011-MILCOM 2011*. IEEE, 2011, pp. 667–672.

[91] J. S. He, Z. Cai, S. Ji, R. Beyah, and Y. Pan, “A genetic algorithm for constructing a reliable mcds in probabilistic wireless networks,” in *Wireless Algorithms, Systems, and Applications*. Springer, 2011, pp. 96–107.

[92] D. Kim, W. Wang, X. Li, Z. Zhang, and W. Wu, “A new constant factor approximation for computing 3-connected m-dominating sets in homogeneous wireless networks,” in *INFOCOM, 2010 Proceedings IEEE*. IEEE, 2010, pp. 1–9.

[93] L. Ding, X. Gao, W. Wu, W. Lee, X. Zhu, and D.-Z. Du, “Distributed construction of connected dominating sets with minimum routing cost in wireless networks,” in *Distributed Computing Systems (ICDCS), 2010 IEEE 30th International Conference on*. IEEE, 2010, pp. 448–457.

[94] J. He, S. Ji, P. Fan, Y. Pan, and Y. Li, "Constructing a load-balanced virtual backbone in wireless sensor networks," in *Computing, Networking and Communications (ICNC), 2012 International Conference on.* IEEE, 2012, pp. 959–963.

[95] X. Gao, Y. Wang, X. Li, and W. Wu, "Analysis on theoretical bounds for approximating dominating set problems," *Discrete Mathematics, Algorithms and Applications*, vol. 1, no. 01, pp. 71–84, 2009.

[96] L. Sun and W. Wang, "On distribution and limits of information dissemination latency and speed in mobile cognitive radio networks," in *INFOCOM, 2011 Proceedings IEEE.* IEEE, 2011, pp. 246–250.

[97] Y. Shi and Y. T. Hou, "A distributed optimization algorithm for multi-hop cognitive radio networks," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE.* IEEE, 2008.

[98] Z. Cai, Z.-Z. Chen, and G. Lin, "A 3.4713-approximation algorithm for the capacitated multicast tree routing problem," *Theoretical Computer Science*, vol. 410, no. 52, pp. 5415–5424, 2008.

[99] Z. Cai, G. Lin, and G. Xue, "Improved approximation algorithms for the capacitated multicast routing problem," in *Proceedings of 11th Annual International Conference on Computing and Combinatorics 2005(COCOON)*, Kunming, China, August 16-29 2005, pp. 136–145.

[100] Z. Cai, R. Goebel, and G. Lin, "Size-constrained tree partitioning: Approximating the multicast k-tree routing problem," *Theoretical Computer Science*, vol. 412, no. 3, pp. 240–245, 2011.

[101] L. Guo, Y. Li, and Z. Cai, "Minimum-latency aggregation scheduling in wireless sensor network," *Journal of Combinatorial Optimization*, 2015.

[102] X. Zheng, J. L. H. Gao, and Z. Cai, "A distributed optimization algorithm for multi-hop cognitive radio networks," in *The 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing.* ACM, 2014.

[103] Z. Cai, R. Goebel, and G. Lin, "Size-constrained tree partitioning: a story on approximation algorithm design for the multicast k-tree routing problem." in *The 3nd Annual International Conference on Combinatorial Optimization and Applications*, 2009.

[104] Y. Duan, G. Liu, and Z. Cai, "Opportunistic channel-hopping based effective rendezvous establishment in cognitive radio networks." in *The 7th International Conference on Wireless Algorithms, Systems and Applications*, 2012, pp. 324–336.

[105] Z. Cai, Y. Duan, and A. Bourgeois, "Delay efficient opportunistic routing in asynchronous multi-channel cognitive radio networks," *Journal of Combinatorial Optimization*, vol. 29, no. 4, pp. 815–835, 2015.

[106] J. Li, S. Cheng, Z. Cai, Q. Han, and H. Gao, "Approximate aggregation in dynamic wireless sensor networks," in *The 10th International Conference on Wireless Algorithms, Systems, and Applications*, 2015.

[107] T. Zhu, X. Wang, S. Cheng, J. Li, and Z. Cai, "Critical points aware adaptive data acquisition algorithm in sensor networks," in *The 10th International Conference on Wireless Algorithms, Systems, and Applications*, 2015.

[108] M. Ren, J. Li, L. Guo, and Z. Cai, "Data collection with probabilistic guarantees in opportunistic wireless networks," *International Journal of Sensor Networks*, 2015.

[109] S. Ji, A. S. Uluagac, R. Beyah, and Z. Cai, "Practical unicast and convergecast scheduling schemes for cognitive radio networks," *Journal of Combinatorial Optimization*, vol. 26, no. 1, pp. 161–177, 2013.

[110] P.-J. Wan, K. M. Alzoubi, and O. Frieder, "Distributed construction of connected dominating set in wireless ad hoc networks," in *INFOCOM 2002. Twenty-First Annual*

*Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3. IEEE, 2002, pp. 1597–1604.

[111] J. Huang, G. Xing, G. Zhou, and R. Zhou, "Beyond co-existence: Exploiting wifi white space for zigbee performance assurance," in *Network Protocols (ICNP), 2010 18th IEEE International Conference on*. IEEE, 2010, pp. 305–314.

[112] X. Zhang and K. G. Shin, "Enabling coexistence of heterogeneous wireless systems: Case for zigbee and wifi," in *Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2011, p. 6.

[113] M. Yan, S. Ji, and Z. Cai, "Time efficient data aggregation scheduling in cognitive radio networks," in *Global Communications Conference (GLOBECOM), 2013 IEEE*. IEEE, 2013, pp. 1185–1190.

[114] S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He, "Atpc: adaptive transmission power control for wireless sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems*. ACM, 2006, pp. 223–236.

[115] M. Yan, S. Ji, M. Han, Y. Li, and Z. Cai, "Data aggregation scheduling in wireless networks with cognitive radio capability," in *Sensing, Communication, and Networking (SECON), 2014 Eleventh Annual IEEE International Conference on*. IEEE, 2014, pp. 513–521.