



HAL
open science

Algorithm design and analysis in wireless networks

Lin Chen

► **To cite this version:**

Lin Chen. Algorithm design and analysis in wireless networks. Data Structures and Algorithms [cs.DS]. Paris-Sud XI, 2017. tel-01688436

HAL Id: tel-01688436

<https://hal.science/tel-01688436>

Submitted on 24 Jan 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Algorithm Design and Analysis in Wireless Networks

Habilitation Thesis

Lin CHEN

Laboratoire de Recherche en Informatique (UMR 8623)

Université Paris-Sud

Defended on July 11, 2017

Committee

M. Tamer BASAR	Reviewer
M. Pierre FRAIGNIAUD	Reviewer
M. Bruno GAUJAL	Reviewer
M. Eitan ALTMAN	Examiner
M. Joffroy BEAUQUIER	Examiner
Mme. Johanne COHEN	Examiner
M. Mérouane DEBBAH	Examiner (president)
M. Fabio MARTIGNON	Examiner

Abstract

Algorithms are perhaps the most fundamental and fascinating elements in computer science as a whole. Networks and networked systems are no exception. This habilitation thesis summarizes my research during the last eight years on some algorithmic problems of both fundamental and practical importance in modern networks and networked systems, more specifically, wireless networks. Generically, wireless networks have a number of common features which form a common ground on which algorithms for wireless networks are designed. These features include the lack of network-wide coordination, large number of nodes, limited energy and computation resource, and the unreliable wireless links. These constraints and considerations make the algorithmic study for wireless networks an emerging research field requiring new tools and methodologies, some of which cannot be drawn from existing state-of-the-art research in either algorithm or networking community.

Motivated by this observation, we aim at making a tiny while systematic step forwards in the design and analysis of algorithms that can scale elegantly, act efficiently in terms of computation and communication, while keeping operations as local and distributed as possible. Specifically, we expose our works on a number of algorithmic problems in emerging wireless networks that are simple to state and intuitively understandable, while of both fundamental and practical importance, and require non-trivial efforts to solve. These problems include (1) channel rendezvous and neighbor discovery, (2) opportunistic channel access, (3) distributed learning, (4) path optimization and scheduling, (5) algorithm design and analysis in radio-frequency identification systems.

Methodologically, most of our analysis is systematically articulated as follows.

- **Theoretical performance bound.** After formulating the target problem, we analytically characterize the performance of the optimal solution as well as some natural and intuitive algorithms in some cases. These results usually give us pertinent insights on the structural properties of the problem including the theoretical limit and the performance gap between the limit and any algorithm that is not carefully devised.
- **Optimum or approximation algorithm design.** Guided by the theoretical results established in the first step, we then direct our efforts to the design and analysis of efficient algorithms for the target problem. By efficient we mean that our algorithms produce either the optimum solution, or, in case where the problem is NP-hard, constant-factor or logarithmic approximations in polynomial or quasi-polynomial time.
- **Further extension and generalization.** Once we have established a complete framework solving or approximately solving the problem, we further analyze the lessons that can be learnt from the analysis process and demonstrate how our framework can be extended or adapted to address a generic class of problems in a wider range of applications presenting similar structural properties.

Contents

Abstract	1
1 Introduction	6
1.1 Background and Motivation	6
1.2 Thesis Methodology	7
1.3 Overview of Major Results and Thesis Organization	8
2 Channel Rendezvous and Neighbor Discovery	12
2.1 Introduction	12
2.1.1 Channel Rendezvous	12
2.1.2 Neighbor Discovery	13
2.1.3 Chapter Organization	14
2.2 The Generalized Telephone Problem	14
2.2.1 Introduction	14
2.2.2 Related Work	16
2.2.3 Problem Formulation	17
2.2.4 Theoretic Performance Bound	19
2.2.5 An Order-optimum Deterministic Algorithm	20
2.2.6 Theoretic Performance Bound for Probabilistic Strategies	22
2.2.7 Variants and Extensions	26
2.3 Neighbor Discovery	27
2.3.1 Context and Motivation	27
2.3.2 Related Work	28
2.3.3 Multi-channel Neighbor Discovery	29
2.3.4 Algorithm-independent Discovery Delay Bound	31
2.3.5 MCD: Single-channel Case	32
2.3.6 MCD: Multi-channel Case	34
2.4 Summary of Other Contributions	36
2.4.1 From Single- to Multi-radio Channel Rendezvous	36
2.4.2 Cooperative Channel Rendezvous	37
2.4.3 Oblivious Neighbor Discovery with Directional Antennas	38
2.5 Conclusion and Perspectives	38
3 Opportunistic Channel Access: A Restless Multi-Armed Bandit Perspective	40
3.1 Introduction	40
3.1.1 Optimality of Myopic Sensing Policy	40
3.1.2 Beyond Myopic Sensing: a Heuristic ν -step Lookahead Policy	41

3.1.3	Chapter Organization	42
3.2	RMAB and its Application	42
3.2.1	MAB and Gittins Index	42
3.2.2	RMAB and Whittle Index	43
3.2.3	Application of RMAB	44
3.2.4	Non-Bayesian MAB	44
3.3	Optimality of Myopic Sensing Policy	45
3.3.1	Problem Formulation	45
3.3.2	Axioms	48
3.3.3	Optimality of Myopic Sensing Policy	49
3.3.4	Discussion	51
3.4	Beyond Myopic Policy: ν -step Lookahead Policy	52
3.4.1	Problem Formulation	52
3.4.2	When Stop Sensing: the ν -step Lookahead Policy	55
3.4.3	One-step Lookahead Policy	57
3.5	Summary of Other Contributions	59
3.5.1	Myopic Policy under Multi-state Markov Channel Model	59
3.5.2	Myopic Channel Probing Policy in Underlay Cognitive Radio Systems	59
3.5.3	RMAB with Multiples Users and Channel Switching Cost	60
3.6	Conclusion and Perspectives	60
4	Distributed Learning in Wireless Networks: A Game-theoretical Perspective	62
4.1	Introduction	62
4.1.1	Imitation-based Spectrum Access	63
4.1.2	Retrospective Spectrum Access	64
4.1.3	Chapter Organization	65
4.2	Imitation-based Spectrum Access	65
4.2.1	System Model and Spectrum Access Game Formulation	65
4.2.2	Imitation-based Spectrum Access	66
4.2.3	Imitation on the Same Channel	69
4.3	Retrospective Spectrum Access	72
4.3.1	System Model and Game Formulation	73
4.3.2	Retrospective Spectrum Access	74
4.3.3	Convergence Analysis	75
4.4	Summary of Other Contributions	79
4.4.1	Joint Operator Pricing and Network Selection in CRNs	79
4.4.2	Distributed Spectrum Management in TV White Space	80
4.4.3	Load Balancing in Smart Grids	80
4.5	Conclusion and Perspectives	81
5	Data Harvesting and Charging Path Optimization	82
5.1	Introduction	82
5.1.1	Time-constrained Data Harvesting in Wireless Sensor Networks	82
5.1.2	Charging Path Optimization and Scheduling in Mobile Networks	83
5.1.3	Chapter Organization	84
5.2	Time-constrained Data Harvesting in Wireless Sensor Networks	84
5.2.1	Background and Related Work	84

5.2.2	Time-constrained Data Harvesting Problem	85
5.2.3	Theoretical Performance Bound: Optimal and Random Algorithms	87
5.2.4	Approximation Algorithm Design	88
5.3	Charging Path Optimization and Scheduling in Mobile Networks	93
5.3.1	Background and Related Work	94
5.3.2	System Model and Problem Formulation	95
5.3.3	Approximation Algorithm Design	98
5.3.4	Discussion on Variants and Extensions	103
5.4	Summary of Other Contributions	104
5.5	Conclusion and Perspectives	105
6	Algorithm Design and Analysis in RFID Systems	106
6.1	Introduction	106
6.1.1	Stability Analysis of Frame Slotted Aloha Protocol	106
6.1.2	Dynamic Tag Population Estimation Algorithm Design	107
6.1.3	Missing Tag Detection Algorithm Design	107
6.1.4	Chapter Organization	108
6.2	Stability Analysis of Frame Slotted Aloha Protocol	108
6.2.1	Introduction	108
6.2.2	Related Work	109
6.2.3	System Model	110
6.2.4	Main Results	111
6.3	Tag Population Estimation Algorithm Design	113
6.3.1	Background and Related Work	113
6.3.2	Technical Preliminaries	114
6.3.3	System Model and Problem Formulation	117
6.3.4	Tag Population Estimation: Static Systems	118
6.3.5	Tag Population Estimation: Dynamic Systems	121
6.3.6	Performance Analysis	124
6.4	Missing Tag Detection Algorithm Design	126
6.4.1	Background and Related Work	126
6.4.2	System Model and Problem Formulation	127
6.4.3	The Baseline approach	129
6.4.4	The Adaptive Approach	131
6.4.5	The Group-wise Approach	132
6.4.6	Discussion	134
6.5	Conclusion and Perspectives	135
7	Conclusion and Perspectives	136
7.1	A Glimpse on other Works	137
7.1.1	Auction-based Resource Allocation	137
7.1.2	Towards Privacy-preserving Resource Allocation Algorithms	138
7.1.3	Game Theoretical Analysis on Smart Grid Security	139
7.2	Perspectives and Future Works	139
7.2.1	Orchestrating Randomness and Determinism	140
7.2.2	Exploring Information Sharing	140
7.2.3	Balancing Optimality and Complexity	140

7.2.4 Towards Online Algorithm Design	141
Bibliography	142
Publications	155

Chapter 1

Introduction

Algorithms are perhaps the most fundamental and fascinating elements in computer science as a whole. Networks and networked systems are no exception. This habilitation thesis summarizes my research during the last eight years on some algorithmic problems of both fundamental and practical importance in modern networks and networked systems, more specifically, wireless networks.

The last two decades have witnessed an unprecedented success of wireless networks due to the proliferation of inexpensive, widely available wireless devices. Examples of classic and emerging wireless networks include cellular networks, wireless local area networks (WLAN), wireless sensor networks (WSN), disruption tolerant networks (DTN), cognitive radio networks (CRN), etc. More formally, a wireless network can be regarded as an interconnection of wireless devices aiming at providing ubiquitous communication and computing services regardless of location, mobility and other properties of individual devices.

1.1 Background and Motivation

Many wireless networks are designed for specific applications and tasks. Nevertheless, there are a number of generic features common for most of them. The following is a summary of these properties that form a common ground on which algorithms for wireless networks are designed. They will be assumed in the thesis. Certain problems and scenarios may require additional constraints, and those specific constraints will be introduced in the corresponding context as needed.

- **Lack of network-wide coordination.** Many wireless networks are by nature distributed systems composed of autonomous wireless devices acting in a decentralized manner. It has become widely recognized that the centralized paradigm of system-wide optimization and design requiring expensive network-wide coordination is no longer adequate for modern large-scale wireless networks.
- **Large number of nodes.** A wireless network may consist of a large number of nodes. An example is wireless sensor networks composed of typically at least thousands of tiny sensors. Any effective algorithm designed for these wireless networks need to scale elegantly.
- **Limited energy resource.** Nodes in wireless networks are usually batter-powered or alimented by local energy-harvesting at a relatively low rate from solar power or other sources. In either case, we need to keep the power consumption of wireless nodes as low as possible.

- **Limited computing resource.** The quest of compatible size and low energy consumption significantly limits the computing and processing capability of individual nodes in many wireless networks. Specifically, many wireless nodes only have a small CPU and memory.
- **Unreliable wireless links.** Wireless links are notoriously unreliable and error-prone. Hence, algorithms should be robust in the sense that they do not rely on long term reliability of wireless links and individual nodes.

The above constraints and considerations make the algorithmic study for wireless networks an emerging research field requiring new tools and methodologies, some of which cannot be drawn from existing state-of-the-art research in either algorithm or networking community. Motivated by this observation, by the present habilitation thesis we hope to make a tiny while systematic step forwards in the design and analysis of algorithms that can scale elegantly, act efficiently in terms of computation and communication, while keeping operations as local and distributed as possible.

1.2 Thesis Methodology

To achieve our humble goal as stated above, we focus in this thesis on a number of algorithmic problems in emerging wireless networks that are simple to state and intuitively understandable, while of both fundamental and practical importance, and require non-trivial efforts to solve. The choice of the problems are to certain extent driven by my preference on fundamental problems that can be posed and formulated neatly and my belief on the existence of simple and elegant solutions for even the hardest problems, for a tiny portion of which we are extremely lucky to find the solutions or approximate them. We would like to point out that our focus on the high-level algorithmic problems does not impair the practicality of our analysis. On the contrary, the simple and generic nature of the problems allow us to concentrate on the intrinsic properties hinging behind them, thus leading to solutions that are applicable in a wide range of engineering applications and easily adaptable and adjustable by incorporating additional design constraints and particularities.

Throughout our analysis, we employ the following high-level methodology to guide our study.

- Algorithmic wireless networking has intrinsic connections to a range of generic and fundamental subjects in theoretical computer science and applied mathematics. Our works thus naturally borrow ideas and tools therein, including some recent and significant developments. These relations are by no means incidental or artificial, but on the contrary can be established rigorously and precisely. Specifically, in our treatment of some networking problems, we sometimes make the problem as generic as possible by removing as many assumptions as possible with significant abstraction in order to reveal the mathematic problem hinging upstream and get insights on the essence of the problem.
- On the other hand, designing and analyzing algorithms under the specific constraints posed by wireless networks reveals or provides a new perspective to some fundamental questions in theoretical computer science and algorithms that would not come to light with a general algorithmic or mathematic point of view. In this regard, we sometimes start from well-studied mathematic problems and demonstrate how incorporating wireless networking constraints brings new flavor to them and make them non-trivial to solve any more. We then develop new algorithms solving them in the new context.

In the thesis, we hope to offer a multi-round and bi-directional journey for readers to appreciate

the synergies created by the above process.

More technically, most of our analysis is systematically articulated as follows.

- **Theoretical performance bound.** After formulating the target problem, we analytically characterize the performance of the optimum solution as well as some natural and intuitive algorithms in some cases, e.g., based on random choices. These results usually give us pertinent insights on the structural properties of the problem including the theoretical limit and the performance gap between the limit and any algorithm that is not carefully devised.
- **Optimum or approximation algorithm design.** Guided by the theoretical results established in the first step, we then direct our efforts to the design and analysis of efficient algorithms for the target problem. By efficient we mean that our algorithms produce either the optimum solution, or, in case where the problem is NP-hard, constant-factor or logarithmic approximations in polynomial or quasi-polynomial time.
- **Further extension and generalization.** Once we have established a complete framework solving or approximately solving the problem, we further analyze the lessons that can be learnt from the analysis process and demonstrate how our framework can be extended or adapted to address a generic class of problems in a wider range of applications presenting similar structural properties.

1.3 Overview of Major Results and Thesis Organization

The results of our study are presented in Chapter 2 to Chapter 6, each devoted to a specific problem, as elaborated below. These chapters together present a selection of our research works and illustrate different facets of the techniques and methodologies used in our research. For convenience, the chapters are arranged as independent units, so that they can be read in any order. In particular, each chapter has its own introduction section, describing the related works and the importance of the results with the specific context of that chapter. In each chapter, we choose to present two or three selected works of us for readers to see insights of the specific problems and the key technicalities in our analysis. We then complete the chapter by briefly summarizing our other contributions related to the topic addressed and outlining the perspectives related to the chapter. To make our presentation concise, readers are referred to our publications for detailed proofs of lemmas and theorems, as well as simulation and numerical analysis.

Chapter 2: Channel Rendezvous and Neighbor Discovery

(Related publications [40, 41, 43, 45, 49, 50, 52])

Chapter 2 focuses on channel rendezvous and neighbor discovery in decentralized networks. The channel rendezvous problem arises from multi-channel wireless networks where establishing communication sessions requires the communicating pairs to meet each other on a common channel to exchange control information. We formulate and analyze the most generic form of the problem termed as *heterogeneous channel rendezvous*: How can two nodes, given asynchronous local clocks, asymmetrical channel perceptions, no universal channel sets, and heterogeneous channel index systems, rendezvous with each other within a minimum upper-bounded rendezvous delay? We remove all the assumptions which may not be realistic in some wireless networking scenarios although they render the problem significantly more tractable. We cast the heterogeneous channel rendezvous problem to the *telephone coordination game*: Two players wishing to communicate are

placed each in a room with N telephones connecting the two rooms. The players do not know how the telephones are interconnected. In each round, each player picks up a phone and says hello until when they hear each other. The problem is to devise an algorithm minimizing the delay to establish communication. We investigate a generalized version where among N telephones, only a subset can establish communication between the two players. We devise a deterministic algorithm achieving bounded and order-minimum worst-case rendezvous delay. The algorithm we develop can be applied to solve the heterogeneous rendezvous problem by regarding telephone lines as channels. We then proceed to study *neighbor discovery*, where a pair of neighbor nodes need to meet each other on the same channel to discover each other. The analogy between channel rendezvous and neighbor discovery is evident. However, neighbor discovery has an additional constraint: each node has a duty cycle which limits the fraction of time when the node is awake. Two nodes can discover each other only if they both wake up on the same channel at the same time. We provide a complete treatment of the neighbor discovery problem, derive the performance limit for any neighbor discovery algorithm and develop an order-optimum algorithm achieving the limit. Our results on this topic are published in [40, 41, 43, 45, 49, 50, 52].

Chapter 3: Opportunistic Channel Access: A Restless Multi-Armed Bandit Perspective (Ph.D. thesis of Kehao Wang, related publications [42, 200–203, 205–207])

Chapter 3 addresses the problem of opportunistic channel access. We consider a communication system in which a sender has access to multiple channels, but is limited to sense and transmit only on one or a limited number of channels in each slot. Sensing is error-prone due to noise and system limitations. We explore how a smart sender should exploit past imperfect observations and the knowledge of the stochastic properties of channels to maximize its transmission rate by switching opportunistically across channels. We provide a generic analysis by casting the problem into the restless multi-armed bandit (RMAB) problem and conduct a systematic analysis on a class of myopic policies of both theoretical and practical importance. We develop three axioms characterizing a family of functions which we refer to as regular functions, which are generic and practically important. We then establish the optimality of the myopic policy when the reward function can be expressed as a regular function and when the discount factor is bounded by a closed-form threshold determined by the reward function. From the methodological perspective, we adopt an axiomatic approach to provide a hierarchical view of the addressed problem, which leads to clearer and more synthetic analysis. The axiomatic approach also helps reduce the complexity of solving the RMAB problem and illustrates some important engineering implications behind the myopic policy. The work of this chapter is in collaboration with my former Ph.D. student Kehao Wang (co-advised with Pr. K. Al Agha) who is now a visiting research associate at MIT. Our results on this topic are published in [42, 200–203, 205–207].

Chapter 4: Distributed Learning in Wireless Networks

(Ph.D. thesis of Stefano Iellamo and Mira Morcos, related publications [23–26, 70, 71, 94–96])

Chapter 4 presents our work on the distributed learning algorithms in wireless networks. We investigate the following problem arising from emerging wireless networks: How to design distributed algorithms that allow users to gradually converge to a stable and desirable system state based on purely location information and interactions? We tackle this problem by using game theory as a systematic framework of modeling and analysis. For concreteness, we instantiate our study by focusing on the distributed channel access in cognitive radio networks and devise a suite of

distributed channel access algorithms based on *imitation*, a behavior rule widely applied in human societies consisting of imitating successful behavior. We establish the convergence of the proposed algorithms to an imitation-stable equilibrium which is also the ϵ -optimum of the system. Simple, natural and incentive-compatible, the proposed spectrum access algorithms can be implemented distributedly based on solely local interactions. We then consider the case where instead of imitating other nodes, a node imitates only its behavior that has brought him higher payoff in the past. Such *self-imitation* demonstrates more robustness in the case where imitating others is not possible or reliable. Technically, we develop and analyze a framework of retrospective spectrum access based on stochastic learning that has two features: (1) the entirely distributed implementation requiring only local observations and (2) the guaranteed statistical convergence to the equilibrium state within a bounded delay. Part of the work of this chapter is the topic of the thesis of my former Ph.D. student Stefano Iellamo (co-advised with Pr. Marceau Coupechoux) who is actually a Marie-Curie research fellow at ICS-FORTH. The ongoing thesis of Mira Morcos (co-advised with Pr. Tijani Chahed) is also related to this topic. Our results on this topic are published in [23–26, 70, 71, 94–96].

Chapter 5: Data Harvesting and Charging Path Optimization

(Related publications [46, 48, 51, 93])

In Chapter 5, we study a class of path optimization and the related scheduling problems arising from data harvesting and mobile charging. We start with data harvesting in sensor networks, where a data ferry (e.g., robot, vehicle) travels across the sensor field and harvests data from sensor nodes while they are within each other's communication range, and later transfers the harvested data to the sink. We investigate the trajectory optimization problem of finding an optimal data harvesting path to collect as much data as possible within a time duration. This *time-constrained data harvesting problem* formulates the situation where delay-sensitive data needs to be reported to the sink within certain amount of time before they become obsolete. We analytically characterize the performance bound of the optimal data harvesting algorithm. We give a formal proof on the NP-hardness of the time-constrained data harvesting problem and design a constant-factor approximation algorithm. We then move to the path optimization and charger scheduling problems in mobile charging applications, where a mobile charger needs to deliver energy to a set of mobile nodes. In this application, optimizing the trajectory of the mobile charger is a primary concern for maintaining the operation of these systems. We formulate a class of generic path optimization problems and concentrate on the problem of maximizing the number of nodes charged within a fixed time horizon. We prove that these problems are either NP-hard or APX-hard. We design a quasi-polynomial time algorithm that achieves logarithmic approximation to the optimum charging path. We also demonstrate how our approximation algorithm can be adapted and extended to solve other charging path optimization and scheduling problems. Our results on this topic are published in [46, 48, 51, 93].

Chapter 6: Algorithm Design and Analysis in RFID Systems

(Ph.D. thesis of Jihong Yu, related publications [227–230])

In Chapter 6, we focus on algorithm design and analysis in RFID (radio-frequency identification) systems, which are becoming ubiquitously available today in many domains ranging from warehouse management, object tracking to inventory control. Our focus is tag counting and monitoring, one of the most fundamental functionalities in RFID systems, particularly when the system

scales. We start by studying the stability of the Frame Slotted Aloha (FSA) protocol, the *de facto* MAC layer standard in tag identification. Very limited work has been done on the stability of FSA. To bridge this gap, we investigate the stability properties of FSA by focusing on two physical layer models of practical importance, the models with single packet reception (SPR) and multipacket reception (MPR) capabilities. By employing drift analysis, we obtain the closed-form conditions for the stability of FSA and when the stability region is maximized. Furthermore, to characterize system behavior in the instability region, we mathematically demonstrate the existence of transience of the backlog Markov chain. We then proceed to study the problem of tag population estimation by developing a generic framework of stable and accurate tag population estimation schemes for both static and dynamic RFID systems. By generic, we mean that our framework does not require any prior knowledge on the tag arrival and departure patterns. By performing Lyapunov drift analysis, we mathematically prove the efficiency and stability of our framework. We complete our work with a comprehensive analysis on the detection of missing tags, one of the most important RFID applications. Specifically, we develop a suite of three missing tag detection algorithms, each decreasing the execution time compared to its predecessor. By sequentially analysing the developed algorithms, we gradually iron out an optimum detection algorithm that works in practice. The work of this chapter is in collaboration with my former Ph.D. student Jihong Yu who has just defended his thesis and started his post-doc at Simon Fraser University in Canada. Our results on this topic are published in [227–230].

Chapter 7 concludes the habilitation thesis by summarizing our other research works related to the thesis and pointing out some perspectives for future research. In addition to a summary of the future directions given in the concluding section of each chapter, focusing on the immediate extensions of the corresponding models and analysis, in this concluding chapter we take a broader view and consider more general directions in algorithm design and analysis in emerging networked systems.

Chapter 2

Channel Rendezvous and Neighbor Discovery

2.1 Introduction

2.1.1 Channel Rendezvous

The operating frequencies of today's mobile devices typically span a swath of spectrum subdivided into multiple orthogonal channels. In such multi-channel wireless networks, establishing communication sessions requires the communicating pairs to meet each other on a common channel via a *rendezvous* process to exchange control information before initiating data communications. The use of a single common control channel for rendezvous simplifies the rendezvous process but it creates a single point of failure as the common control channel may become temporarily unavailable, leading to the rendezvous failure problem. Therefore, channel hopping (CH) approaches have been widely used to create multiple rendezvous channels. Specifically, each node starts a channel hopping process according to its own CH sequence and local clock. The CH sequences are carefully chosen to spread out rendezvous over multiple pairwise rendezvous channels.

There are three major challenges in the design of distributed CH-based rendezvous algorithm.

- *Lack of clock synchronization.* It is difficult to require two independent nodes to have synchronized clocks without any handshake between them.
- *Asymmetrical perceptions of channel availability.* Nodes may have different set of accessible channels, thus increasing the difficulty in finding a common rendezvous channel.
- *No universal channel set or common channel index system.* Nodes may not have the same knowledge on the universal channel set or a common way of mapping channel indices to frequencies. For example, given three channels with center frequencies f_a , f_b and f_c , two nodes using the same CH sequence 0, 1, 2 fail to achieve rendezvous when one node maps indices 0, 1, 2 to frequencies f_a, f_b, f_c , while the other maps indices 0, 1, 2 to f_b, f_c, f_a .

We emphasize that it is the holistic combination of the above three challenges that makes the design of distributed rendezvous algorithm non-trivial. Formally, we coin the term *heterogeneous channel rendezvous problem* to denote the following problem.

Problem 2.1 (Heterogeneous channel rendezvous). *How can two nodes, given asynchronous local clocks, asymmetrical channel perceptions, no universal channel sets, and heterogeneous channel index systems, rendezvous with each other within a minimum upper-bounded rendezvous delay?*

The heterogeneous rendezvous problem we define is the most generic form of the channel rendezvous problem, in which we remove all the assumptions which may not be realistic in some wireless networking scenarios although they render the problem significantly more tractable. To tackle the problem, we cast it to the *telephone coordination game*, a problem of fundamental importance in distributed algorithm design, defined as below.

Problem 2.2 (Telephone coordination game). *Two players wishing to communicate are placed each in a room with N telephones connecting the two rooms. The players do not know how the telephones are interconnected. In each round, each player picks up a phone and says hello until when they hear each other. The problem is to devise an algorithm minimizing the delay to establish communication.*

In our work, we investigate a generalized version where among N telephones, only a subset can establish communication between the two players. We devise a deterministic algorithm achieving bounded and order-minimum worst-case rendezvous delay. The algorithm we develop can be applied to solve the heterogeneous rendezvous problem by regarding telephone lines as channels.

2.1.2 Neighbor Discovery

After solving the channel rendezvous problem, we proceed to study a related problem, *neighbor discovery*, where a pair of neighbor nodes need to meet each other on the same channel to discover each other. The analogy between the two problems is evident. However, neighbor discovery has a distinguished constraint besides the design challenges in channel rendezvous. Each node has a duty cycle which limits the fraction of time when the node is awake. Two nodes can discover each other only if they both wake up on the same channel at the same time. Compared to channel rendezvous, neighbor discovery has an additional dimension, duty cycle, which makes the problem harder and calls for a dedicated investigation. Formally, we define the following heterogeneous neighbor discovery problem.

Problem 2.3 (Heterogeneous neighbor discovery). *How can neighbor nodes with heterogeneous duty cycles, operating on different channels, without clock synchronization, discover each other over every common channel within a bounded delay?*

Particularly, the following requirements should be satisfied:

- Maximum discovery diversity/robustness with bounded (and minimum) worst-case discovery delay;
- Support for heterogeneous and arbitrary duty cycles to provide fine-grained control of energy conservation levels.

Again, it is the combination of the above design requirements that makes the neighbor discovery algorithm design non-trivial and should be handled holistically.

Our work on the heterogeneous neighbor discovery problem is articulated as below.

- We establish a theoretical framework, under which we derive the performance bound of any neighbor discovery algorithm achieving guaranteed discovery. For any algorithm achieving full discovery diversity, we show that the lower-bound of the discovery delay L scales squarely in the number of channels N , and linearly in the duty cycle reciprocal of any of the two neighbor nodes.
- We design a multi-channel discovery algorithm that achieves guaranteed discovery with order-minimum worst-case discovery delay and fine-grained control over duty cycles.

2.1.3 Chapter Organization

The rest of this chapter is structured as follows. Section 2.2 develops our work on the telephone problem and channel rendezvous. Section 2.3 presents our work on neighbor discovery. Section 2.4 concludes the chapter by briefly summarizing our other related work related to this topic. More details of our work on this topic including proofs and numerical analysis can be found in our publications [40, 41, 43, 45, 49, 50, 52].

2.2 The Generalized Telephone Problem

2.2.1 Introduction

2.2.1.1 The Telephone Problem

The *Telephone Coordination Game*, also referred to as the *Telephone Problem*, was first formally introduced by Alpern in 1976 [9] as follows: Two players wishing to communicate with each other are placed each in a distinct room with N telephone lines connecting the two rooms. The players do not know how the telephones are interconnected. The game is played in discrete steps termed as rounds. In each round $t = 0, 1, 2, \dots$, each player picks up a phone and says “hello” until when they hear each other. The common aim of the two players is to minimize the time until they can hear each other. The major difficulty that makes the Telephone Problem non-trivial is the *lack of any form of coordination* between the two players, as summarized in the following:

- *No common phone labeling*: The N telephones are identical and not labeled or ordered in any way. In other words, if each player puts a label on each phone for the purpose of identification, by no means they can have any common labeling of the phones.
- *No time synchronization*: Each player is unaware of the moment when the other starts the game. That is, there does not exist any external signal coordinating the searching process such as starting or stopping signals.
- *No pre-assigned roles*: The players do not have pre-assigned roles as a caller or a callee because such role assignment requires some form of coordination. Even if such coordination is possible, a player may wish to be a caller to initiate a communication session and a callee at the same time to accept incoming communication sessions from other players. In this case, it is impossible to assign a role to each player.

The Telephone Problem reflects a typical paradigm of distributed algorithm design without any prior coordination among agents. Despite (or thanks to) its simple and generic formulation, a number of engineering problems can be cast into it, ranging from communication link establishment, meeting scheduling to web-crawling.

Mathematically, the Telephone Problem belongs to the field of *Rendezvous Search Games* [10, 11]. Due to its application in many engineering problems, the Telephone Problem and its variants have attracted extensive research attention, but the original Telephone Problem still remains open, even though it is simple to state and intuitively understandable. Until today, very little result has been reported on the structure of the optimum algorithm, among which the most important progress toward characterizing the structure of the optimum probabilistic algorithm was made by Anderson and Weber [15].

2.2.1.2 Anderson and Weber Algorithm

Anderson and Weber studied the Telephone Problem using another formulation as a symmetrical rendezvous search game on N locations between two players [15]. Each player can switch across different locations freely from one to any other location and the delay required to pass from one location to another is negligible. The objective is to find the optimum algorithm of visiting a location each round to minimize the expected rendezvous delay. By regarding locations as telephones, the rendezvous search game considered in [15] can be cast into the Telephone Problem.

Aiming at minimizing the expected rendezvous delay, Anderson and Weber developed a probabilistic algorithm, referred to as *Anderson and Weber algorithm*, in which each player waits where it is for $N - 1$ periods with probability θ and switch to a random permutation of the remaining $N - 1$ locations with probability $1 - \theta$. The algorithm is repeated each $N - 1$ periods until rendezvous. By calculating the value of θ , they have demonstrated the optimality of the proposed algorithm in minimizing the expected rendezvous delay for the cases $N = 2$ and 3 . However, for other values of N , even small values, the Telephone Problem remains unsolved.

2.2.1.3 The Generalized Telephone Problem and Our Main Results

We revisit the classical Telephone Problem in its generic form. *Specifically, we investigate a generalized version of the Telephone Problem by studying the situation where among the N telephone lines, only a subset of them, unknown to the players, can establish the communication between the two players.* Others are connected to telephones in other rooms than the rooms of the players. This generalization of the original Telephone Problem, termed as *Generalized Telephone Problem*, can capture a number of engineering and system constraints in practical settings, e.g., in the channel establishment problem, a channel may be temporally occupied by other users and thus cannot be accessed; in the problem of robot rendezvous, some places may be inaccessible for some robots due to resource constraint or security reasons. Compared to the original Telephone Problem, the Generalized Telephone Problem we consider has one more difficulty, recaptured as follows:

- *Partial telephone connection*: Only a subset of the N telephones, not known by the players, can establish the communication between the two players.

In the Generalized Telephone Problem, we are interested in finding the optimum *deterministic* algorithm that is able to achieve *bounded* rendezvous delay and that minimizes the *worst-case* rendezvous delay. Our work consists of a complementary research thrust compared to that pioneered by Anderson and Weber in [15] seeking *probabilistic* strategies minimizing the *expected* meeting delay. Our focus on the deterministic algorithm is motivated by the long-tail effect of the probabilistic strategies in which two players may experience extremely long and unbounded delay before they can meet each other and consequently the need of rendezvous strategies that can satisfy engineering applications requiring bounded rendezvous delay.

Particularly, we investigate the following natural while fundamental questions:

- What is the structure of a deterministic algorithm achieving bounded rendezvous delay?
- What is the worst-case rendezvous delay bound for any deterministic algorithm?
- How to design a deterministic algorithm approaching the worst-case delay bound?

By our analysis, we give the answers to the above questions.

- *D-bounded phone pick sequence*: We characterize the structure of the phone pick sequences of

the deterministic strategies (termed as D -bounded phone pick sequences) that can guarantee rendezvous without any coordination;

- *Worst-case rendezvous delay bound:* We prove that the lower-bound of the worst-case rendezvous delay of any deterministic algorithm is $O(N^2)$;
- *Zero-knowledge Rendezvous:* We devise a deterministic algorithm, called Zero-knowledge Rendezvous, that (1) guarantees rendezvous between the players regardless of their telephone labeling functions and their relative time difference and (2) approaches the performance limit without any prior knowledge or coordination.

2.2.2 Related Work

The Telephone Problem and the related discovery and rendezvous problems belong to the field of Rendezvous Search Games, which is extensively surveyed in [10]. In the following, we briefly summarize the related work.

Original Telephone Problem: Despite significant research efforts, the original Telephone Problem still remains open today, even though it is simple to state and intuitively understandable. The optimum probabilistic algorithm minimizing the expected rendezvous delay has only been derived for the cases $N = 2$ and 3 [15], [211]. For other values of N (even small values), characterizing the optimum probabilistic algorithm remains unsolved.

Rendezvous games on graphs: Recently, motivated by the rendezvous problems between robots, the rendezvous games on graphs and their different variants have attracted significant research attention, both from probabilistic and deterministic perspectives (cf. [61, 63, 64, 186] and the references therein). Particularly, concerning the deterministic strategies which are more related to our work, although a number of solutions have been proposed to achieve rendezvous on graphs, the majority of them are focused on specific graphs and develop strategies with polynomial complexity, leaving the optimality of the propositions unaddressed. Motivated by this observation, we focus on establishing the theoretical performance bound for any deterministic algorithm and devising strategies that can approach the performance bound. The problem we address can be regarded as a specific version of rendezvous game on graphs where players can switch freely from one vertex to any other vertex.

Channel rendezvous problem: Our work is also related and applicable to the channel rendezvous problem in multi-channel networks, in which a number of distributed channel rendezvous solutions have been proposed in the literature recently [19, 30, 31, 121, 190, 214, 236]. However, none of them addressed all the three challenges due to the lack of coordination among players, i.e., common telephone labeling function, synchronization, and preassigned player roles. We would like to point out that it is the holistic combination of the three challenges that makes the design of rendezvous strategies far from trivial. Particularly, most, if not all, of the existing work implicitly assumes identical channel labeling for the rendezvous pair. However, establishing reliable channel labeling requires coordination between the rendezvous pair, which cannot be satisfied before the rendezvous is achieved. Some rendezvous schemes use the quality of a channel as its label. This approach is not reliable, either because the perceived quality of the same channel may vary significantly at different nodes due to their locations. Moreover, due to the application-oriented approach, none of existing work has a complete study from the theoretical perspective as ours on the channel rendezvous problem which can be cast into the Telephone Problem.

2.2.3 Problem Formulation

We first formulate the Generalized Telephone Problem and the deterministic rendezvous algorithm.

2.2.3.1 The Generalized Telephone Problem

Consider two players, Alice (a) and Bob (b), who wish to communicate with each other, each placed in a distinct room. In the room of Alice (Bob), there are N_a (N_b) telephones among which $N_c \leq \min\{N_a, N_b\}$ telephones can be used to connect the two rooms. Other telephones may be connected to those in places other than the rooms of Alice and Bob. In our analysis, we focus on the extreme case by setting $N_c = 1$, i.e., there is only one telephone connecting the two rooms. Our focus on the extreme case allows us to concentrate on the essential properties of the problem and the resulting algorithm. The extension to the general case where $N_c > 1$ is straightforward. The two players do not know how the telephones are interconnected.

Time is divided into rounds. In each round $t = 0, 1, 2, \dots$, each player picks up a phone and says “hello” until when they hear each other, termed as a pairwise *rendezvous*. The common aim of the two players is to minimize the worst-case delay until they can hear each other. As in the original Telephone Problem, no prior coordination is possible between Alice and Bob, meaning that they do not have (1) common phone labeling, (2) time synchronization, (3) pre-assigned roles.

2.2.3.2 Telephone Labeling Function

We now introduce the *telephone labeling function* to formalize the first constraint.

Definition 2.1 (Telephone Labeling Function). Denote \mathcal{C} the index set of all telephone lines where each index $c \in \mathcal{C}$ denotes the pair of telephones connected by connection c . For each player i ($i = a, b$) with a set $\mathcal{N}_i \in \mathcal{C}$ of telephones in its room, we define the telephone label function f_i as a bi-injective mapping

$$f_i : \mathcal{N}_i \rightarrow \{0, 1, \dots, N_i - 1\},$$

where $\forall c_1, c_2 \in \mathcal{N}_i, c_1 \neq c_2$ implies $f_i(c_1) \neq f_i(c_2)$. We define $f_i^{-1} : \{0, 1, \dots, N_i - 1\} \rightarrow \mathcal{N}_i$ as the inverse mapping of f_i .

Remark. Definition 2.1 basically states that each player i has its own labeling of the N_i telephones in his room that may differ from the global label set of the N_i telephones \mathcal{N}_i . Using Definition 2.1, we can express formally the constraint that there is only one telephone connecting the two rooms ($|\mathcal{N}_a \cap \mathcal{N}_b| = 1$) as follows: There exists a unique telephone line $c^* \in \mathcal{C}$ such that there exist $0 \leq h_a \leq N_a - 1$ and $0 \leq h_b \leq N_b - 1$ such that $f_a(c^*) = h_a$ and $f_b(c^*) = h_b$, or equivalently, $f_a^{-1}(h_a) = f_b^{-1}(h_b) = c^*$.

The following example further illustrates Definition 2.1.

Example 1. Consider a system with $\mathcal{C} = \{c_0, c_1, c_2, c_3\}$. There are 2 and 3 telephones in the room of Alice and Bob, respectively, with $\mathcal{N}_a = \{c_0, c_1\}$, $\mathcal{N}_b = \{c_1, c_2, c_3\}$. The telephone labeling functions of Alice and Bob are: $f_a(c_0) = 0$, $f_a(c_1) = 1$ and $f_b(c_1) = 2$, $f_b(c_2) = 1$, $f_b(c_3) = 0$. It can be noted that Alice and Bob can communicate only via telephone line c_1 . Mathematically, $f_a^{-1}(1) = f_b^{-1}(2) = c_1$.

2.2.3.3 Phone Pick Sequence

As analysed previously, any probabilistic algorithm cannot achieve bounded rendezvous delay and suffers from the long-tail rendezvous latency problem in which Alice and Bob may experience extremely long delay before they can rendezvous. Motivated by this observation, we consider deterministic rendezvous strategies in which each player picks up a phone each round based on a specific sequence so as to rendezvous with its peer. We call such sequence the *phone pick sequence* and give its formal definition in the following.

Definition 2.2 (Phone Pick Sequence). *The phone pick sequence of a player is defined as a sequence $\mathbf{u} \triangleq \{u_t\}_{0 \leq t \leq T_u - 1}$ where u_t is the index of the telephone picked by the player in round t based on its own labeling, T_u is the period of the sequence¹.*

Given the phone pick sequences of Alice and Bob denoted as \mathbf{u} and \mathbf{v} , whose periods are denoted as T_a and T_b , if there exists $t \in [0, T_a T_b - 1]$ and $c \in \mathcal{N}_a \cap \mathcal{N}_b$ such that $f_a^{-1}(u_t) = f_b^{-1}(v_t) = c$, Alice and Bob can rendezvous in round t on telephone line c . t is called the rendezvous round and c the rendezvous telephone.

Example 2. Consider the setting of Example 1 with the following phone pick sequences for Alice and Bob: $\mathbf{u} = \{0, 1\}$ with $T_a = 2$ and $\mathbf{v} = \{0, 1, 2\}$ with $T_b = 3$. It can be noted that they can rendezvous in round 5 on telephone c_1 . However, if Alice and Bob operate on the following phone pick sequences: $\mathbf{u} = \{0, 1\}$ with $T_a = 2$ and $\mathbf{v} = \{2, 1, 0, 1\}$ with $T_b = 4$, they can never rendezvous. The phone pick sequences and the rendezvous process are illustrated in Fig. 2.1.

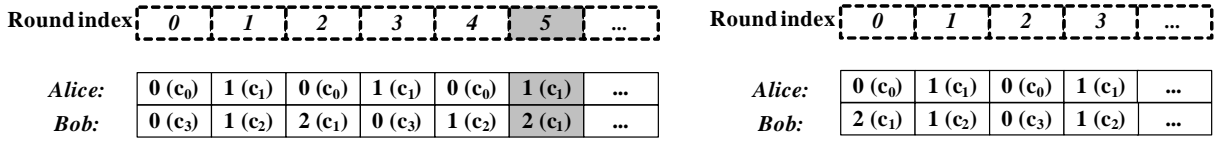


Figure 2.1: Example of phone pick sequences: left: $\mathbf{u} = \{0, 1\}$, $\mathbf{v} = \{0, 1, 2\}$; right: $\mathbf{u} = \{0, 1\}$, $\mathbf{v} = \{2, 1, 0, 1\}$.

To model the situation where the players are not synchronized such that they may start their search in different time instances, we apply the concept of *cyclic rotation* to the phone pick sequences. Specifically, given a phone pick sequence \mathbf{w} , we denote $\mathbf{w}(k)$ a cyclic rotation of \mathbf{w} by k rounds and k is referred to as the *cyclic rotation phase*. Consider an example where $\mathbf{u} = \{0, 1, 2\}$ with $T_u = 3$, we have $\mathbf{u}(2) = \{2, 0, 1\}$.

2.2.3.4 D-bounded Phone Pick Sequence

We define in the following the D -bounded rendezvous system consisting of the D -bounded phone pick sequences, any pair of which can rendezvous within D rounds regardless of the cyclic rotation phases and the telephone labeling functions of the players.

Definition 2.3 (D -bounded Rendezvous System). *A D -bounded rendezvous system is defined as a set of phone pick sequences such that any two distinct sequences \mathbf{u} and \mathbf{v} satisfy the following property:*

$$\exists 0 \leq t < D \text{ such that } f^{-1}[u_t(t_0)] = f'^{-1}[v_t(t'_0)], \forall f, f' \in \mathcal{F}, t_0, t'_0.$$

¹A probabilistic rendezvous algorithm can be regarded as a special case where $T_u \rightarrow \infty$.

where \mathcal{F} denotes the set of telephone labeling functions.

Definition 2.4 (*D*-bounded Phone Pick Sequence). *The phone pick sequences in a D-bounded rendezvous system are called D-bounded phone pick sequences.*

Armed with the above definitions and the mathematic notations introduced in this section, we can formalize the Generalized Telephone Problem with deterministic algorithm as follows.

Generalized Telephone Problem with deterministic algorithm. The Generalized Telephone Problem with deterministic algorithm consists of devising *D*-bounded phone pick sequences \mathbf{u} and \mathbf{v} for Alice and Bob to minimize the worst-case rendezvous delay bound *D*. In other words, we seek an algorithm to construct phone pick sequences to achieve bounded and minimum worst-case rendezvous delay among all possible telephone labeling functions and all cyclic rotation phases of the two players.

2.2.4 Theoretic Performance Bound

We next establish the worst-case rendezvous delay bound for any deterministic algorithm. We also analyse the structure of the phone pick sequence to guarantee rendezvous between Alice and Bob regardless of their telephone labeling functions and cyclic rotation phases. The results derived in this subsection serve as design guidelines for the order-optimum deterministic algorithm devised later in Sec. 2.2.5 that approaches the performance bound. The proofs, detailed in [40], consists of constructing contradictions and applying the definition of the generalized telephone problem.

Lemma 2.1 (Structural property of *D*-bounded phone pick sequence). *If Alice and Bob can rendezvous with D rounds by using the D-bounded phone pick sequences \mathbf{u} and \mathbf{v} , then for any cyclic rotation phases t_0^a and t_0^b and any telephone label pair (h_a, h_b) where $0 \leq h_a \leq N_a - 1$ and $0 \leq h_b \leq N_b - 1$, there exists $t < D$ such that $u_t(t_0^a) = h_a$ and $v_t(t_0^b) = h_b$.*

Lemma 2.1 shows that given any cyclic rotation phases t_0^a and t_0^b , to ensure rendezvous within *D* rounds, the pair $(u_t(t_0^a), v_t(t_0^b))$ ($0 \leq t < D$) must cover all the possible telephone label couples (h_a, h_b) where $0 \leq h_a \leq N_a - 1$ and $0 \leq h_b \leq N_b - 1$. In other words, the two *D*-bounded phone pick sequences of Alice and Bob should cover all couples in $[0, N_a - 1] \times [0, N_b - 1]$. The following result follows straightforwardly.

Theorem 2.1 (Worst-case rendezvous delay lower bound). *For any pair of D-bounded phone pick sequences, the worst-case rendezvous delay among all possible telephone labeling functions and all cyclic rotation phases cannot be lower than $N_a N_b$, i.e., $D \geq N_a N_b$.*

Having established the worst-case rendezvous delay bound of any deterministic rendezvous algorithm, we now investigate the structure of the *D*-bounded phone pick sequences that can guarantee rendezvous between Alice and Bob regardless of their telephone labeling functions and their cyclic rotation phases. Without loss of generality, we investigate the structure of the phone pick sequence of Alice \mathbf{u} by focusing on its period T_u . The following theorem holds symmetrically for Bob, whose phone pick sequence is \mathbf{v} of period T_v .

Theorem 2.2 (Lower-bound of T_u). *If the rendezvous can be guaranteed between Alice and Bob regardless of their cyclic rotation phases and N_a, N_b , then it holds that $T_u \geq N_a^2$.*

2.2.5 An Order-optimum Deterministic Algorithm

We now develop an order-optimum deterministic rendezvous algorithm, Zero-knowledge Rendezvous, which (1) guarantees rendezvous between Alice and Bob regardless of their telephone labeling functions and their cyclic rotation phases and (2) approaches the performance bound derived in Section 2.3.4 without any prior knowledge or coordination.

2.2.5.1 Phone Pick Sequence Construction

The phone pick sequence of each player is constructed based on its ID, which is globally unique. Examples of such globally unique IDs includes one's passport number, biometric identities such as DNA sequence and in case of a communication device its address. Mathematically, we define a player's ID as a binary sequence of length l composed of a sequence of bits where each bit takes either a value of 0 or 1.

Remark. *Using globally unique IDs in the design of Zero-knowledge Rendezvous is a way of breaking the symmetry between the two players and is realistic in many practical engineering problems. A natural question that arises is whether it is possible to devise a deterministic algorithm with bounded rendezvous latency without breaking any form of symmetry between the players. In other words, players are treated as indistinguishable entities. The response is unfortunately no. We next give a counterexample which is simple while sufficient to demonstrate the impossibility of devising a rendezvous algorithm without any form of symmetry breaking. Consider the case where Alice and Bob are perfectly synchronized and $N_a = N_b$. Without symmetry breaking, it holds that $\mathbf{u} = \mathbf{v}$. If the unique telephone connecting them c_0 is indexed differently by Alice and Bob, i.e., $f_a^{-1}(c_0) \neq f_b^{-1}(c_0)$, the rendezvous can never be achieved.*

The phone pick sequence construction process is composed of three steps.

Step 1: Constructing cyclic rotationally distinct padded binary sequence

In the first step, each player independently generates a binary sequence based on its ID such that the binary sequences of any two players are *cyclic rotationally distinct* one to the other. Given any two sequences \mathbf{a} and \mathbf{b} of l bits, they are cyclic rotationally distinct to each other if and only if $\mathbf{a} \neq \mathbf{b}(k)$ for any $k \in [0, l - 1]$. We next show how to construct such cyclic rotationally distinct binary sequences.

Let α denote the ID of a player (say, Alice) and let $\mathbf{1}$ ($\mathbf{0}$) denote a sequence of 1 (0) of length $l' = \lceil \frac{l}{2} \rceil$. We construct the padded ID of Alice as the concatenation of $\mathbf{0}$, α and $\mathbf{1}$, denoted as $\mathbf{0}||\alpha||\mathbf{1}$. By the following lemma, we show that the padded ID sequences generated in such way based on different ID sequences are cyclic rotationally distinct one to another.

Lemma 2.2. *Given any two padded ID sequences \mathbf{a} and \mathbf{b} generated from two ID sequences α and β in the way that $\mathbf{a} \triangleq \mathbf{0}||\alpha||\mathbf{1}$ and $\mathbf{b} \triangleq \mathbf{0}||\beta||\mathbf{1}$, it holds that*

$$\alpha \neq \beta \implies \mathbf{a} \neq \mathbf{b}(k), \forall k \in (0, l + 2l'],$$

where $\mathbf{b}(k)$ is \mathbf{b} with a cyclic rotation of k bits.

Step 2: Generating regular sequence

Denote the padded ID sequence as \mathbf{e}^i for player i , the second step for each player is to generate a sequence \mathbf{o}^i based on \mathbf{e}^i such that for any two players i, j , there always exists t_1 and t_2 such

Algorithm 1 Construct a regular sequence \mathbf{o}^i

Input: Padded ID sequence \mathbf{e}^i of $l + 2l'$ bits

Output: Regular sequence \mathbf{o}^i

- 1: **for** $t = 0$ to $l + 2l' - 1$ **do**
 - 2: **switch** e_t^i
 - 3: **case** 1: expand e_t^i into four bits 0101
 - 4: **case** 0: expand e_t^i into four bits 0011
 - 5: **end for**
 - 6: $\mathbf{o}^i \leftarrow$ the expanded sequence of \mathbf{e}^i
-

Algorithm 2 Construct the phone pick sequence for Alice: the overall algorithm

Input: ID sequence α of l bits

Output: Phone pick sequence \mathbf{u}

- 1: Form the padded ID sequence $\mathbf{a} = \mathbf{0}||\alpha||\mathbf{1}$
 - 2: Construct the regular sequence \mathbf{o}^a (Algorithm 1)
 - 3: Choose two prime numbers p_a^0 and p_a^1 larger than N_a and coprime to $L_o = 4(l + \lfloor \frac{l}{2} \rfloor)$
 - 4: Construct the phone pick sequence \mathbf{u} based on (2.1)
-

that $o_{t_1}^i(t_0^i) = o_{t_1}^j(t_0^j)$ and $o_{t_2}^i(t_0^i) \neq o_{t_2}^j(t_0^j)$ for any cyclic rotation phases t_0^i and t_0^j . We denote such sequences \mathbf{e}^i as *regular sequences*. In Algorithm 1 we develop an algorithm to generate regular sequences.

Lemma 2.3. *The sequence generated by Algorithm 1 is regular.*

Step 3: Generating phone pick sequence

Each player i then constructs his phone pick sequence \mathbf{u} based on the regular sequence \mathbf{o}^i whose length is $L_o \triangleq 4(l + 2l')$. To that end, let p_i^0 and p_i^1 denote the two smallest prime numbers not smaller than N_i and co-prime with L_o , i.e., $p_i^1 > p_i^0 \geq N_i$. The phone pick sequence of player i , \mathbf{u} , is constructed as follows:

$$u_t = \begin{cases} \lfloor t \rfloor_{p_i^0} & o_t^i = 0 \text{ and } \lfloor t \rfloor_{p_i^0} < N_i, \\ \lfloor t \rfloor_{p_i^1} & o_t^i = 1 \text{ and } \lfloor t \rfloor_{p_i^1} < N_i, \\ \text{rand}(N_i - 1) & \text{otherwise,} \end{cases} \quad (2.1)$$

where $\lfloor x \rfloor_y$ denotes $x \bmod y$, $\text{rand}(N_i - 1)$ denotes a random integer in $[0, N_i - 1]$. It can be noted that the period of the phone pick sequence \mathbf{u} is $L_o p_i^0 p_i^1$ without taking into account the randomly chosen telephones. Fig. 2.2 provides an example of the phone pick sequence in Zero-knowledge Rendezvous.

ID $\alpha=01$, Padded ID $\mathbf{a}=0011$, $N_a=3$, $p_a^0=3$, $p_a^1=5$, $t_0^a=0$	Round index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...
	\mathbf{o}^a :	0	0	1	1	0	0	1	1	0	1	0	1	0	1	0	1	...
	\mathbf{u} :	0	1	2	r	1	2	1	2	2	r	1	1	0	r	2	0	...

Figure 2.2: A phone pick sequence example for Alice (r denotes a randomly chosen telephone).

The overall algorithm

Algorithm 2 summarizes the majors steps to construct the phone pick sequence in the proposed Zero-knowledge Rendezvous algorithm by taking Alice as an example (the same steps hold for Bob to construct \mathbf{v}).

2.2.5.2 Correctness and Worst-case Rendezvous Delay Bound

In the following theorem by applying the Chinese Remainder Theorem (proof detailed in [40]), we prove the correctness of Zero-knowledge Rendezvous in guaranteeing rendezvous and establish the worst-case rendezvous delay bound.

Theorem 2.3 (Correctness and worst-case rendezvous delay bound). *Zero-knowledge Rendezvous can ensure rendezvous between Alice and Bob regardless of their telephone labeling functions and cyclic rotation phases. Let p_a^1 and p_b^1 denote the larger prime numbers chosen by Alice and Bob, the worst-case rendezvous delay is upper-bounded by $L_o p_a^1 p_b^1$.*

Remark. *Asymptotically, it follows from Theorem 2.3 that the worst-case rendezvous delay of Zero-knowledge Rendezvous approaches $L_o N_a N_b$, or $O(N^2)$ if $N_a \simeq N_b \simeq N$ which approaches the established rendezvous delay lower-bound established in Theorem 2.1 in Sec. 2.3.4.*

Zero-knowledge Rendezvous can be adapted and simplified if players have pre-assigned roles, such as in half-duplex communication. Specifically, each player is either a caller or a callee, and a rendezvous is required between a caller and a callee. In such context, we can attribute a one-bit ID $\mathbf{a} = 0$ for any caller and an one-bit ID $\mathbf{b} = 1$ for any callee. Following the same analysis, we can show that the worst-case rendezvous delay of Zero-knowledge Rendezvous using the two one-bit IDs is upper-bounded by $p_a^1 p_b^1$, a factor of L_o shorter than the case without pre-assigned roles.

Zero-knowledge Rendezvous can also be adapted and simplified if players are synchronized, i.e., they start the rendezvous search at the same time. In this case, we do not need to pad the ID sequences of the two players to ensure that they are cyclic rotationally distinct one to the other. The sequences o^i can be generated directly using the ID sequence. In terms of rendezvous delay, the worst-case delay is upper-bounded by $4l p_a^1 p_b^1$ in the synchronized case, which is 50% shorter than the case without pre-assigned roles.

To complete our study, we also show in [40] that the average rendezvous delay of Zero-knowledge Rendezvous is upper-bounded by $\frac{L_o p_a^1 p_b^1}{2}$, and asymptotically when $p_i^0 \simeq p_i^1 \simeq N_i \simeq N$, it can be bounded by $\frac{L_o N^2}{2}$.

2.2.6 Theoretic Performance Bound for Probabilistic Strategies

In this subsection, we investigate the probabilistic rendezvous strategies in the generalized telephone problem to provide a comparison reference for the deterministic strategy we develop. By playing a probabilistic strategy, a player picks a telephone each round based on certain probability so as to minimize the expected rendezvous delay. Note that the rendezvous delay cannot be bounded under probabilistic strategies.

Specifically, we derive the expected rendezvous delay for three representative probabilistic strategies, based on which we make a conjecture on the minimum expected rendezvous delay of probabilistic strategies.

2.2.6.1 Purely Random Strategy

We start by investigating the simplest probabilistic strategy, the purely random strategy where each player i picks each telephone $h \in \mathcal{N}_i$ with equal probability $\mu_i(h)$, i.e., $\mu_i(h) = \frac{1}{N_i}$. This

process is repeated each round. Let c^* denote the only telephone via which the two players can rendezvous, the probability that both Alice and Bob picks it in a round is $\frac{1}{N_a N_b}$. The expected rendezvous delay, denoted as d , is thus

$$d = \sum_{t=1}^{\infty} t \left(1 - \frac{1}{N_a N_b}\right)^{t-1} \frac{1}{N_a N_b} = N_a N_b,$$

Asymptotically when $N_a \simeq N_b \simeq N$, $d \simeq O(N^2)$.

2.2.6.2 Anderson-Weber Strategy

We next analyse the Anderson-Weber strategy developed in [15] addressing the original version of the Telephone Problem. To recapture, under the Anderson-Weber strategy, each player i sticks to a telephone for N_i rounds with probability θ and hops across a random permutation of the N_i telephones with probability $1 - \theta$. The process is repeated each N_i rounds until rendezvous. To make the analysis tractable on the expected rendezvous delay d as in their paper [15], we consider the case where $N_a = N_b = N$ and both players begin the search simultaneously. The reasons why we focus on this simplified case are two-fold:

- Dropping either assumption makes the analytical characterization of the expected rendezvous delay intractable;
- The simplified case is sufficient to provide order-magnitude performance bound on this strategy.

To derive the expected rendezvous delay d of the Anderson-Weber strategy, we consider the following three cases:

Case 1: With probability θ^2 , both players stick to a random telephone for N rounds. In this case, with probability $\frac{1}{N}$, a player picks the right telephone c^* that can make them rendezvous. Hence with probability $\frac{1}{N^2}$, rendezvous can be achieved with a delay 1 and with probability $1 - \frac{1}{N^2}$, rendezvous cannot be achieved within N rounds, thus resulting an expected rendezvous delay $N + d$. The expected rendezvous delay in this case is thus $\frac{1}{N^2} + (1 - \frac{1}{N^2})(N + d)$.

Case 2: With probability $2\theta(1 - \theta)$, one player sticks to a telephone for N rounds and the other player hops across the N telephones in his room. In this case, with probability $\frac{1}{N}$, the player sticking to a random telephone chooses the telephone c^* that makes them rendezvous, resulting in an expected rendezvous delay $\frac{N}{2}$. With the complementary probability $1 - \frac{1}{N}$, rendezvous cannot be achieved within N rounds, resulting in an expected rendezvous delay $N + d$. The expected rendezvous delay in this case is thus $\frac{1}{N} \cdot \frac{N}{2} + (1 - \frac{1}{N})(N + d)$.

Case 3: With probability $(1 - \theta)^2$, both players hop across the N telephones in his room in a random permutation. In This case, it can be calculated that the probability that they can rendezvous within N rounds is

$$\frac{\binom{N}{1} [(N-1)!]^2}{(N!)^2} = \frac{1}{N}$$

with an average rendezvous delay $\frac{N}{2}$. With the complementary probability $1 - \frac{1}{N}$, rendezvous cannot be achieved within N rounds, thus resulting an expected rendezvous delay $N + d$. The expected rendezvous delay in this case is thus $\frac{1}{N} \cdot \frac{N}{2} + (1 - \frac{1}{N})(N + d)$.

Based on the above analysis, we can established the following equation:

$$d = \theta^2 \left[\frac{1}{N^2} + \left(1 - \frac{1}{N^2}\right) (N + d) \right] + 2\theta(1 - \theta) \left[\frac{1}{2} + \left(1 - \frac{1}{N}\right) (N + d) \right] + (1 - \theta)^2 \left[\frac{1}{2} + \left(1 - \frac{1}{N}\right) (N + d) \right].$$

After some algebraic operations, we have

$$d = \frac{N[N + \frac{(1-\theta)^2}{2}]}{1 - \theta^2} - N.$$

The minimum expected rendezvous delay can be derived as $d_{min} = N^2 - \frac{N}{2}$ when $\theta = 0$. The Anderson-Weber strategy achieving minimum expected rendezvous delay is a Markovian strategy where each player repeatedly picks all the telephones sequentially following a random permutation of them. Asymptotically, we have $d \simeq O(N^2)$.

2.2.6.3 Impatient Markovian Strategy

The results on the delay of the Anderson-Weber strategy in the generalized telephone problem demonstrate that it is better to explore than sticking to one telephone. Based on this observation, we investigate a natural strategy aiming at further decreasing the rendezvous delay by repeatedly exploring a subset of αN ($0 \leq \alpha \leq 1$) telephones. We term this strategy as the *impatient Markovian strategy* where α characterizes the degree of patience of the players. When $\alpha = 1$, the impatient Markovian strategy becomes a patient strategy and degenerates to the Anderson-Weber strategy with $\theta = 0$.

Specifically, the impatient Markovian strategy works in an epoch-based way in which each epoch is composed of αN rounds. In each epoch, each player randomly picks a subset of αN telephones sequentially following a random permutation of them. Such operation is repeated until when the rendezvous is achieved.

We now derive the expected rendezvous delay for the above impatient Markovian strategy. As in the previous subsection, we consider the case where $N_a = N_b = N$ and both players begin the search simultaneously. Let π denote the probability that the rendezvous can be achieved in an epoch. We can compute π as

$$\pi = \frac{C(\alpha N, 1)[P(N - 1, \alpha N - 1)]^2}{[P(N, \alpha N)]^2} = \frac{\alpha}{N},$$

where $P(n_1, n_2) \triangleq \frac{n_1!}{(n_1 - n_2)!}$ denotes the number of permutations of n_2 elements from a set of n_1 . We further consider the following two cases:

- *Case 1:* With probability π , the rendezvous can be achieved with an expected delay of $\frac{\alpha N}{2}$.
- *Case 2:* With probability $1 - \pi$, the rendezvous cannot be achieved within the current epoch, thus resulting an expected rendezvous delay $\alpha N + d$.

It follows that

$$D = \pi \cdot \frac{\alpha N}{2} + (1 - \pi)(d + \alpha N).$$

We can thus solve d as $d = N(N - \frac{\alpha}{2})$, which is minimized at $\alpha = 1$ with the minimum $N^2 - \frac{N}{2}$. In this case, patience actually leads to minimum rendezvous latency.

2.2.6.4 Discussion

It is insightful to compare the rendezvous delay of probabilistic strategies in the generalized telephone problem to the original version.

Globally, we note that the expected rendezvous delay increases from $O(N)$ in the original version to $O(N^2)$ in the generalized version. Such difference in rendezvous delay can be intuitively explained by the problem setting that in the original Telephone Problem, there are N telephones that connecting the players compared to only one such telephone in the Generalized Telephone Problem. Consequently, the expected rendezvous delay in the original problem is one order smaller than that in the generalized one.

Specifically, the Anderson-Weber strategy is shown to be optimum for $N = 2, 3$ in the original Telephone Problem and has a asymptotical delay of $0.83N$, compare to N for the purely random strategy. In the generalized Telephone Problem, the Anderson-Weber strategy that minimizes the expected rendezvous delay degenerates to the Markovian strategy of repeatedly picking the N telephones sequentially by following a random permutation of them. This difference can be explained by noticing that in the original problem, the best situation for the players is that one follows the *wait-for-mummy* strategy by sticking to a telephone while the other follows the strategy of *exploration* by picking his telephones sequentially. This setting can guarantee rendezvous with an expected delay $\frac{N}{2}$. However, since the players cannot coordinate to achieve such setting, and the situation with both players choosing *wait-for-mummy* leads to longer expected rendezvous delay than the situation with both of them exploring, the optimum strategy consists of sticking a balance with a probability $\theta = 0.25$ in the asymptotic case corresponding to a larger probability of exploration. The situation is different in the generalized problem. For Alice, if Bob chooses the *wait-for-mummy* strategy, it follows the analysis on the previous analysis on the Anderson-Weber strategy that Alice is better off to choose exploration because *wait-for-mummy* leads to a rendezvous probability $\frac{1}{N^2}$ with an average delay of 1 round while exploration leads to a rendezvous probability $\frac{1}{N}$ with an average delay of $\frac{N}{2}$. If Bob chooses to explore, regardless of the strategy of Alice (i.e., *wait-for-mummy* or *exploration*), the rendezvous is achieved with probability $\frac{1}{N}$ with an average delay of $\frac{N}{2}$. Therefore, in the generalized problem, each player is always better off exploring than waiting for mummy. The optimum Anderson-Weber strategy in this case thus degenerates to the Markovian strategy of repeatedly exploring the N telephones.

Furthermore, our results on the impatient Markovian strategy demonstrate that the full exploration minimizes the expected rendezvous delay.

We can summarize our theoretical results on probabilistic strategies in the Generalized Telephone Problem as follows:

- Exploration outperforms *wait-for-mummy*;
- A full exploration outperforms a partial one.

The two findings motivate us to make the following conjecture, the proof or counter-examples of which is far from trivial² and consists of an interesting direction for future research.

Conjecture 2.1. *In the generalized telephone problem, the Markovian strategy of picking all the telephones sequentially by following a random permutation of them minimizes the expected rendezvous delay.*

²Note that even in the original Telephone Problem, it remains an open problem to find the optimum probabilistic strategy that minimizes the expected rendezvous delay.

2.2.7 Variants and Extensions

The Generalized Telephone Problem and the rendezvous algorithm can generate many related searching and rendezvous problems that are interesting and applicable in many engineering domains. In this subsection we review and investigate a number of them.

2.2.7.1 Trading off Worst-case and Average Rendezvous Delay

It can be easily calculated that the purely random probabilistic rendezvous algorithm achieves an average rendezvous delay of $N_a N_b$. However, the worst-case rendezvous delay of a probabilistic algorithm cannot be bounded. On the other hand, the rendezvous delay of the deterministic algorithm Zero-knowledge Rendezvous has a larger average rendezvous delay of $\frac{L_o p_a^1 p_b^1}{2}$ as analysed in Sec. 2.2.5 with the worst-case rendezvous delay bounded by $L_o p_a^1 p_b^1$. A natural question is *how to improve the average performance of Zero-knowledge Rendezvous while still ensuring a bounded rendezvous delay*.

We next investigate how a desired tradeoff between the worst-case and the average rendezvous delay can be achieved by properly choosing the two prime numbers p_0^i and p_1^i . To make our analysis tractable, we focus on a synchronized case where $N_a = N_b = N$ and Alice and Bob choose the same prime numbers. However, the idea presented via this example also holds in the general cases. Recall the phone pick sequence of Zero-knowledge Rendezvous (equation (2.1)), we note that for rounds $N_i \leq [t]_{p_i^0} \leq p_i^0$ (when $o_t^i = 0$) and $N_i \leq [t]_{p_i^1} \leq p_i^1$ (when $o_t^i = 1$), each player randomly picks a telephone. We can configure the duration of such “random periods” via p_i^0 and p_i^1 so as to improve the average performance while still ensuring the bounded rendezvous delay by the operations in the remaining “deterministic rounds”. Specifically, choosing larger p_i^0 and p_i^1 results in longer “random periods”, thus improving the average performance at the price of increasing the worst-case delay. By choosing proper p_i^0 and p_i^1 , we can trade off the worst-case and the average rendezvous delay.

We next provide an approximative quantitative analysis on the above tradeoff. Consider the case where N is sufficiently large and $p_i^0 \simeq p_i^1 \simeq p$. Approximatively, within each p rounds, there are $p - N$ “random rounds” where player i randomly picks a telephone. We call such $p - N$ “random rounds” a random frame. The probability that a rendezvous can be achieved within one random frame can be calculated as

$$q = 1 - \left(1 - \frac{1}{N^2}\right)^{p-N}.$$

Recall that the worst-case delay is bounded by $L_o p^2$ rounds, i.e., $L_o p$ random frames, we can then calculate the upper-bound of the average rendezvous delay \bar{d} as follows:

$$\bar{d} \leq q \cdot p + (1 - q)q \cdot 2p + \cdots + (1 - q)^{L_o p - 2} q \cdot (L_o p - 1)p^2 + (1 - q)^{L_o p - 1} \cdot L_o p^2.$$

Given a target expected delay bound \bar{d} , p can be chosen based on the above inequality. To get more insight, we consider the case where we set p sufficiently larger than N but linear to N , i.e., $p = (1 + \lambda)N$ with sufficiently large λ . We have $q \simeq \frac{\lambda}{N}$. After some algebraic operations, the average delay is bounded by

$$\bar{d} < \sum_{k=0}^{\infty} (1 - q^k) q \cdot (k + 1)p = \left(1 + \frac{1}{\lambda}\right) N^2,$$

with the worst-case rendezvous delay $L_o(1 + \lambda)^2 N^2$. We can clearly see the possibility of trading-off worst-case and average rendezvous delay in Zero-knowledge Rendezvous by configuring λ .

2.2.7.2 The Case of Picking Multiple Telephones Simultaneously

Throughout our analysis, we implicitly assume that each player can pick only one telephone at a time. We now relax this constraint by considering the case where each player can pick two telephones simultaneously. The analysis can be generalized to the case where each player can pick an arbitrary number of telephones. In a broader sense of a rendezvous or searching game, relaxing the constraint of picking only one telephone a time allows to study the case where each player can search more than one places simultaneously or where the game is played between two groups of players and the communication is possible among the players in the same group.

We first establish the lower-bound of the worst-case rendezvous delay. To this end, by performing a similar analysis as that in the proof of Theorem 2.1 [40], we can show that the worst-case rendezvous delay, among all possible telephone labeling functions and all cyclic rotation phases, cannot be lower than $\frac{N_a N_b}{4}$, and more generically $\frac{N_a N_b}{n^2}$ if each player is allowed to pick $2n$ telephones simultaneously.

We then devise a deterministic rendezvous algorithm achieving $O(N_a N_b)$ worst-case rendezvous delay. Specifically, we take Alice as an example to derive the phone pick sequence. Since she can pick two telephones each time, we denote $\mathbf{u}_l = \{u_{t,l}\}$ and $\mathbf{u}_r = \{u_{t,r}\}$ the sequences of telephones picked by her left and right hands, respectively. We devise $\mathbf{u}_l = \{u_{t,l}\}$ as follows:

$$u_{t,l} = \begin{cases} [t]_{p_a^l} & [t]_{p_a^l} < N_a, \\ \text{rand}(N_a - 1) & \text{otherwise,} \end{cases}$$

where p_a^l is a prime number not smaller than N_a . The sequence $\mathbf{u}_r = \{u_{t,r}\}$ can be devised symmetrically using a different prime number p_a^r . To establish that the rendezvous is guaranteed between Alice and Bob, it suffices to note that at least one of p_a^l and p_a^r is co-prime to at least one of p_b^l and p_b^r , the two prime numbers of the phone pick sequences of the left and right hands of Bob. The worst-case rendezvous delay can be proved to be upper-bounded by $p_a^r p_b^r$ if we specify that the prime number of the right hand is larger than that of the left hand.

When each player can pick two telephones simultaneously, a symmetry breaking technique is no more necessary because the phone pick sequence of the left hand can be designed in a different way as that of the right hand, thus naturally breaking the symmetry and resulting in a rendezvous between the left-hand and the right-hand sequences. Consequently, a shorter worst-case rendezvous delay, which is a factor L_o shorter than baseline case, can be achieved.

2.3 Neighbor Discovery

2.3.1 Context and Motivation

The supporting primitive that discovers all the neighbors in a device's communication range is referred to as *neighbor discovery*, which is one of the bootstrapping primitives supporting many basic network functionalities, such as topology control, clustering, medium access control, etc.

Ideally, nodes should discover their neighbors as quickly as possible for other algorithms to quickly start their execution.

Compared to the channel rendezvous problem, designing efficient neighbor discovery algorithms is more challenging in energy-constraint wireless networks because the technique of duty cycling is used to reduce the energy consumption when these devices are in the idle state. Under duty cycling, each device alternates between active and sleeping modes by turning their radios on only periodically to achieve synchronization and save energy. *Duty cycle* refers to the fraction of time a device is in the active mode [14, 89]. For example, a device whose duty cycle is 1% activates during one time slot every 100 slots. Despite its effectiveness in energy conservation, the duty cycling technique significantly challenges the neighbor discovery algorithm design in the quest of limiting discovery latency with low power consumption. Specifically, the two important design objectives, saving energy through a duty-cycle based scheduling and limiting the neighbor discovery latency, are at odd with each other. The problem is more complex if the operating frequencies of wireless nodes span multiple channels.

We investigate the neighbor discovery problem in the multi-channel duty-cycled wireless networks and develop a neighbor discovery algorithm MCD with the following properties.

- *Fine-grained control of duty cycle*: In contrast to existing solutions using prime numbers or power-multiples, MCD can support more than 95% of duty cycles in practical settings, thus providing much more fine-grained control of energy conservation levels.
- *Bounded worst-case discovery delay*: MCD achieves bounded discovery delay even between nodes with heterogeneous duty cycles.
- *Full discovery diversity*: MCD guarantees discovery over *each* channel, thus minimizing the probability of discovery failures caused by various channel problems.
- *Robustness against asymmetrical channel perceptions*: MCD achieves the same discovery performance even if nodes have asymmetrical channel perceptions, either on the accessible channel set or on the channel index.
- *Robustness against clock drift*: MCD achieves the same performance even if clocks of any two nodes drift away from each other by an *arbitrary* amount of time.

2.3.2 Related Work

The neighbor discovery algorithms for duty-cycled networks in the literature can be categorized into *probabilistic* and *deterministic* algorithms. We give a high-level overview of these two types of approaches and briefly analyse the pros and cons of each.

Probabilistic algorithms (cf. major work in this category [107, 139, 183, 197, 232]) adopt probabilistic strategies at each node. Specifically, each node remains active or asleep with different probabilities. A representative one is the birthday algorithm [139] where nodes transmit/receive or sleep with different probabilities. Probabilistic algorithms have the advantages of being memoryless and stationary and thus are especially robust and suitable in decentralized environments where no prior knowledge or coordination is available. Moreover, they usually perform well in the average case by limiting the expected discovery delay. The main drawback of them is the lack of performance guarantee in terms of discovery delay. This problem is referred to as the long-tail discovery latency problem in which two neighbor nodes may experience extremely long delay before discovering each other.

Deterministic algorithms, on the other hand, can provide strict upper-bound on the worst-case discovery delay (cf. major work in this category [20, 67, 103, 106, 114, 140, 194, 210, 238]). In deterministic algorithms, each node wakes up according to its neighbor discovery schedule carefully tuned to ensure that each pair of two wake-up schedules overlap in at least one active slot. The key element in the deterministic algorithm design is how to devise the neighbor discovery schedule to ensure discovery and minimize the worst-case discovery delay, regardless of the duty cycle asymmetry and the relative clock drift. Compared to probabilistic approaches that work well in the average case while fail to bound the worst-case discovery delay, deterministic algorithms have good worst-case performance while usually have longer expected discovery delay.

More specifically, based on the design of wake-up schedule, major existing deterministic algorithms can further be divided into three classes as briefly reviewed in the following.

- The first class of them, based on Quorum [114, 194], construct the wake-up schedule by assigning a column and a row of an $m \times m$ array to each node such that no matter which row and column are selected, any two nodes have at least two overlapping awoken slots. The main drawback of the Quorum-based approaches is the support of only symmetrical duty cycles [194]. Although enhanced solutions have been proposed to support asymmetric duty cycles, only two different duty cycles can be supported [114].
- The second class of deterministic algorithms overcome this limitation by using prime numbers to guarantee bounded discovery delay even for asymmetrical duty cycles. A typical one in this class is Disco [67], in which each node selects two prime numbers, based on which its wake-up schedule is configured. A more recent solution, U-Connect [106], uses a single prime number per node and has a shorter discovery delay, given the same duty cycle.
- The third class, Searchlight, proposed in [20] and a number of follow-up schemes [184, 210], employs two kinds of wake-up slots, termed as anchor and probe slots, to achieve both lower worst-case and average discovery delay.

Despite extensive research efforts devoted to neighbor discovery, none of them can solve the multi-channel neighbor discovery problem by achieving bounded discovery delay for nodes operating on heterogenous duty cycles. In contrast, the solution we develop can achieve bounded discovery delay for nodes operating on heterogeneous duty cycles in multi-channel environments.

2.3.3 Multi-channel Neighbor Discovery

2.3.3.1 System Model

We consider a time-slotted (but not necessarily synchronized) energy-constraint wireless network operating on a set \mathcal{N} of $N \triangleq |\mathcal{N}|$ frequency channels. To discover its neighbors in the multi-channel environment, each node wakes up periodically and switches across different channels. The main design challenges we need to address are: (1) lack of clock synchronization, (2) asymmetrical duty cycles and (3) asymmetrical channel perceptions.

In the following, we formally define the *neighbor discovery schedule* that characterizes the wake-up and channel hopping pattern of a node.

Definition 2.5 (Neighbor discovery schedule). *The neighbor discovery schedule of a node u is defined*

as a sequence $\mathbf{x}_u \triangleq \{x_u^t\}_{1 \leq t \leq T_u}$, where T_u is the period of the sequence³, and

$$x_u^t = \begin{cases} 0 & u \text{ sleeps in slot } t \\ h \in \mathcal{N}_u & u, \text{ operating on channel } h, \text{ wakes up} \end{cases}.$$

Definition 2.6 (Duty cycle). The duty cycle of a node u , denoted by δ_u , is defined as the percentage of slots per period of the neighbor discovery schedule \mathbf{x}_u where u is active. Formally, δ_u is defined as

$$\delta_u \triangleq \frac{|\{t \in [0, T_u - 1] : x_u^t = 1\}|}{T_u}.$$

The reciprocal of δ_u is denoted by d_u .

Consider two nodes a and b with their neighbor discovery schedules being \mathbf{x}_a and \mathbf{x}_b whose periods are T_a and T_b . Given the periodicity of \mathbf{x}_a and \mathbf{x}_b , it suffices to consider consecutive $T_a T_b$ slots, i.e., $1 \leq t \leq T_a T_b$. If $\exists t \in [1, T_a T_b]$ and $h \in \mathcal{N}$ such that $x_a^t = x_b^t = h$, we say that a and b can discover each other in slot t on channel h . Slot t is called the discovery slot and channel h is called the discovery channel between a and b . Example 3 illustrates the above definition.

Example 3. Consider a network of two channels and two nodes a, b whose neighbor discovery schedules are $\mathbf{x}_a = \{0, 0, 1\}$ and $\mathbf{x}_b = \{0, 1, 0, 2\}$ with $T_a = 3$ and $T_b = 4$. The duty cycles of a and b are $\delta_a = \frac{1}{3}$ and $\delta_b = \frac{1}{2}$, or $d_a = 3$, $d_b = 2$. The neighbor discovery schedules of a and b are repeated each 12 slots, as illustrated in Fig. 2.3 for one period. We can observe that a and b can discover each other in slots 6 on channel 1.

Slot index	1	2	3	4	5	6	7	8	9	10	11	12	...
Node a :	0	0	1	0	0	1	0	0	1	0	0	1	...
Node b :	0	1	0	2	0	1	0	2	0	1	0	2	...

Figure 2.3: Neighbor discovery schedule example.

As in the previous section, to model the situation where the clocks of different nodes are not synchronized⁴, we apply the concept of *cyclic rotation* to neighbor discovery schedules. Specifically, given a neighbor discovery schedule \mathbf{x}_a , we denote $\mathbf{x}_a(k)$ a cyclic rotation of \mathbf{x}_a by k slots where k is called the cyclic rotation phase. In Example 3, we have $\mathbf{x}_a(1) \triangleq \{1, 0, 0\}$ and $\mathbf{x}_b(2) = \{0, 2, 0, 1\}$.

2.3.3.2 Optimum Multi-channel Neighbor Discovery

Performance Metric 1: Maximum Time to Discovery. In neighbor discovery, the primary performance metric is the maximum time to discovery (MTTD), i.e., the worst-case discovery delay. Given two nodes a and b , the MTTD between them is defined as the upper-bound of the latency (in number of slots) before successful mutual discovery for all possible clock drift between them. Reconsider Example 3, we can observe that the MTTD is 11, achieved between $\mathbf{x}_a(6)$ and $\mathbf{x}_b(6)$.

³A random neighbor discovery schedule is a special case where $T_u \rightarrow \infty$.

⁴Here we assume that clocks of different nodes are asynchronous but their slot boundaries are aligned. The situation of unaligned slot boundaries is analysed in Sec. 2.3.6.4.

Performance Metric 2: Discovery Diversity. The second metric, particularly pertinent for the multi-channel environment, is the *discovery diversity*, which characterizes the capability of a neighbor discovery algorithm of discovering a neighbor regardless of its operational channel. We say that a neighbor discovery algorithm achieves *full* discovery diversity if the discovery of any pair of nodes is guaranteed on *every* common channel they can access. The neighbor discover schedule in Example 3 cannot achieve full discovery diversity as a and b can never discover each other on channel 2.

Performance Metric 3: Maximum Time to Discovery with Full Diversity. When full discovery diversity can be achieved, we further define the third metric *maximum time to discovery with full diversity* (MTTD-FD) as the worst-case delay to achieve full discovery diversity. The MTTD-FD can be regarded as a generalization of the MTTD in multi-channel networks. The MTTD-FD degenerates to the MTTD in single-channel networks. Throughout our analysis, we analyse the MTTD in single-channel case and the MTTD-FD in multi-channel case.

We next formulate the optimum multi-channel neighbor discovery problem.

Problem 2.4 (Multi-channel neighbor discovery). *The optimum multi-channel neighbor discovery problem is defined as follows:*

minimize T ,

subject to $\forall t_a^0 \in [1, T_a], t_b^0 \in [1, T_b], \forall \delta_a, \delta_b, \exists t \leq T$ such that $x_a^t(t_a^0) = x_b^t(t_b^0) = h, \forall h \in \mathcal{N}_a \cap \mathcal{N}_b$.

That is, devising neighbor discover schedules to minimize T , the worst-case discovery delay while achieving full discovery diversity between any pair of nodes a and b for any duty cycle pair (δ_a, δ_b) , any initial time offset t_a^0 and t_b^0 , and any channel perception \mathcal{N}_a and \mathcal{N}_b .

In what follows, we first establish a theoretical performance bound of any neighbor discovery algorithm. We then present the baseline design and optimization of MCD in the single-channel case, before proceeding to the multi-channel case with symmetrical channel perception (i.e., $\mathcal{N}_a = \mathcal{N}_b$). We complete our analysis by addressing the generic case with asymmetrical channel perceptions and arbitrary clock drift to iron out a version of MCD that works in practice.

2.3.4 Algorithm-independent Discovery Delay Bound

Armed with the theoretical framework established previously, the following theorem derives the performance bound of any multi-channel neighbor discovery algorithm achieving full discovery diversity. This result thus establishes the lower-bound of the solution of Problem 2.4.

Theorem 2.4. (Algorithm-independent Bound of MTTD-FD) *For any neighbor discovery algorithm achieving full discovery channel diversity, the MTTD-FD between any pair of nodes a and b , denoted by L , is lower-bounded by $N^2 d_a d_b$, where d_a and d_b denote the reciprocals of the duty cycles of a and b , i.e., $d_a = \frac{1}{\delta_a}$ and $d_b = \frac{1}{\delta_b}$. Asymptotically, when $d_a \simeq d_b \simeq d$, $L \simeq N^2 d^2$.*

Theorem 2.4 can also be viewed from another angle: to achieve a target MTTD-FD L , the duty cycle reciprocal d should be upper-bounded by $O\left(\frac{\sqrt{L}}{N}\right)$. Consequently, the energy consumption cannot be lower than $O\left(\frac{\sqrt{L}}{N}\right)$.

2.3.5 MCD: Single-channel Case

2.3.5.1 Motivation and Algorithm Design

In the single-channel case, the neighbor discovery schedule \mathbf{x}_u for each node u degenerates to a binary sequence where

$$x_u^t = \begin{cases} 1 & u \text{ wakes up in slot } t, \\ 0 & u \text{ sleeps in slot } t. \end{cases}$$

Each node wakes up periodically to discover its neighbors. The wake-up period is determined by its duty cycle. Specifically, we consider two neighboring nodes a and b with duty cycles $\delta_a = \frac{1}{d_a}$ and $\delta_b = \frac{1}{d_b}$. To discover each other, nodes a and b wake up every d_a and d_b slots, i.e., $x_a(t) = 1$ for $t = kd_a$ and $x_b(t) = 1$ for $t = kd_b + \delta_{ab}$ where δ_{ab} is the clock offset between a and b , $k = 1, 2, \dots$. It follows from the Chinese Remainder Theorem [152] that if d_a and d_b are co-prime to each other, the two nodes are ensured to discover each other regardless of δ_{ab} , i.e., there exists t_d such that $x_a^{t_d} = x_b^{t_d}(\delta_{ab}) = 1, \forall \delta_{ab}$.

However, assigning co-prime numbers to each node in a distributed way is far from trivial. A commonly adopted solution is to use only prime numbers because two distinct prime numbers are by definition co-prime to each other, as in Disco [67] and U-Connect [106]. However, limiting the choices to prime numbers fail to support all the duty cycles due to the limited number of prime numbers. Note that among natural numbers smaller than 1000, only $\frac{1}{6}$ are prime numbers.

Motivated by the above analysis, we devise the following neighbor discovery schedule in MCD. For each node u with duty cycle $\delta_u = \frac{1}{d_u}$,

$$x_u(t) = \begin{cases} 1 & t \text{ is divisible by either } 2d_u - 1 \text{ or } 2d_u + 1, \\ 0 & \text{otherwise.} \end{cases}$$

Example 4. Consider two nodes a and b with duty cycles $\delta_a = \frac{1}{3}$, $\delta_b = \frac{1}{5}$ with a clock offset $\delta_{ab} = 1$. Under MCD, using the time of a as reference, a wakes up in slots $5k$ and $7k$, i.e., 5, 7, 10, 14, 15, 20, 21, \dots , b wakes up in slots $9k + 1$ and $11k + 1$, i.e., 10, 12, 19, 23, \dots , as illustrated in Fig. 2.4. The discovery happens in slot 10.

Slot index	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	...
Node a :	0	0	0	0	1	0	1	0	0	1	0	0	0	1	1	0	0	0	0	1	1	0	0	0	1	...
Node b :	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	...

Figure 2.4: MCD in single-channel case: $d_a = 3$, $d_b = 5$.

The period of \mathbf{x}_u in MCD is $(2d_u - 1)(2d_u + 1)$, in which there are $4d_u - 1$ active slots⁵. Hence, the actual average duty cycle, denoted as $\hat{\delta}_u$, is $\frac{4d_u - 1}{(2d_u - 1)(2d_u + 1)}$ which approaches to the required duty cycle $\delta_u = \frac{1}{d_u}$ when d_u is large. Generally, the relative error between $\hat{\delta}_u$ and δ_u is upper-bounded by $\frac{1}{4d_u}$, as established in the following lemma.

Lemma 2.4. The relative error between the duty cycle of the neighbor discovery schedule $\hat{\delta}_u$ and the required duty cycle δ_u is upper-bounded by $\frac{1}{4d_u}$.

⁵Note that $(2d_u - 1)(2d_u + 1)$ is divisible by both $2d_u - 1$ and $2d_u + 1$.

2.3.5.2 MCD Core Idea: Regular Duty Cycles

Following the Chinese Remainder Theorem, the mutual discovery of two neighbor nodes a and b in MCD, regardless of their clock drift, requires at least one of $2d_a \pm 1$ to be co-prime with at least one of $2d_b \pm 1$. In the vast majority of cases, this requirement can be satisfied. To illustrate this, if we allow the maximum duty cycle reciprocal D to be 100, then all duty cycles $\frac{1}{d}$ except $d = 17$ and 38 can be supported by MCD; if we allow D to be 1000, only 43 duty cycles cannot be supported, i.e., MCD can support nearly 96% of all duty cycles.

In this subsection, we conduct a formal analysis on the design idea of MCD. We start by formulating the definition of *regular duty cycles* that are natively supported by MCD.

Definition 2.7 (Regular Duty Cycle). *Given the duty cycle reciprocal upper-bound D , we call a duty cycle $\delta = \frac{1}{d}$ ($d \leq D$) a regular duty cycle if for any $2 \leq d' \leq D$, at least one number from $2d \pm 1$ is co-prime with at least one number from $2d' \pm 1$.*

For two nodes a and b , if at least one of their duty cycles is regular, a and b can discover each other. Reconsider Example 4 with $D = 1000$, it follows from Definition 2.7 that the duty cycles $\delta_a = \frac{1}{3}$ and $\delta_b = \frac{1}{5}$ are both regular. Evidently the two nodes can discover each other, as illustrated in Fig. 2.4.

We conclude this subsection by stating the following properties of regular duty cycles:

- *The vast majority of duty cycles are regular.* As illustrated in Tab. 2.1, more than 97% duty cycles are regular when D varies from 100 to 900. In contrast, in existing solutions based on prime numbers, only a small portion of duty cycles can be supported due to the limited choice of prime numbers.
- *There are no three consecutive non-regular duty cycle reciprocals.* In other words, if $\frac{1}{d}$ is non-regular, at least one from $\frac{1}{d \pm 1}$ is regular. This implies that if the required duty cycle $\frac{1}{d}$ happens to be non-regular, the node can operate on $\frac{1}{d+1}$ or $\frac{1}{d-1}$. If we take the case of using $\frac{1}{d-1}$, the effective duty cycle is $\frac{4(d-1)-1}{[2(d-1)-1][2(d-1)+1]}$. The relative error to the required duty cycle can be computed as

$$\epsilon = \left| \frac{4(d-1)-1}{[2(d-1)-1][2(d-1)+1]} - \frac{1}{d} \right| / \frac{1}{d} = \frac{4(d_u-1)+1}{4(d_u-1)^2-1},$$

which is decreasing in d . In the case of $D = 1000$, the smallest non-regular duty cycle reciprocal being 17, ϵ is upper-bounded by 4.5% for all non-regular duty cycles.

Duty cycle reciprocal upper-bound D	100	300	500	700	900
Number of non-regular duty cycles	2	11	20	29	38

Table 2.1: Number of non-regular duty cycles as a function of D .

The analysis in this subsection demonstrates that MCD can support most duty cycles and even in the cases where the duty cycles cannot be directly supported, MCD can use neighboring duty cycles with almost negligible errors. Concerning the implementation of MCD, we note that the regular duty cycles can be pre-calculated off-line by exhaustive search and stored in a look-up table.

2.3.5.3 Discovery Delay Upper-bound

In the single-channel case, only the first performance metric (maximum time to discovery, MTTD) is applicable. In Theorem 2.5, we derive the MTTD of MCD between two nodes a and b if at least one of δ_a and δ_b is regular.⁶ The proof consists of applying the Chinese Remainder Theorem and the properties of the regular duty cycles [50].

Theorem 2.5 (Discovery Delay Upper-bound). *Given any two nodes a and b , if at least one of their duty cycles $\frac{1}{d_a}$ and $\frac{1}{d_b}$ is regular, they are ensured to discover each other within at most $(2d_a + 1)(2d_b + 1)$ slots.*

2.3.6 MCD: Multi-channel Case

2.3.6.1 Neighbor Discovery Schedule Construction

The neighbor discovery schedule of MCD in the multi-channel case is constructed based on its globally unique ID such as its MAC address, which can be mathematically expressed as a binary sequence of length l . The neighbor discovery schedule construction process is composed of three steps.

- *Step 1:* Each node u independently generates a padded binary sequence \mathbf{o}_u based on its ID such that the padded binary sequences of any two nodes are *cyclic rotationally distinct* one to the other;
- *Step 2:* Each node u independently generates a sequence \mathbf{s}_u based on \mathbf{o}_u such that for any two nodes a, b and any initial time offset t_a^0 and t_b^0 , there always exist four time slots l_{ij} ($i, j \in \{0, 1\}$) such that $s_a^{l_{ij}}(t_a^0) = i$ and $s_b^{l_{ij}}(t_b^0) = j$.
- *Step 3:* Each node u generates its neighbor discovery schedule based on \mathbf{s}_u .

The first two steps follow the similar procedure as those in the generalized telephone problem analyzed in the previous section. We now discuss the third step.

In the third step, the neighbor discovery schedule is constructed as follows. Each node u hops across different channels $h \in \mathcal{N}$ and wakes up based on the following schedule⁷:

$$x_u^t = \begin{cases} h & t - hd_u \text{ is divisible by } 2Nd_u \pm 1, \\ 0 & \text{otherwise,} \end{cases}$$

where $x_u^t = h$ signifies that u wakes up on channel h in slot t while $x_u^t = 0$ indicates that u sleeps in the slot, $\frac{1}{Nd_u}$ is chosen from the regular duty cycle set.

The above construction of \mathbf{x}_u does not take into account the case where there exist two different channels h^c ($c = 0, 1$) such that $t - h^0d_u$ is divisible by $2Nd_u - 1$ and $t - h^1d_u$ by $2Nd_u + 1$. To resolve such conflict, let $t' = t \bmod L_s$, u operates on channel h^c if $s_u^{t'} = c$. We refer to the slots where u operates on channel h^c in case of conflict as type- c slots.

⁶Throughout our analysis, we focus on the pair-wise discovery between any pair of neighbor nodes a and b . The obtained results can be readily generated to the network level where each node should discover all its neighbor nodes by following the same way as Theorem 2.4.

⁷To make the notation concise, we adopt the notation that $t - hd_u$ is divisible by $2Nd_u \pm 1$ denotes that $t - hd_u$ is divisible by $2Nd_u - 1$ or $2Nd_u + 1$ or both.

To intuitively see that the discovery is ensured between any pair of nodes a, b , note that if $\frac{1}{Nd_a}$ belongs to the usable duty cycle set derived previously, i.e., at least one of $2Nd_a \pm 1$ is co-prime with at least one of $2Nd_b \pm 1$, discovery can be guaranteed for any initial time offset t_a^0 and t_b^0 because there always exist four time slots l_{ij} ($i, j \in \{0, 1\}$) such that $s_a^{l_{ij}}(t_a^0) = i$ and $s_b^{l_{ij}}(t_b^0) = j$ following the second step of construction.

2.3.6.2 Discovery Delay Upper-bound

We next derive the theoretical performance of MCD in the multi-channel environment. In multi-channel case, the second metric on discovery diversity and the third metric on MTTD-FD (worst-case discovery delay with full diversity) are applicable.

Theorem 2.6. *If $\frac{1}{Nd_a}$ and $\frac{1}{Nd_b}$ are regular duty cycles, the MTTD-FD between two nodes a and b is $O(L_s N^2 \max\{d_a^2, d_b^2\})$, where L_s denotes the length of the sequences generated in Step 2.*

2.3.6.3 Robustness against Asymmetrical Channel Perception

In previous analysis, we implicitly assume that a and b have the same channel perception, i.e., they have symmetrical knowledge on \mathcal{N} . In this subsection, we relax this assumption to investigate the scenario where each node u has its own perception on \mathcal{N} , denoted by \mathcal{N}_u , which is a subset of \mathcal{N} . In this context, the neighbor discovery schedule in MCD becomes

$$x_u^t = \begin{cases} h & t - hd_u \text{ is divisible by } 2N_u d_u \pm 1, \\ 0 & \text{otherwise.} \end{cases}$$

Specifically, the channel perception asymmetry between a and b can be characterized at two levels:

- *Asymmetry on accessible channel set:* They have asymmetrical perceptions on the global channel set \mathcal{N} , i.e., $\mathcal{N}_a \neq \mathcal{N}_b$ and $\mathcal{N}_a \cap \mathcal{N}_b \neq \emptyset$;
- *Asymmetry on channel index:* They have asymmetrical perceptions on the channel index, i.e., channel $h \in \mathcal{N}$ is indexed h_a by a and h_b by b where $h_a \in \mathcal{N}_a$ and $h_b \in \mathcal{N}_b$ but $h_a \neq h_b$.

The following theorem established the performance of MCD in such context.

Theorem 2.7. *MCD under asymmetrical channel perceptions achieves the same MTTD-FD as under symmetrical channel perceptions, i.e., within at most $O(L_s \max\{N_a^2 d_a^2, N_b^2 d_b^2\})$ (specifically, $O(L_s N^2 d^2)$) if $d_a \simeq d_b \simeq O(d)$ and $N_a \simeq N_b \simeq O(N)$ slots; the discovery between a and b occurs on each channel $h \in \mathcal{N}_a \cap \mathcal{N}_b$.*

Theorem 2.7 shows that MCD is robust against asymmetrical channel perceptions, either on the channel set or index.

2.3.6.4 Robustness against Slot Non-alignment and Arbitrary Clock Drift

We then study the effect of slot non-alignment caused by relative clock drift between the neighbor nodes.

We first briefly introduce the clock model. Each node is equipped with a local clock, which is a time measurement device composed of a hardware oscillator and an accumulator. Mathematically, consider two nodes a and b , we can express the local time at b , denoted as t_b , as a function of the local time of a , denoted as t_a , by the following formula

$$t_b(t_a) = \int_{t_0}^{t_a} \rho_{ab}(\tau) d\tau + t_b(t_0),$$

where $\rho_{ab}(\tau)$ denotes the relative frequency drift rate of the oscillator of b as a function of a at time τ , $t_b(t_0)$ is the initial clock offset between them.

If a and b are ideally synchronized, it holds that $\rho_{ab}(\tau) = 1$ and $t_b(t_0) = 0$. In practice, $\rho_{ab}(\tau)$ may drift away from each other, as formalized in the following:

$$\rho_{ab} - \Delta\rho_{max} \leq \rho_{ab}(\tau) \leq \rho_{ab} + \Delta\rho_{max},$$

where $\Delta\rho_{max}$ is bounded by 10^{-6} in practice. Hence we can regard $\rho_{ab}(\tau)$ as a constant ρ_{ab} during the discovery process. Without loss of generality, we assume that the clock of b advances no slower than that of a , i.e., $\rho_{ab} \leq 1$.

When $\rho_{ab} = 1$, i.e., the clock difference between a and b remains $t_b(t_0)$, we distinguish the following two cases (to facilitate presentation, we normalize the slot duration of a):

- Case 1: $t_b(t_0) = k \in \mathbb{Z}$: this is the case with aligned slots addressed in previous analysis;
- Case 2: $t_b(t_0) = k + \delta$ with $k \in \mathbb{Z}$ and $\delta \in (-1/2, 1/2]$: the previous analysis can be directly adapted to this case, the difference being that instead of ensuring entire overlap, a discovery in this case is a partial overlap of time $1 - |\delta|$.

We now investigate the case where $\rho_{ab} < 1$, meaning that if we regard the slot duration of a as unit time, the slot duration of b is $\rho_{ab} < 1$. The following theorem establishes the discovery performance of MCD with arbitrary clock drift with $\rho_{ab} < 1$.

Theorem 2.8. *Regard the slot of a as unit time, a and b can discover each other on each channel h within at most $O(\rho_{ab}L_sN^2 \max\{d_a^2, d_b^2\})$ time.*

The results obtained in this subsection, particularly Theorem 2.8, demonstrate that the discovery performance established in previous analysis holds even when the clocks of a and b drift away from each other for an arbitrary amount of time. In other words, MCD is robust against clock drift and slot non-alignment.

2.4 Summary of Other Contributions

In this section, we present a brief summary on a selection of our other work on channel rendezvous and neighbor discovery, which can be regarded as extensions and variants of the work presented in the previous two sections.

2.4.1 From Single- to Multi-radio Channel Rendezvous

With the rapid development of the wireless communication technology and the significant decreasing prices of radios, it is nowadays feasible to equip a wireless device with multiple radios,

each operating on a separated spectrum channel. Equipping all or some nodes with multiple radios can significantly boost the network capacity by enabling simultaneous operations over multiple channels and mitigating interferences through proper channel assignment.

Motivated by the above observation, we propose an order-optimum multi-radio channel rendezvous algorithm by exploiting the benefits brought by the rendezvous diversity created by multiple radios in minimizing the rendezvous delay and increasing the rendezvous robustness. Our solution is a unified channel rendezvous algorithm that can operate in both homogenous case where both of the rendezvous nodes are equipped with only one radio or multiple radios, and heterogeneous case where one of the rendezvous nodes has single radio and the other has multiple radios.

2.4.2 Cooperative Channel Rendezvous

In this work, we exploit a long-neglected opportunity for enabling channel rendezvous in wireless networks, the cooperation among users. Specifically, we investigate how to improve the rendezvous performance by introducing the cooperative helpers that serve as “bridges” between neighboring nodes to facilitate their rendezvous process. A helper can simply relay the rendezvous requests of neighboring nodes on its operating channels such that the nodes that need to rendezvous can switch to a common channel by referring to the information contained in the rendezvous requests.

Having helpers’ relay rendezvous requests brings three benefits: (1) a greater number of channels where the rendezvous requests circulate is equivalent to the increase of rendezvous diversity; (2) the relayed rendezvous requests contain necessary information that can mitigate the rendezvous difficulty caused by the asymmetric perceptions of channel availability between two nodes; and (3) with help of the information in relayed rendezvous requests, it is easier for two nodes to find each other on a common channel at a small rendezvous latency.

Motivated by the above analysis, we develop a systematic approach for devising cooperative rendezvous algorithms and analyzing the resulting performance benefits. Specifically, we develop a mathematic framework to study the following optimization questions: How should helper nodes cooperate? How much performance gain can we obtain with cooperative rendezvous? What is the impact of the number of radios per node and per helper? Our main contributions are articulated as follows.

- *Non-cooperative rendezvous.* We first analyze the non-cooperative rendezvous algorithm with multiple radios per node. We quantify the rendezvous performance and establish performance limits in asymptotic scenarios. These results serve as a theoretic basis and comparison reference for the cooperative rendezvous algorithm.
- *Cooperative rendezvous.* We devise a cooperative rendezvous algorithm by exploiting helper nodes to facilitate rendezvous. By characterizing rendezvous delay and robustness, we quantify the benefits for rendezvous using cooperative helpers.
- *Optimization of cooperative rendezvous.* Based on the theoretical foundation, we further study the following optimization question: how many helpers are necessary to upper-bound the expected rendezvous delay? We then develop a dynamic learning algorithm enabling a helper to configure its cooperation level based on the target performance metric.

2.4.3 Oblivious Neighbor Discovery with Directional Antennas

In this work, we focus on neighbor discovery for wireless nodes with directional antennas. Compared to the traditional omni-direction antenna paradigm, neighbor discovery with directional antennas is intuitively more challenging as directional antennas can only cover a fraction of the azimuth. Hence, neighbor discovery algorithms need to be carefully designed in order to guarantee that any pair of neighbor nodes can eventually steer their antennas toward each other at certain time instance. Moreover, nodes may not be synchronized and their antennas can be heterogeneous in terms of beam-width. Neighbor discovery algorithms should be able to guarantee discovery in this challenging environment in a fully decentralized manner without any prior coordination.

We coin the term *oblivious neighbor discovery* problem to denote the following problem: *How can neighbor nodes with heterogeneous antenna beam-width and without clock synchronization discover each other within a bounded delay in a fully decentralized manner without any prior coordination?* Particularly, the following requirements should be satisfied:

- Bounded (and minimum) worst-case discovery delay;
- Discovery oblivious, the capability of guaranteeing discovery regardless of the antenna beam-width and the relative positions of nodes. This requirement is particular in the neighbor discovery with directional antennas.

We conduct a comprehensive investigation on the oblivious neighbor discovery problem, summarized as follows:

- *Theoretical framework.* We establish a theoretical framework on oblivious neighbor discovery and establish the performance bound of any oblivious neighbor discovery algorithm. Our theoretical results not only shed light on the structure of the problem, but also serve as design guidelines for oblivious neighbor discovery protocols.
- *Algorithm design.* Guided by the theoretical results, we further design an oblivious neighbor discovery algorithm and prove that it achieves guaranteed oblivious discovery with order-minimum worst-case discovery delay in the asynchronous and heterogeneous environment. We further demonstrate how the protocol can be configured to achieve a desired trade-off between average and worst-case performance.

2.5 Conclusion and Perspectives

In this chapter, we have studied channel rendezvous and neighbor discovery in decentralized networks. We have established theoretical limit of any deterministic strategy with bounded rendezvous/discovery delay that can guarantee rendezvous/discovery between two users regardless of their local parameters. We have then devised a series of distributed channel rendezvous and neighbor discovery algorithms enabling users to meet and discover each other within bounded and order-optimum delay without any prior knowledge or coordination. Our results and the method employed in the analysis form a theoretical basis of devising rendezvous strategies in a variety of engineering applications requiring bounded rendezvous delay

We conclude this chapter by outlining two research perspectives from algorithmic (theoretical) and networking (practical) angles. Algorithmically, channel rendezvous and neighbor discovery represents a typical scenario where probabilistic algorithms usually perform well in the average case but fail to provide worst-case performance guarantee, while deterministic algorithms, on

the other hand, have good worst-case performance at the price of worse average performance. A promising research direction is to combine the advantage of both while limiting their side-effects. In our work, we interleave the probabilistic slots, where the operation of the algorithm is randomized, with the deterministic ones to tradeoff the worst-case performance with the average performance. Another very recent result on the rendezvous problem demonstrates how to beat the quadratic theoretical barrier on the worst-case rendezvous delay by utilizing a public source of randomness in conjunction with a Markovian hitter [54]. Generally speaking, how to systematically orchestrate and synergize randomness with determinism is an important research axe for us.

From a networking and system perspective, our work and most neighbor discovery protocols in the literature do not make specific assumptions about the mobility patterns to achieve neighbour discovery. In other words, they can achieve discovery without exploiting knowledge concerning mobility patterns. This is sometimes an advantage as they can be applied to either static or mobile devices and are robust to mobility as their performance does not depend on mobility. However, there are many practical scenarios where by exploiting mobility patterns of devices, neighbor discovery can be facilitated. Hence, how to design mobility-aware neighbor discovery protocols that exploit device mobility to facilitate neighbor discovery, either limiting discovery delay or increasing energy efficiency, is another pertinent research direction. For example, a possible idea is to adapt a more flexible duty cycle mode and to allocate more active slots when mobility is important to ensure more agile neighbor discovery.

Chapter 3

Opportunistic Channel Access: A Restless Multi-Armed Bandit Perspective

3.1 Introduction

In this chapter, we address the problem of opportunistic channel access in a multi-channel communication system. Specifically, we consider a communication system in which a sender has access to multiple channels, but is limited to sense and transmit only on one or a limited number of channels in each slot. We explore how a smart sender should exploit past observations and the knowledge of the stochastic properties of these channels to maximize its transmission rate by switching opportunistically across channels.

Formally, we provide a generic analysis on the opportunistic spectrum access problem by casting the problem into the restless multi-armed bandit (RMAB) problem, one of the most well-known generalizations of the classic multi-armed bandit (MAB) problem, which is of fundamental importance in stochastic decision theory. Despite the significant research efforts in the field, the RMAB problem in its generic form still remains open. Until today, very little result is reported on the structure of the optimum policy¹. Obtaining the optimum policy for a general RMAB problem is often intractable due to the exponential computation complexity. Hence, a natural alternative is to seek a simple myopic policy maximizing the short-term reward.

3.1.1 Optimality of Myopic Sensing Policy

Motivated by the above analysis, we study the following natural while fundamentally important question: *under what conditions is the myopic policy guaranteed to be optimum?* We answer the question by performing an axiomatic study. More specifically, we develop three axioms characterizing a family of functions which we refer to as regular functions, which are generic and practically important. We then establish the optimality of the myopic policy when the reward function can be expressed as a regular function and when the discount factor is bounded by a closed-form threshold determined by the reward function. We also illustrate how the derived results, generic in nature, are applied to analyze a class of RMAB problems arising from multi-channel opportunistic access.

Compared with the existing literature addressing the optimality of the myopic policy of the

¹A policy is in fact an algorithm that tells the user which arm (channel in our case) to activate. We use the term policy to be coherent to the notation convention in the MAB context.

RMAB problem, our contributions can be summarized as follows:

- *Generic analysis:* In contrast to the research line followed by the related works in [6, 116, 125, 148, 237] aiming at showing the optimality/non-optimality of the myopic policy in given application scenarios, our work makes a more generic effort by focusing on the conditions ensuring the optimality without assuming any specific system setting.
- *Sensing error:* The vast majority of studies in the area assume perfect observation of channel states. However, sensing or observation errors are inevitable in practical scenario (e.g., due to noise and system limitations), especially in wireless communication systems which is the focus of our work. Our work captures the sensing error and studies the optimality of the myopic policy under imperfect sensing.

From the methodological perspective, we adopt an axiomatic approach to streamline the analysis in this part. On one hand, such axiomatic approach provides a hierarchical view of the addressed problem and leads to clearer and more synthetic analysis. On the other hand, the axiomatic approach also helps reduce the complexity of solving the RMAB problem and illustrates some important engineering implications behind the myopic policy.

3.1.2 Beyond Myopic Sensing: a Heuristic ν -step Lookahead Policy

In the first part of this chapter, we study the optimality of the myopic sensing policy in the case where the user is allowed to sense k out of N channels. In the second part, we further investigate a more challenging problem where the user has to decide the number of channels to sense in order to maximize its utility. This optimization problem hinges on the following tradeoff between exploitation and exploration: sensing more channels can help learn and predict the future channel state, thus maximizing the long-term reward, but at the price of sacrificing the reward at current slot as sensing more channels reduces the time for data transmission, thus decreasing the throughput in the current slot. Therefore, to find the optimum number of channels to sense consists of striking a balance between the above exploitation and exploration. After showing the NP hardness of the problem, we develop a heuristic ν -step lookahead policy which consists of sensing channels in a myopic way and stopping sensing when the expected aggregated utility from the current slot t to slot $t + \nu$ begins to decrease. In the developed policy, the parameter ν allows to achieve a desired tradeoff between social efficiency and computation complexity.

From the system perspective, our analysis provides insight on the following design tradeoff in opportunistic spectrum access:

- *Gaining immediate access versus gaining information for future use:* Due to hardware limitations and the energy cost of spectrum monitoring, a user may not be able to sense all the channels in the spectrum simultaneously. A sensing strategy is thus needed for intelligent channel selection to track the channel evolution. The purpose of a sensing strategy is twofold: to find good channels for immediate access and to gain statistical information on the channels for future transmissions. The optimum sensing strategy should thus strike a balance between these two often conflicting objectives.
- *Aggressive versus conservative:* Based on the imperfect sensing outcomes given by the spectrum sensor, the user needs to decide whether to access. An aggressive access strategy may lead to excessive transmission failures while a conservative one may result in throughput degradation due to overlooked opportunities.

Despite the our focus on opportunistic communication, the problem formulation is applicable in many other engineering fields such as communication jamming, scheduling and object tracking. Hence the results presented in this thesis are generically applicable in a large range of domains beyond the scope of opportunistic channel access.

3.1.3 Chapter Organization

The rest of this chapter is structured as follows. Section 3.2 provides an overview of the RMAB problem and its application. Section 3.3 develops our work on the myopic policy for the RMAB problem in the context of opportunistic channel access. Section 3.4 presents our work on the ν -step look-ahead strategy. Section 3.5 concludes the chapter by briefly summarizing our other related work related to this topic. The work of this chapter is in collaboration with my former Ph.D. student Kehao Wang (co-advised with Pr. Kaldoun Al Agha) who is now a visiting research associate at MIT. More details of our work on this topic including proofs and numerical analysis can be found in our publications [42, 200–203, 205–207].

3.2 RMAB and its Application

Before presenting our work, we start by providing a literature review on the main theory developed for the classic MAB problem and its extension to the RMAB problem. We then briefly review recent works on the application the RMAB problem in various fields including wireless communication and networking.

3.2.1 MAB and Gittins Index

Multi-armed bandit, first posed in 1933, has become a classical problem in stochastic optimization with a wide range of engineering applications, including but not limited to, multi-agent systems, web search and Internet advertising, social networks, and queueing systems. Recently, it has found new applications in communication networks and dynamic systems.

Consider a dynamic system consisting of a player and N independent arms. In each slot t ($t = 1, 2, \dots$), the state of arm k is denoted by $s_k(t)$ and completely observable to the player. At slot t , the player selects one arm, i.e., arm k , to activate based on the system state $\mathcal{S}(t) = [s_1(t), s_2(t), \dots, s_N(t)]$ and accrues reward $R(s_k(t))$ determined by the state $s_k(t)$ of arm k . Meanwhile, the state of arm k will transmit to another state in the next slot according to certain transition probabilities, i.e., $p_{i,j}^k = P(s_k(t+1) = j | s_k(t) = i)$, $i, j \in \Omega_k$, where Ω_k denotes the state space of arm k . The states of other arms which are not activated will remain frozen, i.e., $s_n(t+1) = s_n(t) \forall n \neq k$.

The player's selection policy $\pi = \{\pi(1), \pi(2), \dots\}$ is a series of mapping from the system state $\mathcal{S}(t)$ to the action $a(t)$ indicating which arm is activated, i.e., $\pi(t) : \mathcal{S}(t) \rightarrow a(t)$. The objective of the player is to obtain the optimum policy π^* maximizing the expected total discounted reward, i.e.,

$$\pi^* = \arg \max_{\pi} \mathbb{E} \left[\lim_{T \rightarrow \infty} \sum_{t=1}^T \beta^{t-1} R(s_{a(t)}(t)) \right],$$

where the discount factor $0 \leq \beta < 1$.

Since the size of the system states grows exponentially with the number of arms, the above problem, called the classic MAB problem, has an exponential complexity for its general numerical solutions.

This sequential decision problem was firstly proposed by Thompson in 1933 [191], but the theoretical structure of the optimum solution for the classic MAB has not been obtained until Gittins's seminal work [83] in 1974. Gittins showed that an index policy is optimum, called Gittins index later, and thus reduced the complexity of the problem from exponential to linear in the number N of arms.

Theorem 3.1. *The optimum policy has an index form. Specially, for all $1 \leq k \leq N$, there exists an index function $G_k(\cdot)$ that maps the state $i \in \Omega_k$ of arm k to a real number. At each time, the optimum action is to activate the arm with the largest index.*

Gittins also gave a specific form of the index function $G_k(\cdot)$, referred as Gittins index, as given in the following definition.

Definition 3.1 (Gittins Index). *For any state $i \in \Omega_k$ of arm k ,*

$$G_k(i) = \limsup_{\sigma \geq 1} \frac{\mathbb{E} \left[\sum_{t=1}^{\sigma} \beta^{t-1} R(s_k(t)) | s_k(1) = i \right]}{\mathbb{E} \left[\sum_{t=1}^{\sigma} \beta^{t-1} | s_k(1) = i \right]},$$

where σ is a stopping time for activating the arm k .

Basically, Gittins index measures the maximum reward rate that can be achieved by focusing on activating one arm starting from its current state. Therefore, by Gittins index, the player can accrue reward as quickly as possible and thus maximize the total discounted reward.

3.2.2 RMAB and Whittle Index

Whittle [212] extended the MAB to a more general model where a set of K ($K > 1$) arms, denoted as $K(t)$, can be activated simultaneously and change their states in each slot and meanwhile the passive arms are also allowed to offer reward and change state, which makes it different from the classic MAB. If arm k is activated, then its state transits according to a transmitting rule P_{k1} and yields the immediate reward $g_{k1}(s_k(t))$ while it transits by another rule P_{k2} and yields the immediate reward $g_{k2}(s_k(t))$ when arm k isn't activated. A policy $\pi = \{\pi(t)\}_{t=1}^{\infty}$ is a series of mappings where $\pi(t)$ maps the system state $\mathcal{S}(t)$ to the set of K arms $K(t)$ to be activated in slot t .

In [212], Whittle considered the above problem of maximizing the average reward over an infinite horizon², which can be formulated as follows:

$$\pi^* = \operatorname{argmax}_{\pi} \mathbb{E} \left\{ \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \underbrace{\left[\sum_{i \in K(t)} g_{i1}(s_i(t)) + \sum_{j=1, j \notin K(t)}^N g_{j2}(s_j(t)) \right]}_{R(t)} \right\}.$$

We now introduce Whittle's index by referring to the problem of opportunistic channel access³. To this end, we consider a Markovian single-armed bandit process, i.e., a single channel. In each

²The case of discounted reward can be treated similarly.

³We refer readers to [212] for a detailed presentation of Whittle's index and the related indexability result.

slot, the user chooses one of two possible actions $a \in \{0 \text{ (passive)}, 1 \text{ (active)}\}$ to make the arm passive or active. An expected reward of ω is obtained when the arm is activated at belief state ω ⁴. The objective is to decide whether to activate the arm in each slot to maximize the total discounted or average reward. The optimum policy is essentially given by an optimum partition of the state space $[0, 1]$ into a passive set $\{\omega : a^*(\omega) = 0\}$ and an active set $\{\omega : a^*(\omega) = 1\}$, where $a^*(\omega)$ denotes the optimum action under belief state ω .

Whittle's index measures how attractive it is to activate an arm based on the concept of subsidy for passivity. Specifically, we construct a single-armed bandit process that is identical to the above specified bandit process except that a constant subsidy m is obtained whenever the arm is made passive. Obviously, this subsidy m will change the optimum partition of the passive and active sets, and states that remain in the active set under a larger subsidy m are more attractive to the user. The minimum subsidy m that is needed to move a state from the active set to the passive set under the optimum partition thus measures how attractive this state is.

Whittle index and Whittle index policy are formally defined as below. Whittle also conjectured the optimality of the Whittle index policy.

Definition 3.2 (Whittle index and Whittle index policy [212]). *If all arms are indexable, its Whittle's index $W(\omega)$ of the state ω is the infimum subsidy m such that it is optimum to make the arm passive at ω . The Whittle index policy consists of activating the K arms of the largest indices in each slot.*

Conjecture 3.1 (Whittle Conjecture [212]). *If all arms are indexable, the Whittle index policy is optimum in terms of average reward per arm in the limit.*

3.2.3 Application of RMAB

The research efforts on analyzing the RMAB problem arising in various applications, especially communication and networking and dynamic systems, usually fall into the following three categories.

- The first one is to seek sufficient conditions for simple and robust policies (e.g., myopic policy, greedy policy) under which the optimality of such policies is guaranteed [6, 116, 125, 148, 237].
- The second one is to construct particular policies whose performance loss to the system optimum is bounded [84, 85].
- The third one is to calculate the Whittle index and derive policies based on the Whittle index [68, 101, 105, 118, 126, 162].

3.2.4 Non-Bayesian MAB

Another extension of MAB, different from the Bayesian formulation in our problem, is the so-called non-Bayesian MAB where the channels' availability statistics are not correlated in time as Markov chains and are initially unknown to the users and need to be estimated via learning. This leads to a tradeoff between exploration, by activating new arms to obtain more statistical information, and exploitation, by activating optimum arms based on current knowledge. The central problem in this formulation is to optimize the asymptotic performance by minimizing the

⁴Please refer to Sec. 3.3.1 for a formal description of the system model.

regret of the developed policy, given that the system regret under a policy π is defined as the accumulative expected reward loss up to time T under the policy π compared to the genie-aided policy where the stochastic properties of arms are known. Readers can refer to the literatures [4, 12, 13, 18, 115, 122, 123, 125, 189] for details.

3.3 Optimality of Myopic Sensing Policy

In this section, we study a natural while fundamental question: under what conditions is the myopic policy guaranteed to be optimum? Specifically, we develop three axioms characterizing a family of functions which we refer to as regular functions, which are generic and practically important. We then establish the optimality of the myopic policy when the reward function can be expressed as a regular function and the discount factor is bounded by a closed-form threshold determined by the reward function. We also illustrate how the derived results, generic in nature, are applied to analyze a class of RMAB problems arising from multi-channel opportunistic access.

In our study, we take into consideration the imperfect channel state observation due to sensing error. Note that the vast majority of studies in the area assume perfect observation of channel states. However, sensing or observation errors are inevitable in practical scenario (e.g., due to noise and system limitations), especially in wireless communication systems which is the focus of our work. More specifically, a good (bad, respectively) channel may be sensed as bad (good) and accessing a bad channel leads to zero reward. In such context, it is crucial to study the structure and the optimality of the myopic sensing policy with imperfect observation. We would like to emphasize that the presence of sensing error brings two difficulties when studying the myopic sensing policy in this new context.

- The belief vector evolves as a non-linear mapping instead of a linear one under perfect sensing;
- The belief value of a channel depends not only on the channel evolution itself, but also on the observation outcome, meaning that the transition is not deterministic.

Our major results are articulated as below. Firstly, we establish the structure of the myopic sensing policy, also termed as myopic policy for brevity. Secondly, we derive closed-form conditions under which the myopic policy is optimum under imperfect sensing and generic utility functions.

3.3.1 Problem Formulation

3.3.1.1 System Model

We consider a multi-channel opportunistic communication system, in which a user is able to access a set \mathcal{N} of N independent channels, each characterized by a Markov chain of two states, *good* (1) and *bad* (0). The channel state transition matrix $\mathbf{P}^{(i)}$ for channel i ($i \in \mathcal{N}$) is given below

$$\mathbf{P}^{(i)} = \begin{bmatrix} p_{11}^{(i)} & 1 - p_{11}^{(i)} \\ p_{01}^{(i)} & 1 - p_{01}^{(i)} \end{bmatrix},$$

where

$$\begin{aligned} p_{01}^{(i)} &= \text{prob}(\text{channel } i \text{ is good in the current slot given being bad in the previous slot}), \\ p_{11}^{(i)} &= \text{prob}(\text{channel } i \text{ is good in the current slot given being good in the previous slot}). \end{aligned}$$

We assume that channels go through state transition at the beginning of each slot t . The system operates in a slotted fashion with slots indexed by t ($t = 1, 2, \dots, T$), where T is the time horizon of interest. Due to hardware constraints and energy cost, the user is allowed to sense only k ($1 \leq k \leq N$) of the N channels at each slot t . We assume that the user makes the channel selection decision at the beginning of each slot after the channel state transition. Once a channel is chosen, the user detects the channel state $S_i(t)$, which can be considered as a binary hypothesis test:

$$\mathcal{H}_0 : S_i(t) = 1 \text{ (good)}, \quad \mathcal{H}_1 : S_i(t) = 0 \text{ (bad)}.$$

The performance of channel i state detection is characterized by the probability of false alarm ϵ_i and the probability of miss detection δ_i :

$$\begin{aligned} \epsilon_i &\triangleq \Pr\{\text{decide } \mathcal{H}_1 \mid \mathcal{H}_0 \text{ is true}\}, \\ \delta_i &\triangleq \Pr\{\text{decide } \mathcal{H}_0 \mid \mathcal{H}_1 \text{ is true}\}. \end{aligned}$$

We denote the set of channels chosen by the user at slot t by $\mathcal{A}(t)$ where $\mathcal{A}(t) \in \mathcal{N}$ and $|\mathcal{A}(t)| = k$. Based on the imperfect sensing results $\{O_i(t) \in \{0, 1\} : i \in \mathcal{A}(t)\}$ in slot t , the user determines which channels to access for transmission.

3.3.1.2 Restless Multi-Armed Bandit Formulation

Obviously, by imperfectly sensing only k out of N channels, the user cannot observe the state information of the whole system. Hence, the user has to infer the channel states from its past decision and observation history so as to make its future decision. To this end, we define the *channel state belief vector* (hereinafter referred to as *belief vector* for brevity) $\Omega(t) \triangleq \{\omega_i(t), i \in \mathcal{N}\}$, where $0 \leq \omega_i(t) \leq 1$ is the conditional probability that channel i is in good state (i.e., $S_i(t) = 1$). Given the sensing action $\mathcal{A}(t)$ and the observations $\{O_i(t) \in \{0, 1\} : i \in \mathcal{A}(t)\}$, the belief vector in $t + 1$ slot can be updated recursively using Bayes Rule as shown in (3.1):

$$\omega_i(t+1) = \begin{cases} p_{11}^{(i)}, & i \in \mathcal{A}(t), O_i(t) = 1 \\ \mathcal{T}_i(\varphi_i(\omega_i(t))), & i \in \mathcal{A}(t), O_i(t) = 0 \\ \mathcal{T}_i(\omega_i(t)), & i \notin \mathcal{A}(t) \end{cases} \quad (3.1)$$

where,

$$\mathcal{T}_i(\omega_i(t)) \triangleq \omega_i(t)p_{11}^{(i)} + (1 - \omega_i(t))p_{01}^{(i)}, \quad (3.2)$$

$$\varphi_i(\omega_i(t)) \triangleq \frac{\epsilon_i \omega_i(t)}{1 - (1 - \epsilon_i)\omega_i(t)}. \quad (3.3)$$

Remark. We would like to emphasize that the sensing error introduces further complications in the system dynamics (i.e., $\varphi_i(\omega_i(t))$ is non-linear with $\omega_i(t)$) compared with the perfect sensing case. Therefore, those results (e.g., [6]) obtained without sensing error cannot be trivially extended to the scenario with sensing error.

A sensing policy π specifies a sequence of functions $\pi = [\pi_1, \pi_2, \dots, \pi_T]$ where π_t maps the belief vector $\Omega(t)$ to the action (i.e., the set of channels to sense) $\mathcal{A}(t)$ in each slot t : $\pi_t : \Omega(t) \rightarrow \mathcal{A}(t)$, $|\mathcal{A}(t)| = k$.

Given the imperfect sensing, we are interested in the user's optimization problem to find the optimum sensing policy π^* that maximizes the expected total discounted reward over a finite horizon:

$$\pi^* = \operatorname{argmax}_{\pi} \mathbb{E} \left[\sum_{t=1}^T \beta^{t-1} R(\pi_t(\Omega(t))) \middle| \Omega(1) \right] \quad (3.4)$$

where $R(\pi_t(\Omega(t)))$ is the reward collected in slot t under the sensing policy π_t with the initial belief vector $\Omega(1)$ ⁵, $0 \leq \beta \leq 1$ is the discount factor characterizing the feature that the future rewards are less valuable than the immediate reward. By treating the belief value of each channel as the state of each arm of a bandit, the user's optimization problem can be cast into a RMAB problem.

3.3.1.3 Myopic Sensing Policy

In order to get more insight on the structure of the optimization problem formulated in (3.4) and the complexity to solve it, we derive the dynamic programming formulation of (3.4) as follows:

$$V_T(\Omega(T)) = \max_{\mathcal{A}(T)} \mathbb{E} [R(\pi_T(\Omega(T)))],$$

$$V_t(\Omega(t)) = \max_{\mathcal{A}(t)} \mathbb{E} \left[R(\pi_t(\Omega(t))) + \beta \sum_{\mathcal{E} \subseteq \mathcal{A}(t)} \prod_{i \in \mathcal{E}} (1 - \epsilon_i) \omega_i(t) \prod_{j \in \mathcal{A}(t) \setminus \mathcal{E}} [1 - (1 - \epsilon_j) \omega_j(t)] V_{t+1}(\Omega(t+1)) \right].$$

In the above Bellman equations, $V_t(\Omega(t))$ is the value function corresponding to the maximum expected reward from slot t to T ($1 \leq t \leq T$) with the believe vector $\Omega(t+1)$ following the evolution described in (3.1) given that the channels in the subset \mathcal{E} are sensed in good state and the channels in $\mathcal{A}(t) \setminus \mathcal{E}$ are sensed in bad state.

Solving (3.4) using the above recursive iteration is obviously computationally heavy. Hence, a natural alternative is to seek simple myopic sensing policy which is easy to compute and implement that maximizes the immediate reward, formally defined as follows.

Definition 3.3 (Myopic Policy). *Let the expected reward function $F(\Omega(t)) \triangleq \mathbb{E}[R(\pi_t(\Omega(t)))]$ denote the expected immediate reward obtained in slot t under the sensing policy π . The myopic sensing policy consists of sensing the k channels that maximizes $F(\Omega(t))$ for each slot t .*

Despite (or due to) its simple and robust structure, the optimality of the myopic sensing policy is not guaranteed. More specifically, when the channels are stochastically identical (i.e., all channels follow the same Markovian dynamics $\mathbf{P}^{(i)} = \mathbf{P}, \forall i \in \mathcal{N}$) and positively correlated (i.e., $p_{11} > p_{01}$), the myopic sensing policy is shown to be optimum when the user is limited to sensing one channel each slot ($k = 1$) and obtains one unit of reward when the sensed channel is good [237]. The analysis [6] and our work [209] further extend the study on the generic case where $k \geq 1$. However, the authors of [6] show that the myopic sensing policy is optimum if the user gets one unit of

⁵If no information on the initial system state is available, each entry of $\Omega(1)$ can be set to the stationary distribution $\omega_0^{(i)} = \frac{p_{01}^{(i)}}{1 + p_{01}^{(i)} - p_{11}^{(i)}}$, $1 \leq i \leq N$.

reward for each channel sensed to be good⁶, while our work [209] shows via counter-examples that the myopic sensing policy is not guaranteed to be optimal when the user's objective is to find at least one good channel⁷. Given that such nuance on the reward function leads to totally contrary results, a natural while fundamentally important question arises: how does the expected slot reward function $F(\Omega(t))$ impact the optimality of the myopic sensing policy? Or more specifically, under what conditions on $F(\Omega(t))$ is the myopic sensing policy guaranteed to be optimum?

In the sequel analysis by performing an axiomatic study, we give affirmative answer to the above posed questions and study some important engineering implications behind the myopic sensing policy. To make the presentation more streamlined, we present our results for the homogeneous case where $\mathbf{P}^{(i)} = \mathbf{P}$ and $\epsilon_i = \epsilon \forall i \in \mathcal{N}$. We thus drop the channel index for notation simplicity. Readers are referred to [201] for the heterogeneous case. We also assume that the channels are positively correlated ($p_{01} < p_{11}$), which corresponds to the realistic scenarios where the channel states are observed to evolve gradually over time.

To conclude this subsection, we state some structural properties of $\mathcal{T}(\omega_i(t))$ and $\varphi(\omega_i(t))$ that are useful in the subsequent analysis.

Lemma 3.1. *For positively correlated channel i , it holds that*

- $\mathcal{T}(\omega_i(t))$ is monotonically increasing in $\omega_i(t)$;
- $p_{01} \leq \mathcal{T}(\omega_i(t)) \leq p_{11}, \forall 0 \leq \omega_i(t) \leq 1$.

Lemma 3.2. *If $0 \leq \epsilon \leq \frac{(1-p_{11})p_{01}}{p_{11}(1-p_{01})}$ and $p_{01} < p_{11}$, it holds that*

- $\varphi(\omega_i(t))$ increases monotonically in $\omega_i(t)$ with $\varphi(0) = 0$ and $\varphi(1) = 1$;
- $\varphi(\omega_i(t)) \leq p_{01}, \forall p_{01} \leq \omega_i(t) \leq p_{11}$.

3.3.2 Axioms

This subsection introduces a set of three axioms characterizing a family of generic and practically important functions, to which we refer as *regular* functions. For presentation convenience, we sort the elements of the believe vector $\Omega(t) = [\omega_1(t), \dots, \omega_N(t)]$ for each slot t such that $\mathcal{A} = \{1, \dots, k\}$ (i.e., the user senses channel 1 to channel k) and let $\Omega_{\mathcal{A}} \triangleq \{\omega_i : i \in \mathcal{A}\} = \{\omega_1, \dots, \omega_k\}$ ⁸. The three axioms derived in the following characterize a generic function f defined on $\Omega_{\mathcal{A}}$.

Axiom 3.1 (Symmetry). *A function $f(\Omega_{\mathcal{A}}) : [0, 1]^k \rightarrow \mathbb{R}$ is symmetrical if $\forall i, j \in \mathcal{A}$ it holds that*

$$f(\omega_1, \dots, \omega_i, \dots, \omega_j, \dots, \omega_k) = f(\omega_1, \dots, \omega_j, \dots, \omega_i, \dots, \omega_k).$$

Axiom 3.2 (Monotonicity). *A function $f(\Omega_{\mathcal{A}}) : [0, 1]^k \rightarrow \mathbb{R}$ is monotonically increasing if it is monotonically increasing in each variable ω_i , i.e., $\forall i \in \mathcal{A}$*

$$\omega'_i > \omega_i \implies f(\omega_1, \dots, \omega'_i, \dots, \omega_k) > f(\omega_1, \dots, \omega_i, \dots, \omega_k).$$

Axiom 3.3 (Decomposability). *A function $f(\Omega_{\mathcal{A}}) : [0, 1]^k \rightarrow \mathbb{R}$ is decomposable if $\forall i \in \mathcal{A}$ it holds that*

$$f(\omega_1, \dots, \omega_i, \dots, \omega_k) = \omega_i f(\omega_1, \dots, 1, \dots, \omega_k) + (1 - \omega_i) f(\omega_1, \dots, 0, \dots, \omega_k).$$

⁶Formally, in [6], the expected slot reward function is defined as $F(\Omega(t)) = \sum_{i \in \mathcal{A}(t)} \omega_i(t)$

⁷In our work [209], the expected slot reward function is defined as $F(\Omega(t)) = 1 - \prod_{i \in \mathcal{A}(t)} (1 - \omega_i(t))$

⁸For presentation simplicity, by slightly abusing notations without introducing ambiguity, we drop the slot index t .

Axioms 3.1 and 3.2 are intuitive. Axiom 3.3 on the decomposability states that $f(\Omega_A)$ can always be decomposed into two terms that replace ω_i by 0 and 1, respectively. The three axioms are consistent and non-redundant. Moreover, they can be used to characterize a family of generic functions, referred to as *regular* functions, defined as follows.

Definition 3.4 (Regular Function). *A function is called regular if it satisfies all the three axioms.*

The developed three axioms characterize a set of generic functions widely used in practical applications. We give two examples to get more insight: (1) The user gets one unit of reward for each channel that is sensed good and is indeed good. In this example, the expected reward function (for each slot), denoted as F , is the expected slot reward function is $F(\Omega_A) = \sum_{i=1}^k [(1 - \epsilon)\omega_i]$; (2) The user gets one unit of reward if at least one channel is sensed good. In this example, the expected reward function is $F(\Omega_A) = 1 - \prod_{i=1}^k [1 - (1 - \epsilon)\omega_i]$. It can be verified that in both examples, F is regular by satisfying the three axioms.

The following definition studies the structure of the myopic sensing policy if the expected reward function is regular.

Definition 3.5 (Structure of Myopic Sensing Policy). *Sort the elements of the belief vector in descending order such that $\omega_1 \geq \dots \geq \omega_N$, if the expected reward function F is regular, then the myopic sensing policy, where the user can sense k channels, consists of sensing channel 1 to channel k .*

3.3.3 Optimality of Myopic Sensing Policy

We now establish closed-form conditions under which the myopic sensing policy achieves the system optimum under imperfect sensing. To this end, we set up by defining an auxiliary function and studying the structural properties of the auxiliary function, which serve as a basis in the study of the optimality of the myopic sensing policy. We then establish the main result on the optimality followed by illustrating how the obtained result can be applied via two concrete application examples.

For the convenience of discussion, we firstly state some notations before presenting the analysis:

- The believe vector $\Omega(t)$ is sorted to $[\omega_1(t), \dots, \omega_N(t)]$ at each slot t such that $\mathcal{A} = \{1, 2, \dots, k\}$;
- $\mathcal{N}(m) \triangleq \{1, \dots, m\}$ ($m \leq N$) denotes the first m channels in \mathcal{N} ;
- Given $\mathcal{E} \subseteq \mathcal{M} \subseteq \mathcal{N}$, $Pr(\mathcal{M}, \mathcal{E}) \triangleq \prod_{i \in \mathcal{E}} (1 - \epsilon)\omega_i(t) \prod_{j \in \mathcal{M} \setminus \mathcal{E}} [1 - (1 - \epsilon)\omega_j(t)]$, herein, $Pr(\mathcal{M}, \mathcal{E})$ denotes the expected probability that the channels in \mathcal{E} are sensed in good state, while the channels in $\mathcal{M} \setminus \mathcal{E}$ are sensed in bad state, given that the channels in \mathcal{M} are sensed;
- $\mathbf{P}_{11}^{\mathcal{E}}$ denotes the vector of length $|\mathcal{E}|$ with each element being p_{11} ;
- $\Phi(l, m) \triangleq [\mathcal{T}(\omega_i(t)), l \leq i \leq m]$ where the components are sorted by channel index. $\Phi(l, m)$ characterizes the updated belief values of the channels between l and m if they are not sensed;
- Given $\mathcal{E} \subseteq \mathcal{M} \subseteq \mathcal{N}$, $\mathbf{Q}^{\mathcal{M}, \mathcal{E}} \triangleq [\mathcal{T}(\varphi(\omega_i(t))), i \in \mathcal{M} \setminus \mathcal{E}]$ where the components are sorted by channel index. $\mathbf{Q}^{\mathcal{M}, \mathcal{E}}$ characterizes the updated belief values of the channels in $\mathcal{M} \setminus \mathcal{E}$ if they are sensed in the bad state.

- Let $\omega_{-i} \triangleq \{\omega_j : j \in \mathcal{A}, j \neq i\}$ and

$$\begin{cases} \Delta_{max} \triangleq \max_{i \in \mathcal{N}, \omega_{-i} \in [0,1]^{k-1}} \{F(1, \omega_{-i}) - F(0, \omega_{-i})\}, \\ \Delta_{min} \triangleq \min_{i \in \mathcal{N}, \omega_{-i} \in [0,1]^{k-1}} \{F(1, \omega_{-i}) - F(0, \omega_{-i})\}. \end{cases}$$

3.3.3.1 Definition and Properties of Auxiliary Value Function

Observing the form of the value function $V_t(\Omega(t))$, we first define the auxiliary value function with imperfect sensing and then derive several fundamental properties of the auxiliary value function, which are crucial in the study on the optimality of the myopic sensing policy.

Definition 3.6 (Auxiliary Value Function under Imperfect Sensing). *The auxiliary value function, denoted as $W_t(\Omega)$ ($t = 1, 2, \dots, T$) is recursively defined as follows:*

$$\begin{cases} W_T(\Omega(T)) = F(\omega_1(T), \dots, \omega_k(T)); \\ W_t(\Omega(t)) = F(\omega_1(t), \dots, \omega_k(t)) + \beta \sum_{\mathcal{E} \subseteq \mathcal{N}(k)} Pr(\mathcal{N}(k), \mathcal{E}) W_{t+1}(\Omega_{\mathcal{E}}(t+1)), \end{cases} \quad (3.5)$$

where $\Omega_{\mathcal{E}}(t+1) \triangleq (\mathbf{P}_{11}^{\mathcal{E}}, \Phi(k+1, N), \mathbf{Q}^{\mathcal{N}(k), \mathcal{E}})$ denotes the belief vector generated by $\Omega(t)$ based on (3.1).

The above recursively defined auxiliary value function gives the expected cumulated reward of the following sensing policy: in slot t , sense the first k channels; if channel i is correctly sensed good, then put it on the top of the list to be sensed in next slot, otherwise drop it to the bottom of the list. Recall Lemma 3.1 and Lemma 3.2, under the condition $0 \leq \epsilon \leq \frac{(1-p_{11})p_{01}}{p_{11}(1-p_{01})}$, if the belief vector $\Omega(t)$ is ordered decreasingly in slot t , the above sensing policy is the myopic sensing policy with $W_t(\Omega(t))$ being the total reward from slot t to T .

We prove some structural properties of the auxiliary value function.

Lemma 3.3 (Symmetry). *If the expected reward function F is regular, the correspondent auxiliary value function $W_t(\Omega)$ is symmetrical in any two channels $i, j \leq k$ for all $t = 1, 2, \dots, T$, i.e.,*

$$W_t(\omega_1, \dots, \omega_i, \dots, \omega_j, \dots, \omega_N) = W_t(\omega_1, \dots, \omega_j, \dots, \omega_i, \dots, \omega_N), \quad \forall i, j \leq k.$$

Lemma 3.4 (Decomposability). *If the expected reward function F is regular, then the correspondent auxiliary value function $W_t(\Omega(t))$ is decomposable for all $t = 1, 2, \dots, T$, i.e.,*

$$W_t(\omega_1, \dots, \omega_i, \dots, \omega_N) = \omega_i W_t(\omega_1, \dots, 1, \dots, \omega_N) + (1 - \omega_i) W_t(\omega_1, \dots, 0, \dots, \omega_N), \quad \forall i \in \mathcal{N}.$$

Lemma 3.4 can be applied one step further to prove the following corollary.

Corollary 3.1. *If the expected reward function F is regular, then for any $l, m \in \mathcal{N}$ it holds that for $t = 1, 2, \dots, T$*

$$\begin{aligned} W_t(\omega_1, \dots, \omega_l, \dots, \omega_m, \dots, \omega_N) - W_t(\omega_1, \dots, \omega_m, \dots, \omega_l, \dots, \omega_N) = \\ (\omega_l - \omega_m) \left[W_t(\omega_1, \dots, 1, \dots, 0, \dots, \omega_N) - W_t(\omega_1, \dots, 0, \dots, 1, \dots, \omega_N) \right]. \end{aligned}$$

Lemma 3.5 (Monotonicity). *If the expected reward function F is regular, the correspondent auxiliary value function $W_t(\Omega)$ is monotonously non-decreasing in $\omega_l, \forall l \in \mathcal{N}$, i.e.,*

$$\omega'_l \geq \omega_l \implies W_t(\omega_1, \dots, \omega'_l, \dots, \omega_N) \geq W_t(\omega_1, \dots, \omega_l, \dots, \omega_N).$$

3.3.3.2 Optimality of Myopic Sensing under Imperfect Sensing

To study the optimality of the myopic sensing policy, we first develop two auxiliary lemmas (Lemma 3.6 and Lemma 3.7) and then establish the sufficient condition under which the optimality of the myopic sensing policy is guaranteed.

Lemma 3.6. *Given that (1) $\epsilon < \frac{p_{01}(1-p_{11})}{P_{11}(1-p_{01})}$, (2) $\beta \leq \frac{\Delta_{min}/\Delta_{max}}{(1-\epsilon)(1-p_{01}) + \frac{\epsilon(p_{11}-p_{01})}{1-(1-\epsilon)(p_{11}-p_{01})}}$, and (3) F is regular, if $p_{11} \geq \omega_l \geq \omega_m \geq p_{01}$ where $l < m$, then it holds that*

$$W_t(\omega_1, \dots, \omega_l, \dots, \omega_m, \dots, \omega_N) \geq W_t(\omega_1, \dots, \omega_m, \dots, \omega_l, \dots, \omega_N), \quad t = 1, \dots, T.$$

Lemma 3.7. *Given that (1) $\epsilon < \frac{p_{01}(1-p_{11})}{P_{11}(1-p_{01})}$, (2) $\beta \leq \frac{\Delta_{min}/\Delta_{max}}{(1-\epsilon)(1-p_{01}) + \frac{\epsilon(p_{11}-p_{01})}{1-(1-\epsilon)(p_{11}-p_{01})}}$, and (3) F is regular, if $p_{11} \geq \omega_1 \geq \dots \geq \omega_N \geq p_{01}$, for any $1 \leq t \leq T$, it holds that*

$$W_t(\omega_1, \dots, \omega_{N-1}, \omega_N) - W_t(\omega_N, \omega_1, \dots, \omega_{N-1}) \leq (1 - \omega_N) \Delta_{max},$$

$$W_t(\omega_1, \omega_2, \dots, \omega_{N-1}, \omega_N) - W_t(\omega_N, \omega_2, \dots, \omega_{N-1}, \omega_1) \leq (p_{11} - p_{01}) \Delta_{max} \frac{1 - [\beta(1 - \epsilon)(p_{11} - p_{01})]^{T-t+1}}{1 - \beta(1 - \epsilon)(p_{11} - p_{01})}.$$

The proof of the above lemmas consists of applying the structural properties of the auxiliary value function and backward induction, as detailed in [200, 206]. Lemma 3.6 states that by swapping two elements in Ω with the former larger than the latter, the user does not increase the total expected reward. Lemma 3.7, on the other hand, gives the upper bound on the difference of the total reward of the two swapping operations, swapping ω_N and ω_k ($k = N - 1, \dots, 1$) and swapping ω_1 and ω_N , respectively.

We now state the main result on the optimality of myopic sensing policy in the following theorem. The proof [200, 206] consists of recursively applying Lemma 3.6 and Lemma 3.7.

Theorem 3.2 (Optimality of myopic sensing policy). *If $p_{01} \leq \omega_i(1) \leq p_{11}$, $1 \leq i \leq N$, the myopic sensing policy is optimum if the following conditions hold: (1) $F(\Omega)$ is regular; (2) $\epsilon < \frac{p_{01}(1-p_{11})}{P_{11}(1-p_{01})}$; (3) $\beta \leq \frac{\Delta_{min}/\Delta_{max}}{(1-\epsilon)(1-p_{01}) + \frac{\epsilon(p_{11}-p_{01})}{1-(1-\epsilon)(p_{11}-p_{01})}}$.*

As noted in [127], when the initial belief ω_i is set to $\frac{p_{01}}{p_{01}+1-p_{11}}$ as the popular case in practical systems, it can be checked that $p_{01} \leq \omega_i(1) \leq p_{11}$ holds. Moreover, even the initial belief does not fall in $[p_{01}, p_{11}]$, all the the belief values are bounded in the interval from the second slot following Lemma 3.1. Hence our results can be extended by treating the first slot separately from the future slots.

3.3.4 Discussion

We illustrate the application of the result obtained above in two concrete scenarios and compare our work with the existing results.

Consider the channel access problem in which the user is limited to sensing k channels and gets one unit of reward if the sensed channel is in the good state, i.e., the utility function can be formulated as $F(\Omega_A) = (1 - \epsilon) \sum_{i \in \mathcal{A}} \omega_i$. Note that the optimality of the myopic sensing policy under this model is studied in [127] for a subset of scenarios where $k = 1$, $N = 2$. We now

study the generic case with $k, N \geq 2$. To that end, we apply Theorem 3.2. Notice in this example, we have $\Delta_{min} = \Delta_{max} = 1 - \epsilon$. We can then verify that when $\epsilon < \frac{p_{01}(1-p_{11})}{P_{11}(1-p_{10})}$, it holds that $\frac{\Delta_{min}}{\Delta_{max} \left[(1-\epsilon)(1-p_{01}) + \frac{\epsilon(p_{11}-p_{01})}{1-(1-\epsilon)(p_{11}-p_{01})} \right]} > 1$. Therefore, when the conditions 1 and 2 hold, the myopic sensing policy is optimum for $0 \leq \beta \leq 1$. This result in generic cases significantly generalizes the results [127] where the optimality of the myopic policy is proved for the case of two channels and only conjectured for general cases.

Next consider another scenario where the user can sense k channels but can only choose one of them to transmit its packets. Under this model, the user wants to maximize its expected throughput. More specifically, the slot utility function $F = F(\Omega_A) = 1 - \prod_{i \in \mathcal{A}} [1 - (1 - \epsilon)\omega_i]$, which is regular. In this context, we have $\Delta_{max} = (1 - \epsilon)^{k-1} p_{11}^{k-1}$ and $\Delta_{min} = (1 - \epsilon)^{k-1} p_{01}^{k-1}$. The third condition on for the myopic policy to be optimal becomes $\beta \leq \frac{p_{01}^{k-1}}{p_{11}^{k-1} \left[(1-\epsilon)(1-p_{01}) + \frac{\epsilon(p_{11}-p_{01})}{1-(1-\epsilon)(p_{11}-p_{01})} \right]}$. Particularly, when $\epsilon = 0$, $\beta \leq \frac{p_{01}^{k-1}}{p_{11}^{k-1}(1-p_{01})}$. It can be noted that even when there is no sensing error, the myopic policy is not ensured to be optimum, which confirms our findings in [128, 209] on perfect sensing scenarios.

3.4 Beyond Myopic Policy: ν -step Lookahead Policy

In the previous section, we have studied the optimality of the myopic sensing policy in the case where the user is allowed to sense k out of N channels. In this section, we further investigate the more generic problem where the user has to decide the number of channels to sense in order to maximize its utility. This optimization problem hinges on the following tradeoff between exploitation and exploration: sensing more channels can help learn and predict the future channel state, thus increasing the long-term reward, but at the price of sacrificing the reward at current slot as sensing more channels reduces the time for data transmission, thus decreasing the throughput in the current slot. Therefore, to find the optimum number of channels to sense consists of striking a balance between the above exploitation and exploration. After showing the hardness of the problem, we develop a heuristic ν -step lookahead policy which consists of sensing channels in a myopic way and stopping sensing when the expected aggregated utility from the current slot t to slot $t + \nu$ begins to decrease. In the developed policy, the parameter ν allows to achieve a desired tradeoff between efficiency and computation complexity.

3.4.1 Problem Formulation

3.4.1.1 System Model

We consider the same scenario as the previous section in which a user tries to access a multi-channel opportunistic communication system consisting of a set \mathcal{N} of N channels, each given by a two state Markov chain with transition probabilities $\{p_{ij}^{(k)}\}_{i,j=0,1}$ ($1 \leq k \leq N$). The system operates in a slotted fashion where the slots are indexed by t ($1 \leq t \leq T$), where T is the time horizon of interest. We assume that channels go through state transition at the beginning of a slot. The length of each slot is denoted as Δ , which is further divided into two parts: the sensing phase and the transmission phase. Let $\delta = a\Delta$ ($a \leq 1$) denote the time needed to sense one channel, the sensing phase lasts $na\Delta$ if the user senses n channels and the transmission phase consists of the rest of the

time $(1 - an)\Delta$.

The user's objective is to maximize its throughput by choosing the appropriate set of channels to sense. Let $\mathcal{A}(t)$, $\mathcal{O}_A(t)$ denote the set of channels sensed and the set of sensing results $\mathcal{O}_A(t) = \{O_i(t) \in \{1, 0\}, i \in \mathcal{A}(t)\}$ by the user at slot t who can sense at most M ($1 \leq M < N$ and $aM \leq 1$) channels due to hardware limit and sensing constraint. If at least one of the sensed channel is in the good state, the user can successfully transmit one packet.⁹ In our study, we also take into consideration the imperfect sensing which is characterized by the missed detection (the channel is sensed good but is in fact bad) rate denoted as ζ and the false alarm rate denoted as ϵ (the channel is sensed bad but is in fact good).

Obviously, by imperfectly sensing only $|\mathcal{A}(t)|$ out of N channels at each slot t , the user cannot observe the state information of the whole system. Hence, the user has to infer the channel states from its past decision and observation history so as to make its future decision. Moreover, the current sensing outcome further serves as statistics for future decision. To this end, as in the previous section we define the *channel state belief vector* (hereinafter referred to as *belief vector* for brevity) $\Omega(t) \triangleq \{\omega_i(t), i \in \mathcal{N}\}$, where $0 \leq \omega_i(t) \leq 1$ is the conditional probability that channel i is in good state. Given the sensing set $\mathcal{A}(t)$ and the detection outcomes $\{O_i(t) \in \{0, 1\} : i \in \mathcal{A}(t)\}$, the belief vector in $t + 1$ slot can be updated recursively using Bayes Rule as shown in (3.6):

$$\omega_i(t+1) = \begin{cases} p_{11}^{(i)}, & i \in \mathcal{A}(t), O_i(t) = 1 \\ \mathcal{T}_i(\varphi(\omega_i(t))), & i \in \mathcal{A}(t), O_i(t) = 0 \\ \mathcal{T}_i(\omega_i(t)), & i \notin \mathcal{A}(t), \end{cases} \quad (3.6)$$

where,

$$\mathcal{T}_i(\omega_i(t)) \triangleq \omega_i(t)p_{11}^{(i)} + (1 - \omega_i(t))p_{01}^{(i)}, \quad (3.7)$$

$$\varphi(\omega_i(t)) \triangleq \frac{\epsilon\omega_i(t)}{1 - (1 - \epsilon)\omega_i(t)}. \quad (3.8)$$

3.4.1.2 Optimum Sensing Problem Formulation: Exploitation versus Exploration

We are interested in the user's optimization problem to find a channel sensing policy π^* that maximize the expected total discounted reward over a finite horizon. Mathematically, a sensing policy π_t is defined as a mapping from the belief vector $\Omega(t)$ to $\mathcal{A}(t)$ in slot t :

$$\pi_t : \Omega(t) \rightarrow \mathcal{A}(t), 1 \leq |\mathcal{A}(t)| \leq M, t = 1, 2, \dots, T.$$

The formal definition of the optimum sensing problem \mathbf{P} is given as follows:

$$\mathbf{P} : \pi^* = \operatorname{argmax}_{\pi_t} \mathbb{E} \left[\sum_{t=1}^T \beta^{t-1} R(\pi_t(\Omega(t)), \mathcal{O}_A(t)) \middle| \Omega(1) \right], \quad (3.9)$$

where the slot reward function $R(\pi_t(\Omega(t)), \mathcal{O}_A(t)) = R(\mathcal{A}(t), \mathcal{O}_A(t))$ is the user throughput in slot t under the sensing policy π_t with the initial belief vector $\Omega(1)$, $0 \leq \beta \leq 1$ is the discount.

⁹Our work can be extended to the case where the user is equipped with more than one radio and can access multiple channels at a time.

Mathematically, \mathbf{P} can be cast into a class of the RMAB problems with unknown number of arms to be activated. It is worth noting that the RMAB problem is proved to be PSPACE-hard. Hence, a natural alternative to tackle \mathbf{P} is to seek myopic sensing policy that maximizes the immediate reward. The motivation of focusing on the myopic sensing policy is two-fold:

- As demonstrated in our previous work, under certain conditions, the myopic sensing policy is ensured to be optimum.
- The myopic sensing policy has a simple and robust structure that makes it easy to implement in practice.

Existing studies on the myopic policy in the RMAB problem implicitly assume that the number of arms to activate (in the context of our work, the number of channels to sense) is fixed. A natural while crucial research problem is how many channels to sense at each slot so as to maximize the expected total reward, which is the focus of our work presented in this chapter. We sort $\Omega(t)$ for each slot t in the descending order such that $\omega_1(t) \geq \omega_2(t) \geq \dots \geq \omega_N(t)$ and thus form a channel list $l^0(t) = (1, 2, \dots, N)$ ¹⁰, the optimization problem on the number of channels to sense at each slot is formalized as follows:

$$\mathbf{P}_1 : n_t^* = \operatorname{argmax}_{|\mathcal{A}(t)|} \mathbb{E} \left[\sum_{t=1}^T \beta^{t-1} R(\mathcal{A}(t), \mathcal{O}_A(t)) \middle| \Omega(1) \right], \quad (3.10)$$

where, in slot t , the first $|\mathcal{A}(t)|$ channels are sensed, i.e., $\mathcal{A}(t) = \{1, \dots, |\mathcal{A}(t)|\}$.

It is insightful to note that the optimization problem \mathbf{P}_1 on the number of channels to sense hinges on the following tradeoff between exploitation and exploration: sensing more channels can help learn and predict the future channel state, thus increasing the long-term reward, but at the price of sacrificing the reward at current slot as sensing more channels reduces the time for data transmission, thus decreasing the throughput in the current slot.

Despite our focus on the opportunistic access problem of multi-channels communication system, the model formulation and the consequent analysis to solve the optimization problem can be generalized in the context of the RMAB problem and are readily applied in a variety of engineering fields such as object tracking, communication jamming and opportunistic packet scheduling. Therefore, the following description and the use of terms in the context of opportunistic spectrum access should be understood generically. Moreover, the slot reward function $R(\mathcal{A}(t), \mathcal{O}_A(t))$ that we adopt can be generically expressed in the normalized form as follows:

$$R(\mathcal{A}(t), \mathcal{O}_A(t)) = \begin{cases} 1 - C(|\mathcal{A}(t)|), & \text{if } \prod_{i \in \mathcal{A}(t)} (1 - O_i(t)) = 0 \\ 0, & \text{otherwise.} \end{cases} \quad (3.11)$$

where $C(|\mathcal{A}(t)|)$ is the cost function monotonously increasing in $|\mathcal{A}(t)|$, representing the time associated to channel sensing and frequency switching. The first line of the right hand side of (3.11) indicates that by sensing the channels in $\mathcal{A}(t)$ that contains at least one channel sensed good, the user obtains a payoff $1 - C(|\mathcal{A}(t)|)$. The second line indicates the case where none of the channels in $\mathcal{A}(t)$ is sensed good, the user obtains 0 as payoff. In the channel access model depicted in Sec. 3.4.1.1, by normalizing $\Delta = 1$, we have $C(|\mathcal{A}(t)|) = a|\mathcal{A}(t)|$.

¹⁰The initial order of list is determined by the initial availability probability of each channel: $\omega_1(1) \geq \omega_2(1) \geq \dots \geq \omega_N(1) \Rightarrow l^0(1) = (1, 2, \dots, N)$.

3.4.2 When Stop Sensing: the ν -step Lookahead Policy

It can be noticed that given a policy $\{n(t), 1 \leq t \leq T\}$ (i.e., the number of channels to sense at each slot, given the myopic sensing order), the belief vectors $\{\Omega(t), 1 \leq t \leq T\}$ form a Markov process with an uncountable state space, which makes the optimization problem \mathbf{P}_1 intractable. Therefore, we turn to the following heuristic policy referred to as ν -step lookahead policy: at each slot t , the user senses the channels in the decreasing order of $\Omega(t)$ and estimates the expected accumulated payoff from slot $t + 1$ to slot $t + \nu$ ($t + \nu \leq T$), assuming that in slots $t + 1, \dots, t + \nu$, the user stops exploring new channels once an available one is found (or the maximum number of channels to be sensed, M , is reached); the user stops sensing new channels when the sum of the reward in the current slot plus that from slot $t + 1$ to $t + \nu$ decreases.

We now give the mathematical description of the ν -step lookahead policy. Let $l^k(t)$ and $\Omega^k(t)$ ($k \leq M$) denote the channel list and belief vector formed in the descending order of $\omega_i(t)$ ($1 \leq i \leq N$) after sensing the first k best channels in slot t , and $l_j^i(t)$ denote the j th channel in $l^i(t)$.

To streamline our presentation, we introduce the pseudo cost function defined as follows:

$$q(\mathcal{A}(t), \mathcal{O}_A(t)) \triangleq 1 - R(\mathcal{A}(t), \mathcal{O}_A(t)) = \begin{cases} C(|\mathcal{A}(t)|) = a|\mathcal{A}(t)|, & \text{if } \prod_{i \in \mathcal{A}(t)} (1 - O_i(t)) = 0 \\ C_0 = 1, & \text{otherwise.} \end{cases} \quad (3.12)$$

The optimization problem \mathbf{P}_1 can be written as the following optimization problem \mathbf{P}_2 :

$$\mathbf{P}_2 : n_t^* = \underset{|\mathcal{A}(t)|}{\operatorname{argmin}} \mathbb{E} \left[\sum_{t=1}^T \beta^{t-1} q(\mathcal{A}(t), \mathcal{O}_A(t)) \middle| \Omega(1) \right]. \quad (3.13)$$

Given the initial belief vector $\Omega^0(t + 1)$ at the beginning of slot $t + 1$ (with the correspondent channel list $l^0(t + 1)$), let $Q_{t+1}^{t+\nu}(\Omega^0(t + 1))$ denote the expected accumulative pseudo cost accrued from slot $t + 1$ to slot $t + \nu$, given that the user stops sensing once a channel is sensed good or M is reached, i.e.,

$$Q_{t+1}^{t+\nu}(\Omega^0(t + 1)) \triangleq \sum_{i=1}^M \underbrace{\left[[\omega_{l_i^0(t+1)}(t+1) \prod_{j=1}^{i-1} (1 - \omega_{l_j^0(t+1)}(t+1))] [C(i) + \beta \cdot Q_{t+2}^{t+\nu}(\mathbb{T}(\Omega_1^i(t+1)))] \right]}_A + \underbrace{\prod_{j=1}^M (1 - \omega_{l_j^0(t+1)}(t+1)) [C_0 + \beta \cdot Q_{t+2}^{t+\nu}(\mathbb{T}(\Omega_0^M(t+1)))]}_B,$$

where term A denotes the pseudo cost when channel $l_i^0(t + 1)$ is sensed good while channels $l_1^0(t + 1), \dots, l_{i-1}^0(t + 1)$ are sensed bad; term B denotes the pseudo cost when the first M channels of $l^0(t + 1)$ are sensed bad; $\Omega_1^i(t + 1)$ and $\Omega_0^i(t + 1)$ denote the belief vectors where the channel $l_i^0(t + 1)$ is sensed good and bad, respectively; \mathbb{T} denotes the mapping from $\Omega^k(t)$ to $\Omega^0(t + 1)$ according to (3.6) at the beginning of slot $t + 1$, i.e., $\mathbb{T} : \Omega^k(t) \rightarrow \Omega^0(t + 1)$; $Q_{t_1+1}^{t_2}(\mathbb{T}(\Omega^i(t_1)))$ denotes the expected accumulative cost from slot $t_1 + 1$ to t_2 when i channels are sensed at slot t_1 .

At each slot t , the ν -step lookahead policy can be implemented in a heuristic approach by transforming the problem into an optimum stopping problem, i.e., the user stops sensing new

channels when the sum of the reward in the current slot plus that from slot $t + 1$ to $t + \nu$ decreases. Mathematically, the number of channels to sense in the ν -step lookahead policy, denoted as $\bar{n}(t)$, is as follows:

$$\bar{n}(t) = \inf \left\{ |\mathcal{A}(t)| : C(\mathcal{A}(t), \mathcal{O}_{\mathcal{A}(t)}(t)) + \beta Q_{t+1}^{t+\nu}(\mathbb{T}(\Omega^{|\mathcal{A}(t)|}(t))) \right. \\ \left. < C(\mathcal{A}'(t), \mathcal{O}_{\mathcal{A}'(t)}(t)) + \beta \bar{Q}_{t+1}^{t+\nu}(\Omega^{|\mathcal{A}(t)|}(t)), 1 \leq |\mathcal{A}(t)| \leq M \right\}, \quad (3.14)$$

where $\mathcal{A}'(t) = \mathcal{A}(t) \cup \{l_{|\mathcal{A}(t)|+1}^0(t)\}$ denotes the best $|\mathcal{A}(t)| + 1$ channels in $l^0(t)$, $Q_{t+1}^{t+\nu}(\mathbb{T}(\Omega^{|\mathcal{A}(t)|}(t)))$ is the expected accumulative pseudo cost from slot $t + 1$ to $t + \nu$ when the best $|\mathcal{A}(t)|$ channels of $l^0(t)$ are sensed, and $\bar{Q}_{t+1}^{t+\nu}(\Omega^{|\mathcal{A}(t)|}(t))$ denotes the expected accumulative pseudo cost from slot $t + 1$ to $t + \nu$ when the $(|\mathcal{A}(t)| + 1)$ th channel of $l^0(t)$ is sensed good with probability $(1 - \epsilon)\omega_{|\mathcal{A}(t)|+1}^0(t)$ and bad with probability $1 - (1 - \epsilon)\omega_{|\mathcal{A}(t)|+1}^0(t)$, i.e.,

$$\bar{Q}_{t+1}^{t+\nu}(\Omega^{|\mathcal{A}(t)|}(t)) \triangleq (1 - \epsilon)\omega_{|\mathcal{A}(t)|+1}^0(t)Q_{t+1}^{t+\nu}(\mathbb{T}(\Omega_1^{|\mathcal{A}(t)|+1}(t))) \\ + (1 - (1 - \epsilon)\omega_{|\mathcal{A}(t)|+1}^0(t))Q_{t+1}^{t+\nu}(\mathbb{T}(\Omega_0^{|\mathcal{A}(t)|+1}(t))). \quad (3.15)$$

The following theorem further studies the structure of the ν -step lookahead policy by developing an optimum stopping algorithm to implement it.

Theorem 3.3. *The ν -step lookahead policy can be implemented by Algorithm 3. In each iteration,*

- *the user continues to sense new channels if all the sensed channels are bad (exploration);*
- *if at least one channel is sensed good, the user stops sensing new channels if the expected pseudo cost increases by sensing a new channel (exploration).*

Algorithm 3 ν -step lookahead policy: executed for each slot t

Input: $\Omega^0(t), l^0(t)$

Output: $\mathcal{A}(t)$

Initialization: $\mathcal{A}(t) = \emptyset$

while $|\mathcal{A}(t)| < M$ **do**

Sense the $(|\mathcal{A}(t)| + 1)$ th channel in $l^0(t)$

Add the sensed channel in $\mathcal{A}(t)$, i.e., $\mathcal{A}(t) \leftarrow \mathcal{A}(t) \cup l_{|\mathcal{A}(t)|+1}^0(t)$

if At least one channel in $\mathcal{A}(t)$ is sensed good and the following inequality holds:

$$C(|\mathcal{A}(t)|) + \beta Q_{t+1}^{t+\nu}(\mathbb{T}(\Omega^{|\mathcal{A}(t)|}(t))) < C(|\mathcal{A}(t)| + 1) + \beta \bar{Q}_{t+1}^{t+\nu}(\Omega^{|\mathcal{A}(t)|}(t)) \quad (3.16)$$

then

Terminate the algorithm by outputting $\mathcal{A}(t)$

end if

end while

Remark. *The ν -step lookahead policy can be decomposed into two steps:*

- *Exploitation: the user exploits the current available information $\Omega(t)$ in a greedy way so as to find a good channel;*

- *Exploration: once a good channel secured, the user proceeds to explore the system state space for long term gain.*

The second step (exploration) may be absent if all the M best channels are sensed bad or if exploring does not increase gain in the long term (i.e., the condition in Algorithm 3 does not hold even once).

The complexity of the algorithm implementing the ν -step lookahead policy lies in the computation of (3.16), whose complexity is exponential with ν . On the other hand, a larger ν leads to better performance of the lookahead policy. Hence, the user can tune the parameter ν to achieve a desired tradeoff between complexity and efficiency.

3.4.3 One-step Lookahead Policy

Having derived the algorithm implementing the proposed ν -step lookahead policy, we now focus on the case of i.i.d. channels and provide an mathematical analysis on the case where $\nu = 1$, i.e., the one-step lookahead policy. Our motivation of investigating this particular policy is two-fold:

- the study on the one-step lookahead policy can provide structural insights on the computation of the expected pseudo cost, which is the foundation of the ν -step lookahead policy. The general case $\nu > 1$ can be extended iteratively from the case $\nu = 1$;
- through numerical experiments (please refer to our publication [207]), we observe that the benefit of the ν -step lookahead policy is most important in the case of $\nu = 1$ and then decreases gradually with the increase of ν ; this observation, combined with the fact that the complexity of the ν -step lookahead policy increases exponentially with ν , motivates a more focused analysis on the one-step lookahead policy, which seems to be the most practical strategy in many scenarios.

Given the system model presented in Subsection 3.4.1.1, assume that the user has sensed k channels with at least one of them is in state good, recalling Algorithm 3, the condition to decide whether to sense channel $k + 1$ in the channel list can be written as:

$$a > \beta [Q_{t+1}^{t+\nu}(\mathbb{T}(\Omega^k(t))) - \bar{Q}_{t+1}^{t+\nu}(\Omega^k(t))]. \quad (3.17)$$

We next show how to compute $Q_{t+1}^{t+1}(\mathbb{T}(\Omega^k(t)))$ and $\bar{Q}_{t+1}^{t+1}(\Omega^k(t))$ in an efficient way. We present the major steps and orient readers to our publication [207] for detailed analysis and algebraic demonstrations. We first show the following lemma on how the channel list should be updated when a new channel is sensed.

Lemma 3.8. *For a system with positively correlated homogeneous i.i.d. channels, if $0 \leq \epsilon \leq \frac{(1-p_{11})p_{01}}{p_{11}(1-p_{01})}$, the channel sensed good (bad) should be moved to the head (tail) of the old channel list to form the new channel list.*

Assume that the channel list at the beginning of slot t before sensing any channels is $l^0(t) = (1, 2, \dots, N)$, sorted in the decreasing order of the belief values. Assume that among the k sensed channels $\{1, \dots, k\}$, m ($m \geq 1$) channels are sensed good while $k - m$ are bad. It follows from Lemma 3.8 that m channels are moved to the head of the channel list and others to the tail, thus forming the new channel list $l^k(t)$. We now show how to compute $Q_{t+1}^{t+1}(\mathbb{T}(\Omega^k(t)))$, $Q_{t+1}^{t+1}(\mathbb{T}(\Omega_1^{k+1}(t)))$ and $Q_{t+1}^{t+1}(\mathbb{T}(\Omega_0^{k+1}(t)))$ in the case of $m \geq 1$ so as to decide whether to sense channel $k + 1$.

We first define an auxiliary vector $\mathbf{X}(\mathbb{T}(\Omega^k(t)), m)$:

$$\mathbf{X}(\mathbb{T}(\Omega^k(t)), m) \triangleq \begin{pmatrix} 1 \\ X_1(\mathbb{T}(\Omega^k(t)), m) \\ X_2(\mathbb{T}(\Omega^k(t)), m) \\ X_3(\mathbb{T}(\Omega^k(t)), m+2) \\ X_4(\mathbb{T}(\Omega^k(t)), m+2) \end{pmatrix} \triangleq \begin{pmatrix} 1 \\ \prod_{j=1}^m (1 - \omega_{l_j^k(t)}(t+1)) \\ 1 + \sum_{i=1}^m \prod_{j=1}^i (1 - \omega_{l_j^k(t)}(t+1)) \\ \prod_{j=m+2}^M (1 - \omega_{l_j^k(t)}(t+1)) \\ \sum_{i=m+2}^M \prod_{j=m+2}^i (1 - \omega_{l_j^k(t)}(t+1)) \end{pmatrix}.$$

The following lemma establishes a structural property of $\mathbf{X}(\mathbb{T}(\Omega^k(t)), m)$ by showing that $\mathbf{X}(\mathbb{T}(\Omega^{k+1}(t)), m+1)$ can be recursively derived based on $\mathbf{X}(\mathbb{T}(\Omega^k(t)), m)$ in both cases where the channel $k+1$ is sensed good and bad, respectively.

Lemma 3.9. *The following recursive update on the auxiliary vector holds:*

- If $k+1$ channel is sensed good, $\mathbf{X}(\mathbb{T}(\Omega_1^{k+1}(t)), m+1) = \mathbf{H}_1 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m)$,
- If $k+1$ channel is sensed bad, $\mathbf{X}(\mathbb{T}(\Omega_0^{k+1}(t)), m+1) = \mathbf{H}_2 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m)$,

where

$$\mathbf{H}_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 - (1 - \epsilon)p_{11} & 0 & 0 & 0 \\ 1 & 0 & 1 - (1 - \epsilon)p_{11} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{1 - \omega_{l_{m+2}^k(t)}(t+1)} & 0 \\ -1 & 0 & 0 & 0 & \frac{1}{1 - \omega_{l_{m+2}^k(t)}(t+1)} \end{pmatrix},$$

$$\mathbf{H}_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{1 - \omega_{l_{M+1}^k(t)}(t+1)}{1 - \omega_{l_{m+2}^k(t)}(t+1)} & 0 \\ -1 & 0 & 0 & \frac{1 - \omega_{l_{M+1}^k(t)}(t+1)}{1 - \omega_{l_{m+2}^k(t)}(t+1)} & \frac{1}{1 - \omega_{l_{m+2}^k(t)}(t+1)} \end{pmatrix}.$$

Theorem 3.4 further shows that $Q_{t+1}^{t+1}(\mathbb{T}(\Omega^k(t)))$, $Q_{t+1}^{t+1}(\mathbb{T}(\Omega_1^{k+1}(t)))$ and $Q_{t+1}^{t+1}(\mathbb{T}(\Omega_0^{k+1}(t)))$ can be easily computed by using the auxiliary vector. Consequently, the one-step lookahead policy can be implemented in an efficient fashion by using the auxiliary vector, which can be updated recursively.

Theorem 3.4. *It holds that*

$$Q_{t+1}^{t+1}(\mathbb{T}(\Omega^k(t))) = a \left[\mathbf{A}_1 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) + \mathbf{A}_2 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) \cdot \mathbf{A}_3 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) \right] \quad (3.18)$$

$$Q_{t+1}^{t+1}(\mathbb{T}(\Omega_1^{k+1}(t))) = a \left[\mathbf{A}_4 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) + \mathbf{A}_5 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) \cdot \mathbf{A}_6 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) \right] \quad (3.19)$$

$$Q_{t+1}^{t+1}(\mathbb{T}(\Omega_0^{k+1}(t))) = a \left[\mathbf{A}_1 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) + \mathbf{A}_7 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) \cdot \mathbf{A}_8 \cdot \mathbf{X}(\mathbb{T}(\Omega^k(t)), m) \right], \quad (3.20)$$

where,

$$\begin{aligned}
 \mathbf{A}_1 &= [0, 0, 1, 0, 0], & \mathbf{A}_2 &= [0, 1 - \omega_{l_{m+1}^k(t)}(t+1), 0, 0, 0] \\
 \mathbf{A}_3 &= [1, 0, 0, \frac{1}{a} - M - 1, 1], & \mathbf{A}_4 &= [1, 0, 1 - (1 - \epsilon)p_{11}, 0, 0] \\
 \mathbf{A}_5 &= [0, 1 - (1 - \epsilon)p_{11}, 0, 0, 0], & \mathbf{A}_6 &= [0, 0, 0, \frac{1}{a} - M - 1, 1] \\
 \mathbf{A}_7 &= [0, 1, 0, 0, 0], & \mathbf{A}_8 &= [0, 0, 0, (\frac{1}{a} - M)(1 - (1 - \epsilon)\mathcal{T}(\varphi(\omega_{l_{m+1}^k(t)}(t))))], 0].
 \end{aligned}$$

Recall Algorithm 3 and (3.18)–(3.20), it can be verified that the one-step lookahead policy has a linear computational complexity $O(M)$.

We further demonstrate some of the theoretical results derived above and gain more insight on the developed ν -step lookahead policy as well as the performance tradeoff via a set of numerical experiments. Readers are referred to [207] for detailed numerical analysis and demonstration.

3.5 Summary of Other Contributions

In this section, we present a brief summary on a selection of our other work on RMAB-based channel access, which can be regarded as extensions and variants of the work presented in the previous two sections.

3.5.1 Myopic Policy under Multi-state Markov Channel Model

Our first extension is to consider the Markov channel model with m states where $m = 2$ has been extensively investigated. Specifically, we consider a communication system composed of N independent channels each of which is modeled as a time-nonhomogeneous m -state Markov chain with known matrix of transition probabilities. At each time period a user selects a number of channels to access and uses it to transmit information. A reward depending on the states of those selected channels is obtained for each transmission. The objective is to design a channel access policy that maximizes the expected accumulated discounted reward (respectively, the expected accumulated reward) collected over a finite (respectively, infinite) time horizon.

Our work along this research axis is the construction a set of conditions to guarantee the optimality of myopic policy. In particular, we show that the structure of the myopic policy is a simple queue determined by the availability probability vector of channels provided that certain condition is satisfied for the transition matrix of multi-state channels. Further, we obtain a set of conditions under which the myopic policy is proved to be optimum. Our derivation demonstrates the advantage of branch-and-bound and the directed comparison based optimization approach. These results consist of a generic complement to the state of the art of the RMAB theory, although the structure of the optimum policy of generic case is still unsolved.

3.5.2 Myopic Channel Probing Policy in Underlay Cognitive Radio Systems

Our second extension is to consider an underlay cognitive radio communication system in which a cognitive user can access multiple primary spectrum channels only when its interference to the primary users is limited. To identify and exploit instantaneous transmission opportunities, the cognitive user probes a subset of primary channels by overhearing the primary feedback signals

so as to learn the primary receiver's channel condition and the interference tolerance level, then chooses appropriate power to transmit its data. In such context, the user cannot probe all the channels for its limited number of receiving antennas, then a crucial optimization problem faced by the user is to probe which channel(s) in order to maximize the long-term throughput given the past probing history.

We tackle this problem by casting it into a RMAB problem. Given the specific and practical constraints posed by the problem, we analyze the myopic probing policy which consists of probing the best channels based on the past observation. We perform an analytical study on the optimality of the developed myopic probing policy. Technically, we divide the belief vector in this work into the value belief vector and the policy belief vector, which reflects the essence of decomposability. The optimality condition derived in this work degenerates to those obtained in the literature by relaxing the corresponding constraints.

3.5.3 RMAB with Multiples Users and Channel Switching Cost

In this work, we study a channel access scenario with multiple users where switching among channels incurs an additional cost. We cast the problem into a non-Bayesian MAB problem. Our objective is to develop a distributed channel access algorithm with logarithmic regret. That is, the performance degradation compared to the system optimum is a logarithmic function in time. The major difficulty in our problem is that the channel switching cost add a new element in the regret.

In order to design asymptotically efficient channel access policies with logarithmic regret, we need to limit the frequency of channel switching at users. In this line of design, we develop the block-based channel access policy. The proposed channel access policy is inspired by the block allocation scheme in [5] on the single-player MAB problem with switching cost and adapted in our multi-user context. The main idea can be summarized as follows: we group slots in blocks; at the beginning of each block, the users choose which channel to sense and stick to that channel for the whole block if no collision is experienced during the whole block; otherwise in case of collision, indicating more than one user on the same channel, a channel randomization is performed such that each user experiencing the collision switches randomly to another channel. The block structure is carefully constructed such that the total cost of channel switching and the loss due to collisions are both controlled to $O(\log t)$, resulting a global $O(\log t)$ regret.

3.6 Conclusion and Perspectives

This chapter addresses the problem of opportunistic channel access. We have provided a generic analysis by casting the problem into the RMAB problem and conducted a systematic analysis on a class of myopic policies of both theoretical and practical importance. The key insight hinging behind our analysis is the fundamental tradeoff between obtaining immediate access (exploitation) and acquiring information for future use (exploration).

In this chapter, we mainly focus on the decision making process for a single user. A natural research direction is to take the obtained results as a building block to further study the scenario of multiple users accessing opportunistically a multi-channel communication system. Here the key research challenge is how to coordinate the users to access different channels in a distributed fashion without or with little explicit network-level feedback. A natural way to tackle this problem

is to model the situation as a non-cooperative game among users and to see how the results obtained in this chapter can further be tailored in the new context.

Another practical extension is to consider the correlated channels, i.e., the Markov chains of different channels can be correlated. This problem can be cast into the RMAB problem (the Markovian formulation) with correlated arms. The introduction of the correlation among arms makes the tradeoff between exploration and exploitation more sophisticated as sensing a channel can not only reveal the state of the sensed channel, but also provide information on other channels as they are not entirely independent. How to characterize the tradeoff in this new context and how to design efficient channel access policies are pertinent research topics in this direction.

Chapter 4

Distributed Learning in Wireless Networks: A Game-theoretical Perspective

4.1 Introduction

In this chapter, we investigate the following problem arising from emerging wireless networks: How to design distributed algorithms that allow users to gradually converge to a stable and desirable system state based on purely location information and interactions? We tackle this problem by using game theory (more precisely, non-cooperative game theory) as a systematic framework of modeling and analysis. The motivation of investigating the problem from the (non-cooperative) game theoretic perspective is three-fold.

- Game theory is a powerful tool to model the interactions of decision makers with mutually conflicting objectives, e.g., the interaction among rational and selfish nodes in wireless networks.
- Non-cooperative game theory can model the features or constraints of wireless networks such as lack of coordination and network feedback. In fact, in such environments, non-cooperative behavior is much more robust and scalable than any centralized cooperative control, which is very expensive or even impossible to implement.
- Game theory can serve as a validation tool to evaluate and benchmark the proposed algorithms.

Under the non-cooperative game-theoretical framework, we model the network as a site where each rational node adjust its strategy under the non-cooperative paradigm to maximize its own payoff. In such game theoretic studies, the central issue is to derive and characterize the resulting Nash equilibria (NE), where no one has incentive to deviate unilaterally. We then develop distributed learning algorithms for nodes to adjust their strategies to converge to a system equilibrium based on only observable local information and interactions.

For concreteness, we instantiate our study by focusing on the distributed channel access in cognitive radio networks and devise a suite of distributed channel access algorithms with guaranteed convergence to the system equilibrium. However, we note that the algorithms developed in our work are generically applicable in a wide range of networking and system problems such as load balancing, selfish routing and resource allocation. In this sense, the model description and the use of terms in this chapter (such as “channels”) should be understood generically.

4.1.1 Imitation-based Spectrum Access

We first consider a generic model of cognitive networks consisting of multiple frequency channels, each characterized by a channel availability probability determined by the activity of primary users (PUs) on it. In such model, from the secondary users' (SUs) perspective, a challenging problem is to compete (or coordinate) with other SUs in order to opportunistically access the unused spectrum of PUs to maximize its own payoff (e.g., throughput); at the system level, a crucial research issue is to design efficient spectrum access algorithm achieving optimum spectrum usage.

We formulate the spectrum access problem as a non-cooperative game and develop distributed spectrum access algorithms based on *imitation*, a behavior rule widely applied in human societies consisting of imitating successful behavior. We establish the convergence of the proposed policies to an imitation-stable equilibrium which is also the ϵ -optimum of the system. Simple, natural and incentive-compatible, the proposed spectrum access algorithms can be implemented distributedly based on solely local interactions and thus are especially suited in decentralized adaptive learning environments as cognitive radio networks.

In our analysis, we start by developing the imitation-based spectrum access policies where a SU can imitate any other SUs. More specifically, we develop two spectrum access policies based on the following two imitation rules:

- the Proportional Imitation (PI) rule where a SU can sample one other SU;
- the more advanced adjusted proportional imitation rule with double sampling (Double Imitation, DI) where a SU can sample two other SUs.

Under both imitation rules, each SU strives to improve its individual payoff by imitating other SUs with higher payoff. We then adapt the proposed spectrum access policies to a more practical scenario where a SU can only imitate the other SUs operating on the same channel. A systematic theoretical analysis is presented for both scenarios on the induced imitation dynamics and the convergence properties of the proposed policies to an imitation-stable equilibrium, which is also the ϵ -optimum of the system.

Distributed spectrum access has been widely addressed in the literature. As discussed in the last chapter, the problem can be cast into the RMAB problem, where spectrum access policies are devised where the number of SUs is known or estimated by each SU [12]. Another important thrust consists of applying game theory to model the competition and cooperation among SUs and the interaction between SUs and PUs. Particularly, a number of works (e.g. [55]) model the spectrum access as a potential game and study the system dynamics under asymptotic assumptions. More related to our work, several works focus on learning via imitation. Alos-Ferrer *et al.* study an imitation-based model of evolution with noise, where players have got memories of their past payoffs and adopt the Imitate If Best Rule (see [7] for a review on imitation rules) among the strategies associated to its recalled payoffs and those of another random player [8]. Ackermann *et al.* investigated the concurrent imitation dynamics in the context of finite population symmetric congestion games by focusing on the convergence properties [2]. Berenbrik *et al.* applied the Proportional Imitation Rule to load-balance system resources by focusing on the convergence speed [29]. Ganesh *et al.* applied the Imitate If Better rule in order to load-balance the service rate of parallel server systems [80].

Compared to the state of the art, the key contribution of our work lies in the systematical application of the natural imitation behavior to address the spectrum access problem with specific constraints (e.g., learning is restricted to nodes on the same channel), the design of a distributed

imitation-based channel access algorithm, and the theoretic analysis on the induced imitation dynamic and the convergence to an efficient and stable system equilibrium.

4.1.2 Retrospective Spectrum Access

We then consider the case where instead of imitating other nodes, a node imitates only its behavior that has brought him higher payoff in the past. Such “self-imitation” demonstrates more robustness in the case where imitating others is not possible or reliable.

Technically, we develop and analyze a framework of retrospective spectrum access protocols based on stochastic learning that can orient the network towards a socially efficient and fair equilibrium state. Our developed retrospective spectrum access protocol has two features: (1) the entirely distributed implementation requiring only local observations and (2) the guaranteed statistical convergence to the equilibrium state within a bounded delay.

The developed retrospective protocol follows a natural design philosophy: in each decision period, an SU j explores a new channel with probability ϵ and migrates to the channel that gives the best utility within last H_j periods with probability $(1 - \epsilon)(1 - \rho)$ (ρ is called endogenous inertia). We note that the protocol can model the particular feature that each SU is equipped with bounded memory and should make its decision based on only local observations. The protocol also models a natural human decision making behavior of striking a balance between exploring a new choice and retrospectively exploiting past successful choices. To analyze the performance of the developed protocol, we apply the mistake model introduced in [77, 143, 144, 225] and establish the statistical convergence of dynamics to the system equilibrium within a bounded latency $O(1/\epsilon)$.

While our model is presented in the specific context of channel access, we intend it more generically as a contribution to the literature on bounded rationality and learning in the presence of noise, which thus far have been mostly explored in biology and economics. Relying on the classical work [72, 77, 143, 144, 225], Dieckmann analyzed the evolution of conventions in a society with local interactions and mobile players [192]. Mertikopoulos *et al.* studied exponential learning in the presence of noise and the induced stochastic replicator dynamics [141]. Friedman and Mezzetti investigated mistakes models that induce better and best reply dynamics [79]. Young *et al.* proposed a series of completely uncoupled rules¹ (e.g., [78, 136, 137, 158, 226]) possessing several appealing properties. The version of Trial and Error presented in [158], for instance, is able to converge to the PNE maximizing the social welfare. Nevertheless, complexity is high and convergence speed is very slow, as it has been shown in [96].

The retrospective learning protocol that we propose has a similar architecture to the learning procedures developed in [136] and [240]. Our main contributions with respect to the existing literature is the introduction of nontrivial memories and inertia, as well as the results on convergence time. We would like to emphasize that despite our focus on channel access, the developed stochastic learning protocol and the analysis methodology in this work also provide some insights on the design of decentralized load balancing algorithms that respect locality constraints (e.g., bounded memory, bounded rationality) while converging to the balanced state within a bounded delay.

¹An individual’s learning rule is *completely uncoupled* if it does not depend directly on the actions or payoffs of anyone else.

4.1.3 Chapter Organization

The rest of this chapter is structured as follows. Section 4.2 develops our work on the imitation-based spectrum access. Section 4.3 presents our work on the retrospective spectrum access. Section 4.4 concludes the chapter by briefly summarizing our other related work related to this topic. Part of the work of this chapter is the topic of the thesis of my former Ph.D. student Stefano Iellamo (co-advised with Pr. Marceau Coupechoux) who is actually a Marie-Curie research fellow at ICS-FORTH. The ongoing thesis of Mira Morcos (co-advised with Pr. Tijani Chahed) is also related to this topic. More details of our work on this topic including proofs and numerical analysis can be found in our publications [23–26, 70, 71, 94–96].

4.2 Imitation-based Spectrum Access

4.2.1 System Model and Spectrum Access Game Formulation

We consider a primary network consisting of a set \mathcal{C} of C frequency channels, each with bandwidth B^2 . The users in the primary network are operated in a synchronous time-slotted fashion. A set \mathcal{N} of N SUs tries to opportunistically access the channels when they are left free by PUs. Let $Z_i(k)$ be the random variable equal to 1 when of channel i is unoccupied by any PU at slot k and 0 otherwise. We assume that the process $\{Z_i(k)\}$ is stationary and independent for each i and k . We also assume that at each time slot, channel i is free with probability μ_i , i.e., $\mathbb{E}[Z_i(k)] = \mu_i$. The channel availability probabilities $\mu \triangleq \{\mu_i\}$ are *a priori* not known by SUs. We assume perfect sensing at the SUs, i.e., any transmission of any PU on a channel is perfectly sensed by SUs sensing that channel and thus no collision occurs between PUs and SUs.

In our work, each SU j is modelled as a rational decision maker, striking to maximize the throughput it can achieve, denoted as T_j , which can be expressed as a function of μ_i and n_{s_j} , where s_j denotes the channel which j chooses, n_{s_j} denotes the number of SUs on channel s_j . More formally, the expected value of T_j can be written as:

$$\mathbb{E}[T_j] = f(\mu_i, n_{s_j}).$$

In order to perform a closed-form analysis, we focus on the scenario where the channel capacity is evenly shared among all SUs on the channel when it is free, i.e.,

$$\mathbb{E}[T_j] = f(\mu_{s_j}, n_{s_j}) = B\mu_{s_j}/n_{s_j}.$$

It should be noted that $f(\mu_{s_j}, n_{s_j})$ depends on the MAC protocol implemented at the cognitive users. Beside the evenly shared model considered here, several other models are also largely applied in practice such as the CSMA-based random access model. Our work can be adapted in those cases by defining appropriate function f .

To study the interactions among autonomous selfish SUs and to derive distributed channel access policies, we formulate the channel selection problem as a spectrum access game where the players are the SUs. Each player j stays on a channel i to opportunistically exploit the unused spectrum of PUs to maximize its expected throughput. The game is defined formally as follows:

²Our analysis can be extended to study the heterogeneous case with different channel capacities.

Definition 4.1. The spectrum access game G is a 3-tuple $(\mathcal{N}, \mathcal{C}, \{U_j\})$, where \mathcal{N} is the player set, \mathcal{C} is the strategy set of each player. Each player j chooses its strategy $s_j \in \mathcal{C}$ to maximize its normalized utility function U_j defined as

$$U_j = \mathbb{E}[T_j]/B = \mu_{s_j}/n_{s_j}.$$

The solution of the spectrum access game G is characterized by a Nash Equilibrium (NE) [149], a strategy profile from which no player has incentive to deviate unilaterally. Using the related theory on congestion games, we can establish the existence and the uniqueness of the NE in the spectrum access game G for the asymptotic case ($N \rightarrow \infty$) in the following theorem.

Theorem 4.1. In the asymptotic case, G admits a unique NE. At the NE, there are $x_i^* N$ SUs staying with channel i , where $x_i^* = \frac{\mu_i}{\sum_{l \in \mathcal{C}} \mu_l}$.

We can observe two desirable properties of the unique NE derived in Theorem 4.1:

- the NE is optimum from the system perspective as the total throughput of the network achieves its optimum at the NE;
- the NE ensures that the spectrum resource is shared fairly among SUs.

One critical challenge in the analyzed spectrum access game is the design of distributed spectrum access strategies for rational SUs to converge to the NE without the *a priori* knowledge of μ . In response to this challenge, we develop an efficient spectrum access policy. Our proposed policy can be implemented distributedly based on solely local interactions without any knowledge on the channel statistics and thus is especially suited in decentralized adaptive learning environments as cognitive radio networks. In terms of performance, we demonstrate both analytically and numerically that the proposed channel access policy converges to the ϵ -NE³ of G which is also the ϵ -optimum of the system.

4.2.2 Imitation-based Spectrum Access

The spectrum access algorithm we develop is based on *imitation*. As a behavior rule widely observed in human societies, imitation captures the behavior of a rational player that mimics the actions of other players with higher payoff in order to improve its own payoff. The induced imitation dynamic models the spreading of successful strategies under imitation [174]. In this section, we focus on the scenario where a SU can imitate any other SUs and develop two spectrum access policies based on the proportional imitation rule and the double imitation rule. We analyze the induced dynamic of the imitation process and show the convergence of the proposed policy to the ϵ -NE of G . In the next section, we extend our efforts to a more practical scenario where a SU can only imitate the other SUs operating on the same channel and develop an adapted imitation-based spectrum access policy in the new context.

4.2.2.1 Proportional Imitation

Algorithm 4 presents our proposed spectrum access policy based on the proportional imitation rule, termed as PISAP. The core idea is as follows: at each iteration, each SU randomly selects

³A strategy profile is an ϵ -NE if no player can gain more than ϵ in payoff by unilaterally deviating from his strategy.

another SU in the network; if the payoff of the selected SU is higher than its own payoff, the SU imitates the strategy of the selected SU at the next iteration with a probability proportional to the payoff difference, coefficiented by the imitation factor σ .⁴

Algorithm 4 PISAP: executed at each SU j

- 1: **Initialization:** set the imitation factor σ and the imitation threshold ϵ_U
 - 2: For the first iteration $t = 1$, randomly choose a channel to stay
 - 3: **while** at each iteration $t \geq 2$ **do**
 - 4: Randomly select a SU j'
 - 5: **if** $U_j < U_{j'} - \epsilon_U$ **then**
 - 6: Migrate to the channel $s_{j'}$ with probability $p = \sigma(U_{j'} - U_j)$
 - 7: **end if**
 - 8: **end while**
-

We first study the dynamic induced by PISAP by setting $\epsilon_U = 0$. It is shown in [170] that in the asymptotic case, the proportional imitation rule in Algorithm 4 generates a population dynamic described by the following set of differential equations:

$$\dot{x}_i(t) = \sigma x_i(t) [\pi_i(t) - \bar{\pi}(t)] \quad i \in \mathcal{C}, \quad (4.1)$$

where π_i denotes the expected payoff of the SUs on channel i , $\bar{\pi} \triangleq \sum_{i \in \mathcal{C}} x_i \pi_i$ denotes the expected payoff of all SUs in the network. Injecting $\pi_i = \mu_i / (x_i N)$ into the differential equations, (4.1) becomes:

$$\frac{\dot{x}_i(t)}{\sigma} = \frac{\mu_i}{N} - x_i(t) \sum_{l \in \mathcal{C}} \frac{\mu_l}{N}.$$

This equation can be easily solved as:

$$x_i(t) = K_i e^{-(\sum_{l \in \mathcal{C}} \frac{\mu_l}{N}) \sigma t} + \frac{\mu_i}{\sum_{l \in \mathcal{C}} \mu_l}, \quad (4.2)$$

where the constant $K_i = x_i(0) - \frac{\mu_i}{\sum_{l \in \mathcal{C}} \mu_l}$.

As the first result of this section, the following theorem states the convergence of the dynamic to the NE of the spectrum access game G .

Theorem 4.2. *The imitation dynamic induced by PISAP converges exponentially to the NE of G .*

We then study the convergence of PISAP in the general case with $\epsilon_U > 0$. Specifically, we define the *imitation-stable* equilibrium as a state where no further imitations can be conducted based on the imitation policy [2]. The following theorem analyzes the convergence of PISAP with respect to this concept.

Theorem 4.3. *PISAP converges to an imitation-stable equilibrium in expected $O(\frac{N^2}{\mu_{\min} \sigma \epsilon_U})$ iterations where $\mu_{\min} \triangleq \min_{i \in \mathcal{C}} \mu_i$. The converged equilibrium is an ϵ -NE of G with $\epsilon = 2\epsilon_U$.*

Note that the convergence delay $O(\frac{N^2}{\mu_{\min} \sigma \epsilon_U})$ derived in Theorem 4.3 consists of the upper bound and through the simulations we conduct, we observe that the convergence is achieved in a much shorter delay.

⁴One way of setting σ is to set $\sigma = 1/(\omega - \alpha)$, where ω and α are two exogenous parameters such that $U_j \in [\alpha, \omega], \forall j \in \mathcal{C}$.

4.2.2.2 Double Imitation

We next turn to a more advanced imitation rule, the double imitation (DI) rule [173], and propose the DI-based spectrum access policy, termed as DISAP. Under DISAP, each SU randomly samples two SUs and imitates them with a certain probability determined by the utility difference. The spectrum access policy based on the double imitation is detailed in Algorithm 5, in which each SUs randomly samples two other SUs j_1 and j_2 (without loss of generality, assume that j_1 and j_2 operate on channel i_1 and i_2 respectively, with corresponding utilities $U_{j_1} \leq U_{j_2}$) and updates the probabilities of switching to channels i_1 and i_2 , denoted as p_{j_1} and p_{j_2} respectively.

Algorithm 5 DISAP: executed at each SU j for each iteration

- 1: **Initialization:** set the two exogenous parameters ω and α such that the payoff of SUs falls into the interval $[\alpha, \omega]$, set the imitation factor σ and the imitation threshold ϵ_U
 - 2: Randomly sample two SUs j_1 and j_2
 - 3: Compute $p_{j_1} = \frac{\sigma}{2} [Q(U_j)(U_{j_1} - U_{j_2}) + Q(U_{j_2})(U_{j_1} - U_j)]^+$, where $[A]^+$ denotes $\max\{0, A\}$ and $Q(U_r) \triangleq \frac{1}{\omega - \alpha} \left(2 - \frac{U_r - \alpha}{\omega - \alpha}\right)$ for SU r
 - 4: **if** $Q(U_j)(U_{j_1} - U_{j_2}) < Q(U_{j_2})(U_j - U_{j_1})$ **then**
 - 5: $p_{j_2} = \frac{\sigma}{2} [Q(U_{j_1})(U_{j_2} - U_j) + Q(U_{j_2})(U_{j_1} - U_j)]^+$
 - 6: **else**
 - 7: $p_{j_2} = \frac{\sigma}{2} [Q(U_{j_1})(U_{j_2} - U_{j_1}) + Q(U_j)(U_{j_2} - U_{j_1})]^+$
 - 8: **end if**
 - 9: Switch to channel i_1 with probability p_{j_1} if $U_j < U_{j_1} - \epsilon_U$, switch to channel i_2 with probability p_{j_2} if $U_j < U_{j_2} - \epsilon_U$
-

The double imitation rule generates an aggregate monotone dynamic [169, 173], which is defined as follows:

$$\dot{x}_i = \frac{x_i}{\omega - \alpha} \left[1 + \frac{\omega - \bar{\pi}}{\omega - \alpha}\right] (\pi_i - \bar{\pi}) \quad \forall i \in \mathcal{C} \quad (4.3)$$

Injecting $\pi_i = \mu_i / (x_i N)$ into the differential equations, we have:

$$\dot{x}_i = \frac{\sigma \bar{\pi}}{\omega - \alpha} \left(1 + \frac{\omega - \bar{\pi}}{\omega - \alpha}\right) - \frac{\sigma \bar{\pi}}{\omega - \alpha} \left(1 + \frac{\omega - \bar{\pi}}{\omega - \alpha}\right) x_i,$$

whose solution is

$$x_i(t) = K e^{-\frac{\sigma \bar{\pi}}{\omega - \alpha} \left(1 + \frac{\omega - \bar{\pi}}{\omega - \alpha}\right) t} + \frac{\mu_i}{\sum_{l \in \mathcal{C}} \mu_l}, \quad (4.4)$$

where $\bar{\pi} = \sum_{l \in \mathcal{C}} \mu_l / N$ and $K = x_i(0) - \frac{\mu_i}{\sum_{l \in \mathcal{C}} \mu_l}$. In the studied scenario, α and ω are the lower and upper bound of the SUs' utility, which are 0 and 1, respectively.

The following theorem stating the major result in this subsection follows immediately.

Theorem 4.4. *DISAP converges exponentially to the NE of the spectrum access game G .*

Compared with the proportional imitation rule, which produces the replicator dynamic (Eq. (4.1)), the adjusted proportional imitation rule induces the aggregate monotone dynamic (Eq. (4.3)) that converges to the NE at a higher rate.

We then study the convergence to an imitation-stable equilibrium of DISAP in the general case with $\epsilon_U > 0$ in the following theorem.

Theorem 4.5. *DISAP converges to an imitation-stable equilibrium in expected $O(\frac{N^2}{\mu_{min}\sigma\epsilon_U})$ iterations where $\mu_{min} \triangleq \min_{i \in \mathcal{C}} \mu_i$. The converged equilibrium is an ϵ -NE of G with $\epsilon = 2\epsilon_U$.*

4.2.2.3 Discussion

As desirable properties, the proposed imitation-based spectrum access policies (both PISAP and DISAP) are stateless, incentive-compatible for selfish autonomous SUs and requires no central computational unit. The spectrum assignment is achieved by local interactions among autonomous SUs and the ϵ -optimum of the system is achieved when the algorithm converges, which is achieved in polynomial time. The autonomous behavior and decentralized implementation make the proposed policies especially suitable for large scale cognitive radio networks. The imitation factor σ controls the tradeoff between the convergence speed and the channel switching frequency in that larger σ represents more aggressiveness in imitation and thus leads to fast convergence, at the price of more frequent channel switching for the SUs which may consist of significant cost for today's wireless devices in terms of delay, packet loss and protocol overhead. The imitation threshold ϵ_U , on the other hand, can be tuned to balance between the convergence speed and the optimality of the converged equilibrium.

4.2.3 Imitation on the Same Channel

Up to now, we have studied the imitation-based channel access policy where a SU can imitate any other SU whatever the channel the latter stays in. This approach implicitly assumes that a SU can interact with SUs on different channels, which may not be realistic in some cases or pose additional system overhead (e.g., sensing a different channel). In this subsection, we focus on a more practical scenario, where a SU only imitates the SUs on the same channel and the imitation is based on the payoff difference of the precedent iteration. In the considered scenario, a SU only needs to locally interact with the SUs on the same channel (e.g., exchange payoff of the precedent iteration, which can be piggybacked with the data packets transmitted on the channel).

In the sequel analysis, we first study the induced imitation dynamic and the convergence of the proposed spectrum access policies PISAP and DISAP subject to channel constraint on imitation.

4.2.3.1 Imitation Dynamic and Convergence

We first derive in Theorem 4.6 the dynamic for a generic imitation rule F with large population. We then derive in Lemma 4.1, Theorem 4.7 and Theorem 4.8 the dynamic of the proposed proportional imitation policy PISAP and its convergence under the channel constraint. The counterpart analysis for the double imitation policy DISAP is explored in Lemma 4.2, Theorem 4.9 and Theorem 4.10.

We start by introducing the notations used in our analysis. At an iteration, we label all SUs performing strategy i (channel i in our case) as SUs of type i and we refer to the SUs on s_j as neighbors of SU j . We denote $n_i^l(t)$ the number of SUs on channel i at iteration t and operating on channel l at $t - 1$. It holds that $\sum_{l \in \mathcal{C}} n_i^l(t) = n_i(t)$ and $\sum_{i \in \mathcal{C}} n_i^l(t) = n_l(t - 1)$. For a given state $s(t) \triangleq \{s_j(t), j \in \mathcal{C}\}$ at iteration t and a finite population of size N , we denote $p_i(t) \triangleq n_i(t)/N$ the proportion of SUs of type i and $p_i^l(t) \triangleq n_i^l(t)/N$ the proportion of SUs migrating from channel l to

i. We use x instead of p to denote these proportions in asymptotic case. It holds that $p \rightarrow x$ when $N \rightarrow +\infty$.

In our study, a generic imitation rule under the channel constraint is termed as F . In the case of the proportional imitation rule (PISAP), F is characterized by the probability set $\{F_{j,k}^i\}$ where $F_{j,k}^i$ denotes the probability that a SU choosing strategy j at the precedent iteration imitates another SU choosing strategy k at the precedent iteration and then switches to channel i at next iteration after imitation. Instead, by applying the double imitation rule (DISAP), we can characterize F by the probability set $\{F_{j,\{k,l\}}^i\}$ where $F_{j,\{k,l\}}^i$ denotes the probability that a SU choosing strategy j at the precedent iteration imitates two neighbors choosing respectively strategy k and strategy l at the precedent iteration and then switches to channel i at next iteration after imitation. In both cases the only way to switch to a channel i is to imitate a SU that was on channel i . That means $F_{j,k}^i = 0, \forall k \neq i$ (PISAP) and $F_{j,\{k,l\}}^i = 0, \forall k, l \neq i$ (DISAP).

At the initialization phase (iteration 0 and 1), each SU randomly chooses its strategy. After that, the system state at iteration $t + 1$, denoted as $\mathbf{p}(t + 1)$ ($\mathbf{x}(t + 1)$ in the asymptotic case), depends on the states at iteration t and $t - 1$.

Theorem 4.6. *For any imitation rule F , if the imitation among SUs of the same type occurs randomly and independently, then $\forall \delta > 0, \epsilon > 0$ and any initial state $\{\tilde{x}_i(0)\}, \{\tilde{x}_i(1)\}$, there exists $N_0 \in \mathbb{N}$ such that if $N > N_0, \forall i \in \mathcal{C}$, the event $|p_i(t) - x_i(t)| > \delta$ occurs with probability less than ϵ , where $p_i(0) = x_i(0) = \tilde{x}_i(0), p_i(1) = x_i(1) = \tilde{x}_i(1)$. In the case of proportional imitation policy it holds that*

$$x_i(t + 1) = \sum_{j,l,k \in \mathcal{C}} \frac{x_j^l(t)x_j^k(t)}{x_j(t)} F_{l,k}^i \quad \forall i \in \mathcal{C}$$

Differently, the double imitation policy yields:

$$x_i(t + 1) = \sum_{j,l,k,z \in \mathcal{C}} \frac{x_j^l(t)x_j^k(t)x_j^z(t)}{[x_j(t)]^2} F_{l,\{k,z\}}^i \quad \forall i \in \mathcal{C}$$

The proof of Theorem 4.6 [96] consists of first showing the theorem holds for iteration $t = 2$ and then proving the case $t \geq 3$ by induction. Theorem 4.6 is an important result on the short run adjustments of large populations under any generic imitation rule F : the probability that the behavior of a large population differs from the one of an infinite population is arbitrarily small when N is sufficiently large. In what follows, we study the convergence of PISAP and DISAP under the channel constraint.

(1) Spectrum access policy PISAP under channel constraint

We now focus on PISAP under channel constraint and derive the induced imitation dynamic by setting $\epsilon_U = 0$ in the following analysis.

Lemma 4.1. *On the proportional imitation policy PISAP under channel constraint, it holds that*

$$x_i^j(t + 1) = \sum_{l,k \in \mathcal{C}} \frac{x_j^l(t)x_j^k(t)}{x_j(t)} F_{l,k}^i \quad \forall i, j \in \mathcal{C}. \quad (4.5)$$

Theorem 4.7. *The proportional imitation policy PISAP under channel constraint generates the following dynamic in the asymptotic case:*

$$x_i(t+1) = x_i(t-1) + \sigma \pi_i(t-1) x_i(t-1) - \sigma \sum_{j,l \in \mathcal{C}} \pi_l(t-1) \frac{x_j^i(t) x_j^l(t)}{x_j(t)}, \quad (4.6)$$

where $\pi_i(t)$ denotes the expected payoff of an individual SU on channel i at iteration t .

We observe via extensive numerical experiments that (4.6) always converges to the equilibrium. To get more in-depth insight on the dynamic (4.6), we notice that under the following approximation:

$$\sum_{l \in \mathcal{C}} \pi_l(t-1) \frac{x_j^l(t)}{x_j(t)} \approx \bar{\pi}(t-1), \quad (4.7)$$

where $\bar{\pi}(t-1)$ is the average individual payoff for the whole system at iteration $t-1$, noticing $\sum_j x_j^i(t) = x_i(t-1)$, (4.6) can be written as:

$$x_i(t+1) = x_i(t-1) + \sigma x_i(t-1) [\pi_i(t-1) - \bar{\pi}(t-1)]. \quad (4.8)$$

Note that the approximation (4.7) states that in any channel j at iteration t , the proportions of SUs coming from any channel l are representative of the whole population.

Under the approximation (4.7), given the initial state $\{x_i(0)\}$, $\{x_i(1)\}$, we can decompose (4.8) into the following two independent discrete-time replicator dynamics:

$$\begin{cases} x_i(u) = x_i(u-1) + \sigma x_i(u-1) [\pi_i(u-1) - \bar{\pi}(u-1)] \\ x_i(v) = x_i(v-1) + \sigma x_i(v-1) [\pi_i(v-1) - \bar{\pi}(v-1)] \end{cases} \quad (4.9)$$

where $u = 2t$, $v = 2t + 1$. The two equations in (4.9) illustrate the underlying system dynamic hinged behind the proportional imitation policy under channel constraint under the approximation (4.7): it can be decomposed into two independent delayed replicator dynamics that alternatively occur at the odd and even iterations, respectively. The following theorem establishes the convergence of (4.9) to a unique fixed point which is also the NE of the spectrum access game G .

Theorem 4.8. *Starting from any initial point, the system described by (4.9) converges to a unique fixed point which is also the NE of the spectrum access game G .*

Furthermore, performing the same analysis as that of Theorem 4.3, we can establish the same convergence property on the imitation algorithm under channel constraint under the approximation (4.7) for the general case with $\epsilon_U \geq 0$.

(2) Spectrum access policy DISAP under channel constraint

We then focus on DISAP under channel constraint and derive the induced imitation dynamic.

Lemma 4.2. *On the double imitation policy DISAP under channel constraint, it holds that*

$$x_i^j(t+1) = \sum_{l,k,z \in \mathcal{C}} \frac{x_j^l(t) x_j^k(t) x_j^z(t)}{[x_j(t)]^2} F_{l,\{k,z\}}^i \quad \forall i, j \in \mathcal{C}. \quad (4.10)$$

Theorem 4.9. *The double imitation policy DISAP under channel constraint generates the following dynamic in the asymptotic case*

$$\begin{aligned}
 x_i(t+1) = x_i(t-1) + 2x_i(t-1)\pi_i(t-1) + \sum_j x_j^i(t) \left[\sum_k \frac{x_j^k(t)}{x_j(t)} \right]^2 - 2x_j^i(t) \sum_k \frac{x_j^k(t)}{x_j(t)} \pi_k(t-1) \\
 - x_j^i(t)\pi_i(t-1) \sum_k \frac{x_j^k(t)}{x_j(t)} \pi_k(t-1) \quad (4.11)
 \end{aligned}$$

where $\pi_i(t)$ denotes the expected payoff of an individual SU on channel i at iteration t .

By performing the same approximation as in (4.7), (4.11) can be written as:

$$x_i(t+1) = x_i(t-1) + x_i(t-1)(2 - \bar{\pi}(t-1))(\pi_i(t-1) - \bar{\pi}(t-1)). \quad (4.12)$$

Under the approximation (4.7), given the initial state $\{x_i(0)\}, \{x_i(1)\}$, we can decompose (4.12) into the following two independent discrete-time aggregate monotone dynamics:

$$\begin{cases}
 x_i(u) = x_i(u-1) + x_i(u-1)[2 - \bar{\pi}(u-1)] \cdot [\pi_i(u-1) - \bar{\pi}(u-1)] \\
 x_i(v) = x_i(v-1) + x_i(v-1)[2 - \bar{\pi}(v-1)] \cdot [\pi_i(v-1) - \bar{\pi}(v-1)]
 \end{cases} \quad (4.13)$$

where $u = 2t, v = 2t+1$. The above two equations illustrate the underlying system dynamic hinged behind the double imitation policy under channel constraint under the approximation (4.7): it can be decomposed into two independent delayed aggregate monotone dynamics that alternatively occur at the odd and even iterations, respectively. The following theorem establishes the convergence of (4.13) to a unique fixed point which is also the NE of the spectrum access game G .

Theorem 4.10. *Starting from any initial point, the system described by (4.13) converges to a unique fixed point which is also the NE of the spectrum access game G .*

4.2.3.2 Imitation-Based Channel Access under Channel Constraint

Based on the theoretic results derived previously, we develop a fully distributed channel access policy for the general case with finite population based on the imitation rule among SUs on the same channel (i.e. neighbors). The proposed policy, detailed in Algorithm 6, is suitable both for proportional and double imitation. Run at each SU j and at each iteration, it consists of:

- sampling randomly one (proportional imitation) or two (double imitation) neighbors;
- comparing the payoff achieved at the previous iteration $t-1$ with that of the neighbor(s) selected for imitation;
- performing channel migration with the probability dictated by the applied imitation rule.

Algorithm 6 is evaluated by extensive simulations detailed in [96].

4.3 Retrospective Spectrum Access

We now consider the case where instead of imitating other nodes, a node imitates only its behavior that has brought him higher payoff in the past. Such “self-imitation” demonstrates more robustness in the case where imitating others is not possible or reliable.

Algorithm 6 Imitation-based Spectrum Access Policy under Channel Constraint: executed at each SU j

- 1: **Initialization:** set the imitation factor σ , the imitation threshold ϵ_U and the learning rate $\epsilon(t)$
- 2: Randomly choose a channel for the first two iterations $t = 0, 1$
- 3: **while** for each iteration $t \geq 2$ **do**
- 4: With probability $1 - \epsilon(t)$
- 5: Perform imitation in PISAP or DISAP on the same channel
- 6: With probability $\epsilon(t)$
- 7: Switch to a random channel
- 8: $t \leftarrow t + 1$:
- 9: **end while**

4.3.1 System Model and Game Formulation

We consider the downlink of primary network and SUs trying to opportunistically accessing the free spectrum (Fig. 4.1). The primary spectrum consists of a set \mathcal{C} of C frequency channels, each with bandwidth B . The users in the primary network are operated in a synchronous time-slotted fashion. A set \mathcal{N} of N SUs tries to opportunistically access the channels when they are left free by PUs.

Each SU j has a finite memory containing the history (strategies and payoffs) relative to the H_j past iterations. Let \mathcal{H}_j be the set of iterations recalled by SU j . Let $\xi_i(k)$ be the random variable equal to 1 when channel i is unoccupied by the PU at slot k and 0 otherwise. We assume that the process $\{\xi_i(k)\}$ is stationary and independent for each i and k . We also assume that at each time slot, channel i is free with probability μ_i , i.e., $\mathbb{E}[\xi_i(k)] = \mu_i$. We define an iteration t as a block of PU-slots of fixed duration T during which the SUs don't change their strategy (see Fig. 4.2). At the end of each iteration, SUs obtain a payoff which corresponds to the achieved throughput.

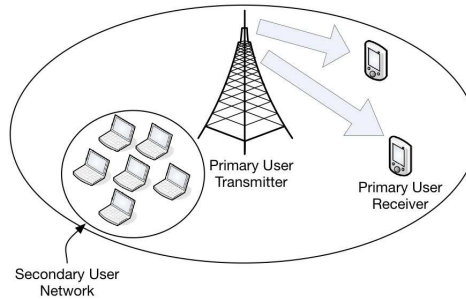


Figure 4.1: Network model.

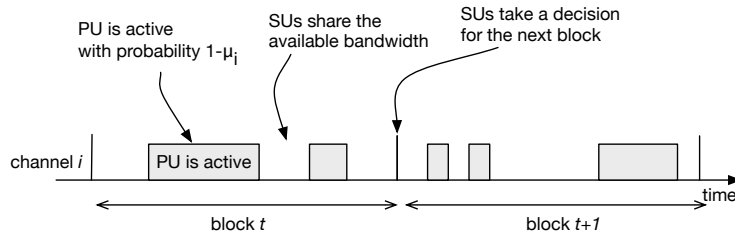


Figure 4.2: SU operation on a channel $i \in \mathcal{C}$.

In our work, each SU j is modeled as a rational decision maker, aiming at load-balancing the total system throughput. The *instantaneous* throughput it can achieve in terms of packets per second, denoted as T_j , can be expressed as a function of μ_{s_j} and n_{s_j} , where s_j denotes the channel which j chooses, and n_{s_j} denotes the number of SUs choosing channel s_j . The expected value of T_j , which has to be intended as the *long-term* throughput when T is very large, can be written as:

$$\mathbb{E}[T_j] = f(\mu_{s_j}, n_{s_j}). \quad (4.14)$$

In our work, SUs implement a generic random access protocol to avoid collisions. This yields:

$$f(\mu_{s_j}, n_{s_j}) = B\mu_{s_j}p(n_{s_j}), \quad (4.15)$$

where $p(n_{s_j})$ is a *decreasing* function denoting the successful transmission probability with n_{s_j} SUs interfering with SU j on channel s_j . B is a constant standing for the available bandwidth per channel. Without loss of generality, we will now assume that $B = 1$.

We next formulate the channel selection problem as a spectrum access game where the players are the SUs. The game is defined formally as follows:

Definition 4.2 (Spectrum access game). *The spectrum access game \mathcal{G} is a 3-tuple $(\mathcal{N}, \mathcal{C}, \{U_j(\mathbf{s})\})$, where \mathcal{N} is the player set, \mathcal{C} is the strategy set of each player. Let $\mathbf{s}_{-j} = \{s_1, \dots, s_{j-1}, s_{j+1}, \dots, s_C\}$ be the channels chosen by all users except user j . When a player j chooses strategy $s_j \in \mathcal{C}$, its player-specific utility function $U_j(s_j, \mathbf{s}_{-j})$ is defined as*

$$U_j(s_j, \mathbf{s}_{-j}) = \mathbb{E}[T_j] = \mu_{s_j}p(n_{s_j}).$$

The users struggle for maximizing their utility function and a commonly accepted solution for the game is a Pure Nash Equilibrium (PNE), which in our case can be thought as a mutually acceptable channel selection. More formally:

Definition 4.3 (Pure Nash Equilibrium). *A Pure Nash Equilibrium is a point \mathbf{s}^* in the action profiles space, from which no user has incentive to deviate unilaterally. Thus*

$$s_j^* \in \operatorname{argmax}_{s_j \in \mathcal{C}} U_j(s_j, \mathbf{s}_{-j}^*), \forall j \in \mathcal{N}. \quad (4.16)$$

We can recognize that \mathcal{G} is a congestion game with player-specific payoff functions. It then follows from [145] and [96] that \mathcal{G} possesses at least one PNE in the general case.

4.3.2 Retrospective Spectrum Access

We now develop a distributed retrospective spectrum access protocol (RSAP) that achieves a PNE of the spectrum access game. We firstly provide some definitions we shall need in the sequel analysis.

Define the state $z(t)$ of the system at iteration t by $z(t) \triangleq \{U_j(t-h)\}_{j \in \mathcal{N}, h \in \mathcal{H}_j}$. Let $\lambda_j = \operatorname{argmax}_{h \in \mathcal{H}_j} U_j(t-h)$ be the number of iterations passed from the SU j highest remembered payoff. Furthermore, let ρ_j denote the *inertia*, which is defined as a positive probability that SU j is unable to adjust its strategy at each iteration. Note that the concept of inertia has already been included in models of evolution with noise (see, e.g. [8]). In those cases however, inertia was defined as an

exogenous parameter, meaning that the probability of inertia could be take equal to zero. We will show in Proposition 4.5 that RSAP converges 1) in the general case if $\rho_j > 0$ for all j and 2) in the particular case where $H_j = 1$ and $\rho_j = 0$ for all j .

We now introduce the RSAP, as detailed in Algorithm 7. At each iteration t each user j applies the following revision scheme. With probability $(1 - \epsilon(t))(1 - \rho_j)$, SU j switches to channel $s_j(t - \lambda_j)$ if $U_j(t - \lambda_j) > U_j(t)$, and with probability $\epsilon(t)$ selects for the next iteration a channel with uniform distribution.

Algorithm 7 RSAP: executed at each SU j

- 1: **Initialization:** Set $\epsilon(t)$ and ρ_j .
 - 2: At $t = 0$, randomly choose a channel to stay, store the payoff $U_j(0)$ and set $U_j(t - h)$ randomly $\forall h \in \{1, \dots, H_j\}$.
 - 3: **while** at each iteration $t \geq 1$ **do**
 - 4: With probability $1 - \epsilon(t)$
 - 5: **if** $U_j(t - \lambda_j) > U_j(t)$ **then**
 - 6: Migrate to channel $s_j(t - \lambda_j)$ with probability $1 - \rho_j$
 - 7: **end if**
 - 8: With probability $\epsilon(t)$ switch to a random channel.
 - 9: **end while**
-

To characterize the equilibrium state of RSAP, we define a migration-stable state (MSS) as follows.

Definition 4.4 (Migration-stable state). *A migration-stable state ω is a state where no more migration is possible, i.e., $U_j(t) \geq U_j(t - h)$, $\forall h \in \mathcal{H}_j, \forall j \in \mathcal{N}$.*

4.3.3 Convergence Analysis

Foster and Young, with their pioneering work dated 1990 [77], were the first to argue that the Evolutionary Stable Strategy (ESS) does not capture the notion of long-run stability when the system is subjected to *continual* (rather than isolated) stochastic perturbations. In this new context, it is possible to identify a set of stochastically stable equilibria which consists of the states attained almost surely by a dynamical system when the noise level approaches zero. The identification of such system states is particularly useful in games with multiple equilibria (e.g., coordination games) as it permits to find out whether some outcomes are much more likely than others when the noise vanishes. Our protocol is characterized by stochastic perturbations and we study the small noise limit by making use of the tools provided in [72]. For the sake of a self-contained exposition, we include here some definitions and results we shall need.

4.3.3.1 Model of Evolution with Noise

Definition 4.5 (Model of evolution with noise [72]). *A model of evolution with noise or mistakes model is a triple $(Z, P, P(\epsilon))$ where:*

1. Z is the state space of a stochastic process X and is supposed to be finite;
2. $P = (p_{zz'})_{(z, z') \in Z^2}$ is a Markov transition matrix defined on Z ;

3. $P(\epsilon) = (p_{zz'}(\epsilon))_{(z,z') \in Z^2}$ is a family of Markov transition matrices on Z with $\epsilon \in [0, \bar{\epsilon})$ such that:

- (a) $P(\epsilon)$ is ergodic for $\epsilon > 0$;
- (b) $P(\epsilon)$ is continuous in ϵ and $P(0) = P$;
- (c) there is a cost function $c : Z^2 \rightarrow \mathcal{R}^+ \cup \{\infty\}$ s.t. for any pair of states (z, z') , $\lim_{\epsilon \rightarrow 0} \frac{p_{zz'}(\epsilon)}{\epsilon^{c_{zz'}/k}}$ exists and is strictly positive for $c_{zz'} < \infty$ and $p_{zz'}(\epsilon) = 0$ for small ϵ if $c_{zz'} = \infty$.

Definition 4.6 (Unperturbed and perturbed Markov chain). In a model of evolution with noise $(Z, P, P(\epsilon))$, (Z, P) is called the unperturbed Markov chain and, for any ϵ , $(Z, P(\epsilon))$ is a perturbed Markov chain. The family of perturbed Markov chains with ϵ is called a regular perturbation.

Remark. The fact that $P(\epsilon)$ is ergodic ensures that from any state $z \in Z$, we can reach any state $z' \in Z$ in a finite number of steps with positive probability. The unperturbed Markov chain is however not necessarily ergodic. If not, the Markov chain (Z, P) has one or more limit sets.

Definition 4.7 (Limit set). A limit set or recurrent class L of a Markov chain $X = (Z, P)$ is a set of states of X such that $\forall z \in L, P[X_{t+1} \in L | X_t = z] = 1$ and $\forall z, z' \in L$, there exists $\tau > 0$ s.t. $P[X_{t+\tau} = z' | X_t = z] > 0$.

The unperturbed Markov chain can be interpreted as the evolution of the system when players follow a predefined rule of evolution like *Best Response*. Noise ϵ can be interpreted as a probability that players do not follow the rule of the dynamics. For example, if the rule is *Best Response*, players choose the best response strategy at the next iteration step with probability $1 - \epsilon$ and choose any other strategy at random with probability ϵ . When a player does not follow the predefined rule, we say that there is a *mutation* by analogy with what happens in species evolution.

Definition 4.8 (State transition cost). The cost or resistance $c_{zz'}$ of the transition $z \rightarrow z'$ is the rate at which the transition probability $p_{zz'}(\epsilon)$ tends to zero as ϵ vanishes:

$$c_{zz'} = \begin{cases} 0 & \text{if } P_{zz'}(0) > 0 \\ k & \text{if } P_{zz'}(\epsilon) = (a_{zz'} + o(1))\epsilon^k \\ \infty & \text{if } P_{zz'}(\epsilon) = 0, \forall \epsilon \in [0, \bar{\epsilon}] \end{cases}$$

for some $\bar{\epsilon} > 0$.

Let $\mu(\epsilon)$ be the stationary probability distribution of the perturbed Markov chain $(Z, P(\epsilon))$.

Lemma 4.3 (Existence of limit distribution [225]). There exists a limit distribution

$$\mu^* = \lim_{\epsilon \rightarrow 0} \mu(\epsilon). \quad (4.17)$$

Thus, μ^* is a stationary probability of the unperturbed Markov chain (Z, P) :

Lemma 4.4 ([72]). The set of stochastically stable states is included in the limit sets of the unperturbed Markov chain (Z, P) .

Definition 4.9 (Long-run stochastically stable set). A state $z \in Z$ is said to be long-run stochastically stable if and only if $\mu_z^* > 0$.

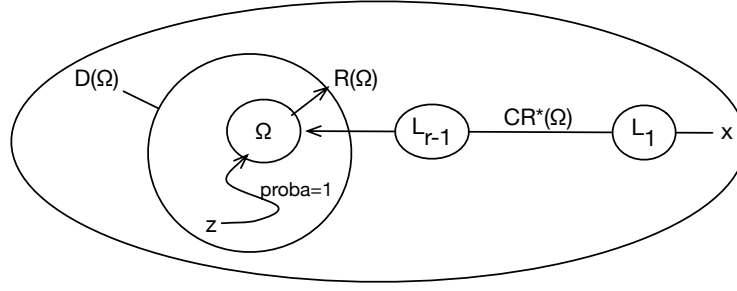


Figure 4.3: Illustration of the main concepts.

Let Ω be a union of one or more limit sets of (Z, P) . We now want to study the conditions for Ω to be stochastically stable. We also want to know the speed at which Ω is reached. For this purpose, [72] defines $W(x, \Omega, \epsilon)$ to be the expected time until set Ω is reached knowing that we start in state x and that the system follows the perturbed Markov chain $(Z, P(\epsilon))$. The goal is to characterize $\max_{x \in Z} W(x, \Omega, \epsilon)$.

4.3.3.2 Radius and Coradius Theorem

We start with some definitions of concepts illustrated in Fig. 4.3 before giving the main theorem. Define a path $(z_1, z_2, \dots, z_\tau)$ as a sequence of states.

Definition 4.10 (Basin of attraction). *Let Ω be a union of one or more limit sets of (Z, P) and let $(z_1, z_2, \dots, z_\tau)$ be a sequence of states. The basin of attraction $D(\Omega)$ of Ω is the set of initial states from which the unperturbed Markov chain converges to Ω with probability 1, i.e.:*

$$D(\Omega) = \{z \in Z \mid Pr[\exists \beta \text{ s.t. } \forall \tau > \beta, z_\beta \in \Omega \mid z_0 = z] = 1\}.$$

Definition 4.11 (Path cost). *For two sets X and Y , a path in Z is a sequence of states $(z_1, z_2, \dots, z_\tau)$ with $z_1, z_2, \dots, z_{\tau-1} \in X$ and $z_\tau \in Y$. The cost of the path is the sum as below:*

$$c(z_1, z_2, \dots, z_\tau) = \sum_{i=1}^{\tau-1} c_{z_i, z_{i+1}}.$$

Let $S(X, Y)$ be the set of all paths from X to Y and

$$C(X, Y) = \min_{(z_1, \dots, z_\tau) \in S(X, Y)} c(z_1, \dots, z_\tau)$$

be the set-to-set cost between X and Y . The radius of the basin of attraction of Ω is defined as the minimum number of mutations needed to leave $D(\Omega)$ given that we start in Ω .

Definition 4.12 (Radius). *The radius $R(\Omega)$ of Ω is the minimum cost of any path from Ω out of $D(\Omega)$, i.e.:*

$$R(\Omega) = C(\Omega, Z - D(\Omega)).$$

Definition 4.13 (Coradius). *The coradius $CR(\Omega)$ of Ω is defined by:*

$$CR(\Omega) = \max_{x \notin \Omega} \min_{(z_1, \dots, z_\tau) \in S(x, \Omega)} c(z_1, \dots, z_\tau).$$

In other words, the coradius is the maximum number of mutations needed to reach Ω .

Consider now (z_1, \dots, z_τ) a path from x to Ω . Let L_1, \dots, L_r be a set of consecutive limit sets with $L_r \subset \Omega$ and $L_i \not\subset \Omega$ for all $i < r$, through which the path passes. We define the *modified cost function* by subtracting from the initial cost function the intermediate radii of the limit sets L_i :

$$c^*(z_1, \dots, z_\tau) = c(z_1, \dots, z_\tau) - \sum_{i=2}^r R(L_i). \quad (4.18)$$

We now define the modified cost between a state x and Ω :

$$c^*(x, \Omega) = \min_{(z_1, \dots, z_\tau) \in S(x, \Omega)} c^*(z_1, \dots, z_\tau). \quad (4.19)$$

Definition 4.14 (Modified coradius). *The modified coradius of the basin of attraction of Ω is defined as:*

$$CR^*(\Omega) = \max_{x \notin \Omega} c^*(x, \Omega). \quad (4.20)$$

The theorem proposed by Ellison in [72] is a sufficient condition to identify a long-run stochastically stable set of the system. It also gives an lower bound on convergence rate.

Theorem 4.11 (Convergence to long-run stochastically stable set with modified cost [72]). *Let $(Z, P, P(\epsilon))$ be a model of evolution with noise, and suppose that for some set Ω which is a union of limit sets $R(\Omega) > CR^*(\Omega)$. Then:*

1. *The long-run stochastically stable set of the model is contained in Ω .*
2. *For any $y \notin \Omega$, $W(y, \Omega, \epsilon) = O(\epsilon^{-CR^*(\Omega)})$ as $\epsilon \rightarrow 0$.*

In other words, if it is more difficult to leave Ω and its basin of attraction than to come back to it, the long-run stochastically stable set is contained in Ω .

4.3.3.3 Main Results

We now establish the convergence of the retrospective spectrum access algorithm. We start by stating the following definitions required in the study of convergence.

Definition 4.15 (Single player improvement [79]). *A strategy profile s' is a single player improvement over the strategy profile s if it coincides with s in every coordinate except one, say coordinate j , and the payoff of player j is higher under s' than under s .*

Definition 4.16 (Weak finite improvement property [79]). *A game \mathcal{G} has the weak finite improvement property (weak-FIP) if from each strategy profile s there exists a finite sequence of single-player improvements that ends in a pure NE.*

We then recall the following result from [145]:

Theorem 4.12 ([145]). *Given a congestion game \mathcal{G} with player-specific decreasing payoff functions, the weak-FIP holds and \mathcal{G} admits at least one pure NE.*

We next analyze structural properties of the spectrum access game dynamics and evolution under the the retrospective spectrum access protocol.

Lemma 4.5. *Under the RSAP, there is a one-to-one mapping between the set of migration stable states and the limit set in the following cases:*

- in the general case with endogenous inertia $\rho_j > 0$,
- in the particular case $H_j = 1$ and $\rho_j = 0$ for all $j \in \mathcal{N}$.

Let Ω^* denote the union of all limit sets in pure NEs, we can establish the following properties on Ω^* .

Lemma 4.6. *It holds that $R(\omega) = 1, \forall \omega \notin \Omega^*$.*

Lemma 4.7. *Ω^* can be reached from any limit set $L \notin \Omega^*$ by stepwise mutations.*

Lemma 4.8. *It holds that $CR^*(\Omega^*) = 1$ and $R(\Omega^*) > 1$.*

We can then establish the convergence of the retrospective spectrum access protocol by applying Theorem 4.11.

Theorem 4.13 (Convergence of RSAP and convergence rate). *If all SUs 1) adopt the RSAP and 2) adopt a random strategy at each iteration with probability $\epsilon \rightarrow 0$, then the system dynamics converges a.s. to Ω^* , i.e. to a PNE of the game. The expected delay until a state in Ω^* is reached, given that the play in the ϵ -perturbed model begins in any state not in Ω^* , is $O(\epsilon^{-1})$ as $\epsilon \rightarrow 0$.*

Our study can be extended to other games possessing the weak-FIP. These include dominance solvable games, quasi-acyclic games (similar to acyclic games as defined by Young [225]), power set graphical congestion games⁵ and games with the finite improvement property (as defined by Monderer and Shapley [146]).

4.4 Summary of Other Contributions

In this section, we present a brief summary on a selection of our other work on game-theoretical algorithm design and distributed learning.

4.4.1 Joint Operator Pricing and Network Selection in CRNs

We consider a cognitive radio scenario which consists of primary and secondary networks, as well as a large set of cognitive users, and we focus on a fundamental issue concerning such systems, i.e. whether it is better for a cognitive user to act as a primary user, paying the primary operator for costlier, dedicated network resources with Quality of Service guarantees, or act as a secondary user (paying the Secondary operator), sharing the spectrum holes left available by licensed users and facing lower costs with degraded performance guarantees. At the same time, we consider the pricing problem of both Primary and Secondary operators, who compete with each other, setting access prices to maximize their revenues.

⁵Power set graphical congestion games are extended graphical congestion games where players can use any subset of resources.

The joint pricing and cognitive radio network selection problem is modeled as a Stackelberg game, where first the primary and secondary operators set their access prices in order to maximize their revenues. In this regard, we study both practical cases where (1) the primary and secondary operators fix access prices at the same time, and (2) the primary operator exploits his dominant position by playing first, anticipating the choices of the secondary operator. Then, network users react to the prices set by the operators, choosing which network they should connect to, therefore acting either like primary or secondary users.

The solution provides an insight on how rational users will distribute among existing access solutions (higher-price primary networks vs. lower-price secondary networks), i.e., the proportion of players who choose different strategies.

We then adopt a fluid queue approximation approach to study the steady-state performance of these users, focusing on delay as QoS metric. Besides considering static traffic equilibrium settings, we further formulate the network selection process of cognitive radio users as a population game, which provides a powerful framework for characterizing the strategic interactions among large numbers of agents, whose behavior is modeled as a dynamic adjustment process. More specifically, we study the cognitive users' behavior according to replicator dynamics, since such users adapt their choices and strategies based on the observed network state. We provide equilibrium and convergence properties of the proposed game, and derive optimum stable price and network selection settings.

4.4.2 Distributed Spectrum Management in TV White Space

We then investigate the distributed spectrum management problem in opportunistic TV White Space (TVWS) systems using a game theoretical approach that accounts for adjacent-channel interference and spatial reuse. TV Bands Devices (TVBDs) compete to access idle TV channels and select channel "blocks" that optimize an objective function. This function provides a tradeoff between the achieved rate and a cost factor that depends on the interference between TVBDs. We consider practical cases where contiguous or non-contiguous channels can be accessed by TVBDs, imposing realistic constraints on the maximum frequency span between the aggregated/bonded channels. We show that under general conditions, the proposed TVWS management games admit a potential function. Accordingly, the best response strategy allows us to determine the spectrum assignment of all players. This algorithm is shown to converge in a few iterations to a NE. Furthermore, we develop another algorithm based on imitation dynamics, where a TVBD probabilistically imitates successful selection strategies of other TVBDs in order to improve its objective function.

4.4.3 Load Balancing in Smart Grids

We also extend our methodology to study the load balancing in the smart grid. Specifically, we propose a fully distributed Demand-Side Management (DRM) system for smart grid infrastructures, especially tailored to reduce the peak demand of residential users. In particular, we use a dynamic pricing strategy, where energy tariffs are function of the overall power demand of customers.

We model our system using a game theoretical approach, considering two practical cases where (1) each appliance decides autonomously its scheduling in a fully distributed fashion (Single-Appliance DSM), and (2) each user must schedule all his home appliances (Multiple-Appliance

DSM). The proposed approach automatically ensures the reduction of the electricity demand at peak hours due to dynamic pricing.

We compare numerically these two cases, showing that the first is characterized only by a negligible performance degradation in all the considered grid scenarios. Nevertheless, while both mechanisms achieve almost the same performance level, the Multiple-Appliance DSM system requires a more complex architecture with a central server for each house that collects all appliances information and plays on behalf of the house-holder. Such an approach would increase the installation and operating costs due to the higher system complexity. On the contrary, in the Single-Appliance DSM system, one can use the processing and communication capabilities of devices that can autonomously optimize their usage, thus greatly simplifying the architecture design and system configuration.

We demonstrate that our game is a generalized ordinal potential game under some simple and very general conditions (viz., the regularity of the pricing function). Such feature guarantees some nice properties, such as the existence of at least one pure NE where no player has an incentive to deviate unilaterally from the scheduling pattern he decided upon. Furthermore, we show that any sequence of asynchronous improvement steps is finite and always converges to a pure NE.

4.5 Conclusion and Perspectives

This chapter mainly presents our work on the distributed learning algorithms in wireless networks. The key problem we address is how to design distributed algorithms that allow users to gradually converge to a stable and desirable system state based on location information and interactions? The core idea in our design is imitation, by imitating successful behaviors of either other users or itself.

Following the path of the chapter, one direction of further work is to extend our analysis, focused on single-hop networks, to the more generic multi-hop network paradigm. We expect to derive sufficient conditions, under which convergence to a network equilibrium can be achieved. Additional efforts are further called for if the equilibrium is far from the system optimum. Another direction is to investigate the scenario where the payoff observed by individual users are noisy, or even corrupted, and to develop robust learning algorithms with limited performance loss.

Chapter 5

Data Harvesting and Charging Path Optimization

5.1 Introduction

In this chapter, we focus on path optimization and the related scheduling problems arising from data harvesting and mobile charging. We present a generic analysis on these problems and design polynomial or quasi-polynomial time algorithms achieving constant or logarithmic approximation to the optimum.

5.1.1 Time-constrained Data Harvesting in Wireless Sensor Networks

We first consider the problem of data harvesting in wireless sensor networks. The task of data harvesting is traditionally accomplished by multi-hop forwarding, which is known to suffer from high energy consumption of forwarding nodes, especially those near the sink. Recently, as an efficient alternative, data harvesting using mobile devices, also termed as data mules [176] or data ferries [223], has been proposed and implemented in several applications such as underwater environmental monitoring [196]. The core idea can be summarized as follows: a data ferry (e.g., robot, vehicle) travels across the sensor field and harvests data from sensor nodes while they are within each other's communication range, and later transfers the harvested data to the sink. The use of data ferries in data harvesting can significantly reduce energy consumption at sensor nodes and thus increase network lifetime. However, as the data ferry can harvest data only when it travels close to the target node, it usually incurs longer data delivery latency, during which some delay-sensitive data may become obsolete. Therefore, optimizing the trajectory of the data ferry to limit or minimize data delivery latency is a primary concern for this approach to be effective in practice.

We investigate the trajectory optimisation problem in data collection applications for wireless sensor networks. This problem seeks an optimal data harvesting path to collect as much data as possible within a time duration. We call the problem *time-constrained data harvesting problem*. Specifically, our problem formulates the situation where delay-sensitive data needs to be reported to the sink within certain amount of time before they become obsolete. To make the analysis theoretically complete and generic, we investigate the generic m -dimensional context, of which the cases of $m = 1, 2, 3$ are particularly pertinent.

Our main results are naturally articulated as follows.

- *Theoretical performance bound.* We formulate the time-constrained data harvesting problem. We analytically characterize the performance bound of the optimal data harvesting algorithm. Our analysis demonstrates that in a network where nodes are randomly deployed with fixed density and the data ferry moves at constant speed, the quantity of harvested data does not scale with the number of nodes in the network under the random data harvesting algorithm, while this quantity scales logarithmically for the optimal algorithm design, indicating a significant performance gain when the network scales. Even though the trend is logarithmic, the gap can still be significant in large networks. In other words, a data harvesting algorithm not carefully chosen, such as randomly choosing a data harvesting path, can be very inefficient.
- *Approximation algorithm design.* We give a formal proof on the NP-hardness of the time-constrained data harvesting problem. Motivated by the NP-hardness of the problem, we focus on the design of constant-factor approximation algorithms. Given the complexity of the problem, we first study a specific scenario with non-overlapping neighborhoods, i.e., the network is sufficiently sparse such that the data ferry cannot harvest data from multiple sensors without changing its location. We then extend the analysis to the generic case with overlapping neighborhoods, i.e., the network is sufficiently dense such that the data ferry can harvest data from multiple sensors without changing location. As the key technicality in our design, we develop a methodology that relates the performance of topological paths to geometrical paths, based on which we mathematically prove the output of our algorithm is a constant-factor approximation of the optimal solution.

In our analysis, despite our focus on the data harvesting problem, the generic problem formulation of our work makes the analysis methodology and the obtained results broadly applicable to several engineering domains ranging from mobile charger scheduling, target monitoring to security patrolling, with a common generic objective of designing an optimal path such that a time-constraint utility function depending on the number of encountered targets is maximized.

5.1.2 Charging Path Optimization and Scheduling in Mobile Networks

In many emerging applications such as first responder, infrastructure monitoring, and scientific exploration, battery-powered mobile agents (e.g., sensors [220], robots [17], drones [65], and vehicles [161]) usually have specified tasks and mobility patterns [60]. To supply energy to these agents, mobile chargers are dispatched to visit these agents, which can significantly prolong the lifetime of mobile nodes. However, as the mobile charger only deliver energy to a target node when it encounters the node (or close to the node if wireless charging is used), inefficient path planning may incur long latency to pursue target nodes, result in dead nodes or even task failures. Therefore, optimizing the trajectory of the mobile charger is a primary concern for maintaining the operation of these systems.

Motivated by the above observation, we focus on the path optimization and charger scheduling problems with mobile nodes in mobile charging applications. To make our analysis generic and widely applicable, we do not impose any constraint on the mobility patterns of nodes, i.e., the trajectory of any node can be a curve of any form. We formulate a class of generic path optimization problems and concentrate on the problem of maximizing the number of nodes charged within a fixed time horizon. Our framework allows a variety of path optimization problems to be formulated with realistic constraints, such as limited time and energy budget. We prove that these problems are either NP-hard or APX-hard. We design a quasi-polynomial time algorithm that

achieves logarithmic approximation to the optimum charging path. We also demonstrate how our approximation algorithm can be adapted and extended to solve other charging path optimization and scheduling problems.

Again, despite our focus on mobile charging, the generic formulation of the problem makes our analysis methodology and obtained results applicable to a wide range of related problems such as data mule scheduling, package delivery, target monitoring, and security patrolling. These problems have a common generic objective of designing an optimum path and the corresponding scheduling that maximize the number of encountered targets under a given budget in terms of time or energy, or minimizes the cost of encountering a given minimum number of targets.

5.1.3 Chapter Organization

The rest of this chapter is structured as follows. Section 5.2 develops our work on the time-constrained data harvesting. Section 5.3 presents our work on the charging path optimization and scheduling. Section 5.4 concludes the chapter by briefly summarizing our other related work related to this topic. More details of our work on this topic including proofs and numerical analysis can be found in our publications [46, 48, 51, 93].

5.2 Time-constrained Data Harvesting in Wireless Sensor Networks

5.2.1 Background and Related Work

The problem we address and our methodology are related to the following research fields.

Data ferry Assisted Data Harvesting. There is a large body of existing work on data ferry assisted data harvesting [59, 135, 182, 222, 231] (cf. [69] for a comprehensive survey). The problem we address is the optimisation of data harvesting trajectory of the data ferry, which is a hard problem in general, since we are constrained in both space (communication range between the data ferry and sensors) and time domain (limiting data harvesting latency). Existing solutions contour this difficulty by either using simple mobility and communication models [59, 135, 182, 222, 231] or assuming that the trajectory is already given [182].

The authors in [108, 223] address a similar problem of designing data harvesting path for data ferries to minimize the data harvesting latency under the constraint that all sensors are visited. The algorithms they propose are based on the well-known travel salesman problem (TSP) [16] and its variant TSP with neighbors (TSPN) [66]. However, our problem is different because TSP requires the path to pass all sensors while we seek the most profitable path to harvest maximum data given the time constraint. Our problem formulation complements the TSP formulation and is particularly pertinent when the network is large and it is impossible for the data ferry to traverse every node.

Mobile Charger Scheduling. Another similar problem is the mobile agent scheduling problem where a mobile charger needs to travel within the charging range of each sensor node to recharge them under the constraint of the battery life of sensor nodes, which is similar to the time constraint in our data harvesting problem (cf. [218, 221, 235] and references therein). However, they rely on additional assumptions or simplifications to make the problem tractable. For example, the authors of [221] find out a near-optimum traveling path to recharge all sensor nodes using linear programming, assuming the traveling speed being infinite, and then remove this assumption and

derive a bound of performance degradation. However, their algorithm implicitly assumes the travelling is fast enough. In our work, we remove these assumptions and analytically establish the performance properties of the proposed data harvesting algorithm.

Related Theoretical Problems. From a theoretical point of view, the problem we address is related to several fundamental problems in theoretical computer science, particularly the Orienteering problem [33] and the weight-constrained minimum spanning tree problem [3]. In the Orienteering problem, each node of a given graph has certain quantity of reward. The problem is to find a path that maximizes the reward collected, subject to a constraint on the path length. In the weight-constrained minimum spanning tree problem, each edge has a cost and weight. The problem is to find a spanning tree with minimum total cost subject to an upper-bound on the total weight. Both problems are NP-hard and have constant-factor approximation algorithms. However, these approximation algorithms cannot be directly applied in our problem as they are focused on topological paths.

5.2.2 Time-constrained Data Harvesting Problem

5.2.2.1 Network Model

We consider a sensor network composed of n nodes, denoted by the set $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$, deployed in an m -dimensional Euclidean cube $[0, D]^m$. We are interested in the asymptotic scenario where both n and D are large with the node density $\lambda = \frac{n}{D^m}$ being a constant. We note that our motivation of considering a generic m -dimensional problem is to make our analysis mathematically complete and generically applicable. For the particular problem of time-constrained data harvesting in wireless sensor networks, the cases with $m = 1, 2$, and 3 are relevant. Tab. 6.2 lists the major notations used in this section.

In the considered sensor network, each node v_i has unit data message¹ to be harvested by a data ferry, denoted by s , moving at a constant speed. To harvest data generated by v_i , s needs to move into the communication range of v_i , which is modeled as an m -dimensional ball D_i centered at v_i with radius r . We call D_i the neighborhood of node v_i . By slightly abusing notations, we also use D_i to denote the border of the ball which is an m -dimensional sphere. We use \mathcal{P} to denote the possible path set for s . For a path $P \in \mathcal{P}$, we denote $d(P)$ the Euclidean length of P . We say that a path P covers a point M if there exists a point on P within distance r to M . In other words, if s moves along P , it can harvest the data generated by all the nodes that it covers. Denote $\Lambda(P)$ the number of nodes P covers. The following example further illustrates the notations and terminologies in our study.

Example 5. Consider the two-dimensional network illustrated in Fig. 5.5 composed of three sensor nodes v_1, v_2, v_3 with the circles around them indicating their neighborhoods D_1, D_2, D_3 . The path P covers both v_1 and v_2 , but not v_3 . When moving along P , s can harvest data generated at both v_1 and v_2 . We thus have $\Lambda(P) = 2$. $d(P)$ is the Euclidean length of P .

¹The case where nodes generate multiple data messages can be tackled by devising the node generating k unit data messages to k virtual nodes at the same position, each generating unit data message.

Table 5.1: Notations

\mathcal{V}	Set of nodes in the network
v_i	Node i
s	The mobile sink (data ferry)
n	Number of nodes in the network
D	Side length of the network area
m	Network dimension
λ	Node density
r	Communication range of nodes
D_i	Neighborhood ball of node v_i
\mathcal{P}	Path set
$\Lambda(P)$	Number of nodes covered by path P
P^*	The optimal path maximizing harvested data
\mathcal{P}_t	Topological path set
\mathcal{P}_g	Geometrical path set
\mathcal{P}_b	Backbone path set
$d(P)$	Euclidean length of path P
T	Time duration within which data need to be deposited, T is also the maximum path length s can traverse before deposing data moving at unit speed

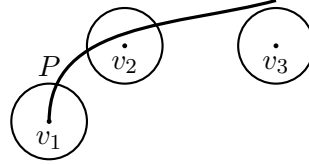


Figure 5.1: An example illustrating notations and terminologies.

5.2.2.2 Problem Formulation

We consider the data harvesting problem faced by s to seek an optimum path to harvest as much data as possible within a time duration T . The problem we address models the situation where delay-sensitive data should be reported to the sink within certain time in order to be further analysed. To make the notation concise, we let s move at unit speed and thus T is the maximum path length s can traverse before depositing the harvested data. The results obtained can be easily scaled to arbitrary speed by scaling the time duration T . Throughout our analysis, we are interested in the non-trivial case where $r \ll T$ and $Tr^{m-1} \ll D^m$, i.e., the maximum path length is much longer than the communication range, while the space covered by a path of length T is much smaller than the network space. The time-constrained data harvesting problem is formalized below.

Problem 5.1 (Time-constrained Data Harvesting Problem). *The time-constrained data harvesting problem is as follows:*

$$\text{maximize } \Lambda(P), \text{ subject to } d(P) \leq T.$$

That is, s seeks the optimal path $P^* \in \mathcal{P}$ of Euclidean length $d(P^*) \leq T$, along which it can harvest

the maximum quantity of data. When there are more than one maximum, the optimal path P^* is the one with minimum Euclidean length.

It is worth noting that the time-constrained data harvesting problem has a number of important variants. In some applications, we require that the data harvesting path to be a cycle or have predefined starting and end points; it is sometimes required to differentiate sensor nodes by giving weights to them (e.g., giving higher weights to sensors at key positions) and seek the path maximizing the weighted sum of harvested data; furthermore, we may dispose multiple data ferries to for data harvesting. Many of these variants can be addressed using the framework established in this section to design and optimize data harvesting path.

5.2.3 Theoretical Performance Bound: Optimal and Random Algorithms

We establish the theoretical foundation of the time-constrained data harvesting problem by quantifying the performance of the optimal data harvesting algorithm and a natural algorithm where the data harvesting path is randomly chosen. In our analysis, we assume that nodes are randomly deployed in $[0, D]^m$.

5.2.3.1 Random Data Harvesting Algorithm

A simple data harvesting algorithm is to randomly choose a data harvesting path of length T . We call this algorithm random data harvesting algorithm, termed concisely as random algorithm. Our motivation of starting with the random algorithm is two-fold:

- It is a natural strategy and easy to implement;
- It provides a reference for performance comparison with more sophisticated algorithms as well as the optimal one.

The following theorem states the main result.

Theorem 5.1 (Performance of Random Data Harvesting Algorithm). *Consider the random data harvesting algorithm where s randomly chooses a path P of length T , it holds that*

- $\mathbb{E}[\Lambda(P)] = O(\lambda r^{m-1}T)$;
- $\Pr \{\Lambda(P) \geq n^\epsilon \mathbb{E}[\Lambda(P)]\} \rightarrow 0$, when $n \rightarrow \infty$, $\forall \epsilon > 0$, that is, $\Pr \{\Lambda(P) = \Theta(n^\epsilon)\} \rightarrow 0$.

With Theorem 5.1, we have the following result:

- In average, the expected harvested data for the random algorithm does not scale with respect to either the population size n of the network or its geometrical size D ;
- With high probability, we cannot expect better outcome than $\Theta(\lambda r^{m-1}T)$.

5.2.3.2 Optimal Data Harvesting Algorithm

Having derived the performance of the random algorithm, we proceed to investigate the performance of the optimal data harvesting algorithm, as stated in Theorem 5.2.

Theorem 5.2 (Performance of Optimum Algorithm). *Let P^* denote the path of the optimal data harvesting algorithm, it holds that*

- $\mathbb{E}[\Lambda(P^*)] = \Theta\left(\frac{\log n}{\log \log n}\right)$;
- $\Pr\left\{\Lambda(P^*) = \Theta\left(\frac{\log n}{\log \log n}\right)\right\} \rightarrow 1$, when $n \rightarrow \infty$.

The intuition behind Theorem 5.2 (cf. [46]) is that by the bins and balls problem, in average we can find a region in the network such that it contains at least $\Theta\left(\frac{\log n}{\log \log n}\right)$ nodes and all the nodes in the region can be covered by a data harvesting path of length T . We have also shown in the proof that this bound is tight.

5.2.3.3 Discussion

Comparing the performance of optimal and random data harvesting algorithms, we can observe that when the network scales, especially when $n \rightarrow \infty$, the optimal algorithm significantly outperforms the random one. Even though the trend is logarithmic not polynomial or exponential, the gap can still be significant in large networks. In other words, a data harvesting algorithm not carefully chosen, such as randomly choosing a harvesting path, can be very inefficient. This motivates our second part of work on the following fundamental question:

How to design efficient data harvesting algorithms that approaches the solution of Problem 5.1?

Remark. Theorem 5.2 establishes the performance of the optimal algorithm. However, it does not specify how the optimal path can be constructed given a network instance. Choosing the path as indicated in the first step in the proof of Theorem 5.2 only performs well in the average sense when a large number of instances are executed, but it cannot give the optimal path for a given network instance. In fact, as we will show in the next subsection by Theorem 5.6, the problem of constructing the optimal path as formulated in Problem 5.1 is NP-hard.

5.2.4 Approximation Algorithm Design

We first show that Problem 5.1 is NP-hard. We then design constant-factor approximation data harvesting algorithms with polynomial-time complexity.

Theorem 5.3 (NP-hardness of Time-constrained Data Harvesting Problem). *Problem 5.1 is NP-hard.*

The proof, detailed in [46], consists of relating Problem 5.1 to the TSP which is NP-hard.

5.2.4.1 Non-overlapping Neighborhood Case

Given the complexity of the time-constrained data harvesting problem, we first investigate a specific scenario where the neighborhoods of any two nodes are non-overlapped (i.e., $D_i \cap D_j = \emptyset$, $\forall v_i, v_j \in \mathcal{V}$) and develop an approximation algorithm for Problem 5.1. We start by the following definition of topological path.

Definition 5.1 (Topological path). *A path P_t is called a topological path in a graph if P_t is composed of uniquely the edges in the graph.*

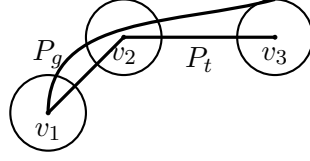


Figure 5.2: A topological path P_t and a geometrical path P_g , both covering 3 nodes.

Generically, we call a path *geometrical path*, denoted as P_g for presentation clarity, to emphasize that P_g is not necessarily a topological path as P_g may contain curves and may start and end at any point. Of course, a topological path is also a geometrical one, i.e., let \mathcal{P}_g and \mathcal{P}_t denote the sets of geometrical and topological paths, it holds that $\mathcal{P}_t \subset \mathcal{P}_g$. Fig. 5.2 illustrates the notions of topological and geometrical paths.

The key element towards designing approximation algorithms for Problem 5.1 is to establish the relationship between geometrical and topological paths in terms of path length and number of covered nodes. This relationship is established in two steps:

- *Step 1:* We show that any geometrical path P_g can be approximated by a topological path P_t such that

$$d(P_t) = O(d(P_g)), \text{ and } \Lambda(P_t) = \Lambda(P_g).$$

- *Step 2:* We show that any topological path P_t can be approximated by a geometrical path P_g via a *geometrisation* procedure that we develop such that

$$d(P_g) = O(d(P_t)), \text{ and } \Lambda(P_g) \geq \Lambda(P_t).$$

We start by the first step to approximate a geometrical path P_g by a topological path P_t . By slightly abusing the notation, for a given path P , we reuse $\Lambda(P)$ to denote an ordered set of nodes covered by P . Using this notation, a topological path P_t can be uniquely noted by $\Lambda(P_t)$. Taking the topological path P_t in Fig. 5.2 as an example, P_t can be denoted by $\{v_1, v_2, v_3\}$. Given an ordered set of nodes $\mathcal{V}_g = \{v_1, v_2, \dots, v_{|\mathcal{V}_g|}\}$, for any geometrical path P_g with $\Lambda(P_g) = \mathcal{V}_g$, we construct a topological path $P_t = \mathcal{V}_g$. It holds that $d(P_t) = O(d(P_g))$ and P_t covers all nodes in \mathcal{V}_g . Let the geometrical path P_g^* denote the geometrical path of minimum length among those covering \mathcal{V}_g , it holds that $d(P_t) = \Theta(d(P_g^*))$. This approximation result is mathematically formalized in Lemma 5.1. The proof consists of proving that $1 \leq \frac{d(P_t)}{d(P_g^*)} \leq 2$ and that the maximum of the ratio is achieved when the neighborhoods of nodes D_i are “packed” one to the other [46].

Lemma 5.1. *Given an ordered set of nodes \mathcal{V}_g , $\forall P_g$, $\Lambda(P_g) = \mathcal{V}_g$, let $P_t = \mathcal{V}_g$, it holds that $d(P_t) = O(d(P_g))$. Particularly, let $P_g^* = \operatorname{argmin}_{\Lambda(P_g)=\mathcal{V}_g} d(P_g)$, it holds that $d(P_t) = \Theta(d(P_g^*))$.*

We then proceed to the second step to approximate a topological path P_t by a geometrical path P_g by introducing geometrisation, formally defined in the following.

Definition 5.2 (Geometrisation). *Given a topological path P_t , the geometrisation procedure finds a geometrical path P_g that approximates P_t . By approximation we require that*

$$d(P_t) = \Theta(d(P_g)), \text{ and } \Lambda(P_t) \leq \Lambda(P_g).$$

Algorithm 8 details the proposed geometrisation procedure, whose core part is further illustrated in Fig. 5.3. It is straightforward to see that $d(P_g) < d(P_t)$. One technical point worth commenting is how to find M_i on D_i such that $|\overline{M_{i-1}M_i}| + |\overline{M_iv_{i+1}}|$ is minimized (line 6). M_i can be efficiently found by using the following technique: consider the outside border of D_i as a mirror; let a light beam be emitted from M_{i-1} and then be reflected by D_i to reach v_{i+1} ; it follows from the theory of optics that light always travels using the shortest path; hence M_i corresponds to the reflection point of the light beam on D_i and can be found geometrically by equalizing the angle of incident and the angle of reflection.

Algorithm 8 Geometrisation

Input: Topological path P_t passing nodes in \mathcal{V}_t

Output: Geometrized path P_g

- 1: Denote the intersection point of v_1v_2 and D_1 by M_1 ;
 - 2: **for** $i = 2$ to $|\mathcal{V}_t| - 1$ **do**
 - 3: **if** $\overline{M_{i-1}v_{i+1}}$ covers D_i **then**
 - 4: Denote the first intersection point between $\overline{M_{i-1}v_{i+1}}$ and D_i by M_i ; // See Fig. 5.3 (left);
 - 5: **else**
 - 6: Find a point M_i on D_i such that $|\overline{M_{i-1}M_i}| + |\overline{M_iv_{i+1}}|$ is minimized; // See Fig. 5.3 (right);
 - 7: **end if**
 - 8: **end for**
 - 9: Denote the intersection point of $\overline{M_{|\mathcal{V}_t|-1}v_{|\mathcal{V}_t}}|$ and $D_{|\mathcal{V}_t|}$ by $M_{|\mathcal{V}_t|}$;
 - 10: Return $P_g = \{\overline{M_1M_2}, \dots, \overline{M_{|\mathcal{V}_t|-1}M_{|\mathcal{V}_t}}|\}$;
-

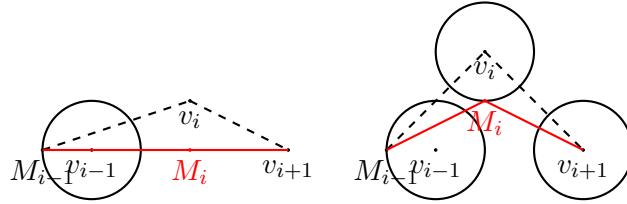


Figure 5.3: Illustration of the core part of Algorithm 8.

It is worth mentioning that the for loop in Algorithm 8 can be repeated so as to further improve geometrisation effectiveness (i.e., decrease $d(P_g)$). To make this clearer, let $P_g^{j-1} = \{\overline{M_1^{j-1}M_2^{j-1}}, \dots, \overline{M_{|\mathcal{V}_t|-1}^{j-1}M_{|\mathcal{V}_t}^{j-1}}\}$ denote the output of Algorithm 8 at iteration $j - 1$, for iteration j , it suffices to set $P_t = P_g^{j-1}$ by letting $v_k = M_k^{j-1}$ ($2 \leq k \leq |\mathcal{V}_t| - 1$) in the algorithm. We observe via simulation that that the improvement is not significant or even negligible when Algorithm 8 is executed more than a handful of times.

After establishing the relationship between geometrical and topological paths, we are now ready to present the global algorithm for Problem 5.1, as detailed in Algorithm 9.

The core idea of Algorithm 9 is as follows: for each node pair, we find the topological path $\Pi_t(i, j)$ passing the maximum number of nodes in \mathcal{V} whose geometrized path $\Pi_g(i, j)$ satisfies $d(\Pi_g(i, j)) \leq T$; we then return $\Pi^* = \operatorname{argmax}_{\Pi_g(i, j), \forall v_i, v_j \in \mathcal{V}} \Lambda(\Pi_g(i, j))$. The two building blocks in Algorithm 9 is the geometrisation algorithm (Algorithm 8) and the algorithm of max-prize path in [33]. Given a graph in which each node has a certain amount of prize, the max-prize algorithm finds in

Algorithm 9 Approximation algorithm solving Problem 5.1: non-overlapping neighborhood case

Input: Coordinates of nodes in \mathcal{V}
Output: Π^* : a constant factor approximation of P^*

- 1: Construct a complete graph G with node set \mathcal{V} ; set the length of the edge between v_i and v_j to be $\overline{v_i v_j}$;
 - 2: For each node pair (v_i, v_j) , find the topological path $\Pi_t(i, j)$ passing the maximum number of nodes in \mathcal{V} whose geometrized path $\Pi_g(i, j)$ satisfies $d(\Pi_g(i, j)) \leq T$ using Algorithm 8 and the algorithm of max-prize path in [33] by setting the prize for each node to be 1;
 - 3: Return $\Pi^* = \underset{\Pi_g(i, j), \forall v_i, v_j \in \mathcal{V}}{\operatorname{argmax}} \Lambda(\Pi_g(i, j))$;
-

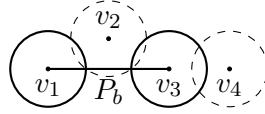


Figure 5.4: An example of MIS composed of nodes v_1 and v_3 .

polynomial time a path collecting the maximum quantity of prize whose length is bounded by a constant, given as an input parameter. The following theorem formally establishes the performance of Algorithm 9.

Theorem 5.4 (Performance of Algorithm 9). *Algorithm 9 returns Π^* within polynomial time. It holds that $\Lambda(\Pi^*) = \Theta(\Lambda(P^*))$, where P^* denotes the optimal data harvesting path under time constraint T .*

5.2.4.2 Overlapping Neighborhood Case

We next extend our efforts to study the generic case with overlapping neighborhoods.

We first construct a graph G' whose node set is \mathcal{V} and there is an edge between v_i and v_j if $\overline{v_i v_j} \leq 2r$. We then construct a maximal independent set (MIS)² of G' using a coloring algorithm similar as presented in [108, 199], detailed in Algorithm 10 for completeness. Fig. 5.4 illustrates an example of MIS composed of nodes v_1 and v_3 .

We then define backbone topological paths, which can be regarded as topological paths using nodes in the MIS \mathcal{U} .

Definition 5.3 (Backbone Topological path). *A path P_b is called a backbone topological path, or backbone path for short, in a graph if P_b is composed of uniquely the edges whose endpoints are in the MIS of the graph except the source and the destination nodes.*

As in the case of non-overlapping neighborhood, we call a path *geometrical path*, denoted as P_g , to emphasize that P_g is not necessarily a backbone path. Of course, a backbone path is also a topological path, and a geometrical one: i.e., let \mathcal{P}_g , \mathcal{P}_t and \mathcal{P}_b denote the sets of geometrical,

²An independent set (IS) of an undirected graph is a subset \mathcal{U} of nodes such that no two nodes in \mathcal{U} are neighbors. An IS is maximal if no node can be added to \mathcal{U} without violating IS. A maximal IS, or MIS, can be found in polynomial-time. Note that a related concept, a maximum IS (called MaxIS), is one IS of maximum cardinality. Finding MaxIS, however, is NP-hard.

Algorithm 10 MIS Construction of G' **Input:** Graph G' **Output:** MIS \mathcal{U}

- 1: **Initialisation:** Set $\mathcal{U} = \emptyset$; Color all D_i ($v_i \in \mathcal{V}$) white;
- 2: **repeat**
- 3: Color a white ball D_i black and add v_i into \mathcal{U} ;
- 4: Color every white ball D_j gray if v_j is v_i 's neighbor;
- 5: **until** there is no white ball
- 6: Return \mathcal{U} ;

topological and backbone paths, it holds that $\mathcal{P}_b \subset \mathcal{P}_t \subset \mathcal{P}_g$. As an example, the path P_b is Fig. 5.4 is a backbone path.

We apply the same analysis and design methodology in the non-overlapping neighborhood case and adapt it in the overlapping neighborhood case. A point M is said to be touched by path P if the minimum distance between any point of P and M is larger than r but smaller or equal to $2r$. The key element of designing approximation algorithm for Problem 5.1 with overlapping neighborhoods is to establish the relationship among geometrical, backbone, and geometrized backbone paths in terms of path length and number of touched and covered nodes. Specifically, we establish the relationship two steps:

- *Step 1:* We show that any geometrical path P_g can be approximated by a backbone path P_b such that $d(P_b) = O(d(P_g))$ and $\forall v_i$ covered by P_g , v_i is either covered or touched by P_b ;
- *Step 2:* We show that any geometrical path P_g can be approximated by another geometrical path P'_g geometrized from a backbone path P_b via a *backbone geometrisation* procedure such that

$$d(P'_g) = O(d(P_g)), \text{ and } \Lambda(P'_g) \geq \Lambda(P_g).$$

We start with the first step by showing the following lemma. The proof uses similar reasoning technique as the proof of Lemma 5.1 detailed in [46].

Lemma 5.2. *Given any geometrical path P_g , there exists a backbone path P_b such that $d(P_b) = O(d(P_g))$ and $\forall v_i$ covered by P_g , v_i is either covered or touched by P_b . Particularly, let $P_g^* = \underset{\Lambda(P_g)=\mathcal{V}_g}{\operatorname{argmin}} d(P_g)$, it holds that $d(P_b) = \Theta(d(P_g^*))$.*

We then proceed to approximate a backbone path P_b by a geometrical path P_g by introducing *backbone geometrisation*, formally defined in the following.

Definition 5.4 (Backbone Geometrisation). *Given a backbone path P_b , the backbone geometrisation procedure finds a geometrical path P_g that approximates P_b . By approximation we require that $d(P_b) = \Theta(d(P_g))$, and $\Lambda(P_b) \leq \Lambda(P_g)$.*

In [108], the authors develop a polynomial-time backbone geometrisation algorithm, which will be used in our design.

The following lemma, which follows straightforwardly from Lemma 5.2 and the backbone geometrisation algorithm, approximates a geometrical path by another geometrical path geometrized from a backbone path.

Lemma 5.3. *Given any geometrical path P_g , there exists a path P'_g geometrized from a backbone path P_b such that*

$$d(P'_g) = O(d(P_g)), \text{ and } \Lambda(P'_g) \geq \Lambda(P_g).$$

After establishing the relationship among geometrical, backbone and geometrized backbone paths, we now present the design of the global approximation algorithm for Problem 5.1 for the overlapping neighborhood case, as detailed in Algorithm 11.

Algorithm 11 Approximation algorithm solving Problem 5.1: overlapping neighborhood case

Input: Coordinates of nodes in \mathcal{V}

Output: Π^* : a constant factor approximation of P^*

- 1: Construct a graph G' whose node set is \mathcal{V} and there is an edge between v_i and v_j if $\overline{v_i v_j} \leq 2r$;
 - 2: Run Algorithm 10 on G' to construct an MIS \mathcal{U} ;
 - 3: Construct a complete graph G with node set \mathcal{V} ; set the length of the edge between v_i and v_j to be $\overline{v_i v_j}$;
 - 4: For each node pair (v_i, v_j) , with the MIS \mathcal{U} constructed in 2, find the backbone path $\Pi_b(i, j)$ passing the maximum number of nodes in \mathcal{V} whose geometrized path $\Pi_g(i, j)$ satisfies $d(\Pi_g(i, j)) \leq T$ using the algorithm in [108] and the algorithm of max-prize path in [33] by setting the prize for each node i to be the number of nodes covered or touched by D_i ;
 - 5: Return $\Pi^* = \underset{\Pi_g(i, j), \forall v_i, v_j \in \mathcal{V}}{\operatorname{argmax}} \Lambda(\Pi_g(i, j))$;
-

The core idea of Algorithm 11 is as follows: for each node pair (v_i, v_j) , we find the backbone path $\Pi_b(i, j)$ passing the maximum number of nodes in \mathcal{V} whose geometrized path $\Pi_g(i, j)$ satisfies $d(\Pi_g(i, j)) \leq T$; we then return $\Pi^* = \underset{\Pi_g(i, j), \forall v_i, v_j \in \mathcal{V}}{\operatorname{argmax}} \Lambda(\Pi_g(i, j))$. The two building blocks in Algorithm 9 is the backbone geometrisation algorithm [108] and the algorithm of max-prize path [33]. When running the algorithm of max-prize path, we set the prize of each node v_i to be the number of nodes covered or touched by D_i , which allows us to achieve constant-factor approximation. The following theorem establishes the performance of Algorithm 11.

Theorem 5.5 (Performance of Algorithm 11). *Algorithm 11 returns Π^* within polynomial time. It holds that $\Lambda(\Pi^*) = \Theta(\Lambda(P^*))$, where P^* denotes the optimal data harvesting path.*

The time complexity of Algorithm 9 and Algorithm 11 is $O(n^5)$ following that the complexity of the max-prize path is $O(n^3)$. The approximation ratio can be derived from the approximation ratio of the prize-collecting problem (approximately 2) and the geometrisation in the non-overlapping case (2) and the backbone geometrisation in the overlapping case $(1 + \frac{20}{\pi})$ for the algorithm in [108]). The overall approximation ratio is thus 4 and $2(1 + \frac{20}{\pi})$ in the non-overlapping and overlapping cases.

5.3 Charging Path Optimization and Scheduling in Mobile Networks

In this section, we proceed to study the charging path optimization and the related scheduling problems. Compared to the data harvesting problem, a fundamental difference is that now the network nodes are also mobile and the design of charging path should take the mobility into account.

5.3.1 Background and Related Work

Path Optimization in Vehicular Routing and Data Ferry Assisted Data Harvesting. There have been a significant amount of research works on the Vehicular Routing Problem (VRP) [193]. In general, the VRP seeks to optimize the routing decisions of a single or fleet of vehicles to deliver goods to specified locations according to demand requirements and other specified constraints. The VRP has many variants based on the application scenarios and constraints. Some recent papers deal with optimizations with dynamism [87, 151], uncertainty [142], and many real-life constraints [224, 233]. These works provide a rich foundation for related algorithmic research. However, it is commonly assumed in these works that the targets are stationary and the vehicle travels through a set of fixed locations.

Another related problem is path optimization in data ferry assisted data harvesting [53, 59, 182, 222] (cf. [69] for a comprehensive survey), where a data ferry (e.g., robot, vehicle) travels across the sensor field and harvests data from sensor nodes. Again, existing works focus on finding the optimum path of minimum length in the setting where sensor nodes are stationary.

Mobile Charger Scheduling. The mobile charger scheduling problem [92, 164] seeks paths for mobile chargers (e.g., mobile robots) to replenish batteries for sensor networks. Many variants are studied by introducing different optimization goals, application scenarios, and constraints. For example, the authors of [235] consider collaborate mobile charger scheduling, where different mobile chargers can recharge each other so that the chargers can cover a larger area. In [157], the authors assume that the charging time for sensor nodes is much longer than the mobile charger's traveling time. Their goal is to maximize the timespan that all sensor nodes are alive. In [221], the authors assume that the charger can recharge all the sensor nodes lying within a distance through wireless power transfer. Their goal is to find a charging path and stopping points to minimize the charging time. In a recent paper [90], the authors consider the power heterogeneity of sensor nodes. They divide sensor nodes into groups, and apply the TSP algorithms to recharge nodes within each group. In [62], the problem of ensuring monitoring quality for stochastic events is studied. When the traveling time of the mobile charger is ignored, this problem can be reformulated as a sub-modular problem. A polynomial time algorithm with constant time approximation ratio is then developed.

Existing works on mobile charger scheduling all assume that the nodes are fixed in locations. In contrast, we study a more generic and practical scenario of moving target nodes, and develop a quasi-polynomial time scheduling algorithm that achieves logarithmic approximation.

Traveling Salesman Problem and Orienteering Problem. The Traveling Salesman Problem (TSP) is a class of combinatorial optimization problems which have been extensively studied. Many approximation algorithms have been proposed [16, 117]. A relevant extension to the TSP is the deadline-TSP or the TSP with time window, where each node can be visited only within a time interval [21, 22, 33, 38].

In [33] and [38], the authors study the Max-Prize Path problem, also referred to as the Orienteering problem, where the goal is to visits as many nodes as possible, but only before a hard time deadline D . The APX-hardness of the Orienteering problem can be shown via reduction from the TSP on bounded metrics which is APX-hard [33]. For the Orienteering problem on undirected graphs, the best approximation ratio in the literature is $2 + \epsilon$ [38]. For the Orienteering problem on directed graphs, the best approximation ratio is $O(\log^2 OPT)$ [38].

Our work can be regarded as a non-trivial extension of the Orienteering problem in a more

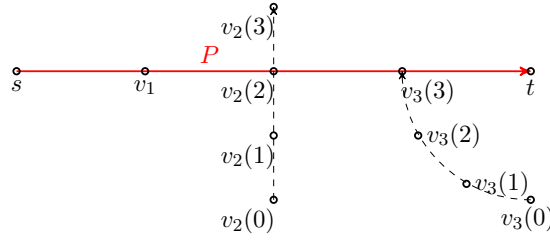


Figure 5.5: An example illustrating network model.

generic case where nodes are mobile. In this regard, only [91] has considered the TSP with moving nodes by designing a $(1 + \alpha)$ -approximation algorithm, where α denotes the approximation ratio of the TSP heuristic. However, the algorithm developed in [91] only works when the number of moving targets is sufficiently small. In contrast, we address the generic case even when the network scales. We would like to emphasize that the analysis method of the classic TSP and the (directed) Orienteering problems where the heuristic path is formed by joining several trees cannot be applied in our problem as the resulting path may not be feasible when nodes are mobile. Therefore, an original study is called for which cannot draw on existing results.

5.3.2 System Model and Problem Formulation

5.3.2.1 Network Model

We consider a network composed of n mobile nodes (e.g., robots, sensors, drones), denoted by the set $V \triangleq \{v_i\}_{i=1}^n$, deployed over a 2-D Euclidean plan. Nodes are battery-powered and thus need to be recharged periodically. To perform the charging task a mobile wireless charging vehicle, referred to as charger for short, travels from a starting point, denoted by a virtual node s , then visits a number of nodes to charge them before returning to a terminal point, denoted by t . If the charger needs to return to the starting point, t coincides with s , and the charging path becomes a tour. A node remains stationary when being charged.

The novel challenge we address, w.r.t. the state-of-the-art works, is that both the charger and the nodes are mobile. We denote $v_i(t)$ ($1 \leq i \leq n$) the position of node v_i at time t . We assume the mobility pattern of the nodes (i.e., their moving trajectories $v_i(t)$) are known to the charger and leave the unknown or partially known mobility case for future research. We denote r_i the upper-bound of the moving speed of v_i and r_s the moving speed of the charger. We assume that $r_s > r_i, \forall i \in [1, n]$ which holds in typical mobile sensing applications. We denote $P \in \mathcal{P}$ the charging path followed by the charger, where \mathcal{P} denotes the set of all possible paths. To make our analysis generic and widely applicable, we do not impose any constraint on the mobility patterns of nodes, i.e., the trajectory of any node can be a curve of any form. Table 6.2 lists the major notations used in this section. The example in Figure 5.5 further illustrates our network model.

Example 6. Consider the network illustrated in Figure 5.5 composed of three nodes v_1, v_2, v_3 . v_1 is stationary. v_2 and v_3 are mobile with their trajectories illustrated in the figure. Charging is immediate at any node. By following the path P , the charger can charge v_1, v_2 and v_3 at time 1, 2 and 3.

Table 5.2: Notations

Section 5.3.2: System Model and Problem Formulation	
V	Set of nodes in the network
n	Number of nodes in the network
v_i	Node i
$v_i(t)$	Position of node i at time t
s	Starting point of the charger
t	Terminal point of the charger
r_s	Moving speed of the charger
r_i	Upper-bound of moving speed of v_i
\mathcal{P}	Possible charging path set
$d(P)$	Euclidean length of path P
$t_s(p)$	Sojourn time of the charger at p without charging
$\Lambda(P)$	Set of nodes charged on path P
$\Gamma(P)$	Timespan of path P
$f_i(t)$	Charging level of node v_i as a function of charging time t
$g_i(x)$	Inverse function of $f_i(t)$: $g_i(x) = f_i^{-1}(t)$
α	Required charging level
B	Maximum charging path timespan
Section 5.3.3: Approximation Algorithm Design	
Δt	Time stepsize
G_d	Discretized graph with node set $V(G_d)$ and edge set $E(G_d)$
n_d	Number of nodes in G_d : $n_d = V(G_d) $
$c_d(e)$	Cost of edge e in G_d
λ	Scaling factor in edge cost rounding
$\epsilon(\lambda)$	Relative error under rounding factor λ
$C(v)$	Set of nodes in the same clique of v
V_{v-}	Set of nodes in G_d to which there exists an edge from v
V_{v+}	Set of nodes in G_d from which there exists an edge to v
L	Maximum recursion level in Algorithm 12

5.3.2.2 Optimum Charging Path Design

We consider a class of charging path optimization and scheduling problems including (but not limited to) the following:

- The charger has a fixed time budget for the charging journey and aims at charging the maximum number of nodes in one journey;
- The charger has a battery reservoir and aims at charging the maximum number of nodes before returning to its service station to replenish itself;
- The charger has a fixed number of $M \leq n$ nodes to charge and aims at minimizing the total charging time;
- The charger needs to charge all nodes within a charging journey and seeks the charging path minimizing the energy consumption or the total time.

The above problems can be classified into two categories:

- The charger has a certain budget (e.g., in terms of time, energy) and it seeks a path to maximize the number of nodes it can charge within the given budget;
- The charger has a number of nodes to charge and it seeks a path of minimum cost (e.g. in terms of time, energy consumption) to accomplish the charging task.

Our work establishes a generic framework on the charging path optimization and scheduling problems. To instantiate our work, we focus on the first problem of maximizing the number of nodes charged within a fixed time horizon, as formulated below. We discuss in Section 5.3.4 how our framework can be applied to address other problems formulated above.

5.3.2.3 Problem Formulation

We start by modeling the charging process. Let $x \in [0, 1]$ denote the battery level (in percentage) of a node during the charging process. For each node v_i , x can be expressed as a function of the charging time t and the initial battery level x_0 : $x = f_i(t, x_0)$. Figure 5.6 traces an example of charging curve for $x_0 = 0$. Throughout our analysis, we assume the battery level of any node remains constant during the charging journey unless it is charged by the charger. This assumption is reasonable as the duration of one charging journey is typically negligibly small compared to the lifetime of a node. This assumption also implies that each node is charged at most once during a charging journey. Under this assumption, f_i can be expressed as a function of t from the charger's perspective. Generically, the following property holds from elementary electrical circuit analysis:

- $f_i(t)$ is continuous, derivable and monotonously increasing in t ;
- $f_i(t)$ is concave in t , meaning that the marginal charging utility decreases in t .

Define $g_i(x) \triangleq f_i^{-1}(t)$. The following properties on $g_i(x)$ directly follow from the properties of $f_i(t)$:

- $g_i(x)$ exists and is the time required to charge v_i to x ;
- $g_i(x)$ is continuous and convex in x ;
- The derivative of $g_i(x)$, $g'_i(x)$, is increasing in x .

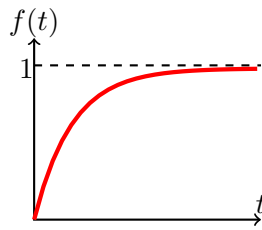


Figure 5.6: Example of charging curve.

Given a charging path P whose Euclidean length is denoted by $d(P)$, let $\Lambda(P) \subseteq P$ denote the set of nodes that the charger charges to battery level $\alpha \in [0, 1]$ (e.g., 90%) while traveling along P ³. For any point $p \in P$, let $t_s(p)$ denote the sojourn time during which s stays at p without charging

³To make our analysis clear, we assume nodes need to be charged to the same battery level α . The extension to the generic case with different charging levels is straightforward.

any node. The entire charging time along P , denoted by $\Gamma(P)$, can be established below

$$\Gamma(P) = \frac{d(P)}{r_s} + \sum_{v_i \in \Lambda(P)} g_i(\alpha) + \sum_{p \in P} t_s(p),$$

where $\frac{d(P)}{r_s}$ is the time required for the charger to travel distance $d(P)$, $\sum_{v_i \in \Lambda(P)} g_i(\alpha)$ is the time to charge all nodes in $\Lambda(P)$ to α and $\sum_{p \in P} t_s(p)$ is the total sojourn time.

We refer to the total charging time along a path P as the *timespan* of P . Without introducing ambiguity, a charging path P also denotes the corresponding charging schedule.

Problem 5.2 (Charging Path Optimization). *The charging path optimization problem is as follows:*

$$P^* = \operatorname{argmax}_{P \in \mathcal{P}, \Gamma(P) \leq B} |\Lambda(P)|.$$

That is, given the required charging level α and the maximum timespan B , the charger seeks an optimum path that maximizes the number of charged nodes. The solution P^ is termed as an optimum charging path.*

In many practical scenarios, it is acceptable to have small margin on the charging performance, which motivates the following definition.

Definition 5.5 (ϵ -optimum Charging Path). *A charging path P_ϵ is called an ϵ -optimum charging path ($0 \leq \epsilon \leq 1$) if the following conditions are satisfied:*

- $\Gamma(P_\epsilon) \leq B$, i.e., the timespan of P_ϵ does not exceed B ;
- By following P_ϵ , the charger can charge all the nodes in $\Lambda(P^*)$ to at least $(1 - \epsilon)\alpha$.

5.3.2.4 Problem Hardness

We prove the APX-hardness of Problem 5.2 below.

Theorem 5.6 (APX-hardness of Charging Path Optimization). *Problem 5.2 is APX-hard, or NP-hard to approximate within an arbitrarily small constant factor.*

The proof [51] consists of relating Problem 5.2 to the Orienteering problem which has been proved APX-hard. In fact, even the static version of Problem 5.2 is APX-hard. It becomes much more complex with nodes being mobile.

5.3.3 Approximation Algorithm Design

In this subsection, we design an approximation algorithm to solve Problem 5.2.

5.3.3.1 Problem Discretization

We divide time into time instances $\{t_k, k = 0, 1, \dots\}$ with stepsize $\Delta t \triangleq t_k - t_{k-1}$ ⁴. The trajectory of v_i can then be discretized into a vector $[v_i(t_0), v_i(t_1), \dots]$ where $v_i(t_k)$ is the position

⁴To make the analysis concise, we set the same stepsize for all the nodes. Nevertheless, our analysis can be slightly adapted to the case with different per-node stepsize Δt_i .

of v_i at time t_k . By the discretization above, we construct a directed acyclic graph, denoted by G_d , whose vertices and edges are specified as follows.

Vertices. The vertex set $V(G_d)$ includes:

- the discretized points on the trajectories of all nodes in V , i.e., $v_i(k_i\Delta t)$ ($0 \leq k_i \leq K_i$), $\forall v_i \in V$ ⁵,
- the starting point s which is denoted as $v_0(0)$,
- the terminal point t denoted as $v_{n+1}(K_{n+1}\Delta t)$ with $K_{n+1} = \lfloor \frac{B}{\Delta t} \rfloor$.

Edges. For any two points $v_i(k_i\Delta t)$ and $v_j(k_j\Delta t)$ with $v_i \neq v_j$, there exists an edge $\overrightarrow{v_i(k_i\Delta t)v_j(k_j\Delta t)}$ in G_d iff the inequality below holds

$$(k_j - k_i)\Delta t \geq g_i[(1 - \epsilon)\alpha] + \frac{|v_i(k_i\Delta t)v_j(k_j\Delta t)|}{r_s}, \quad (5.1)$$

where $g_i[(1 - \epsilon)\alpha]$ is the time to charge v_i to $(1 - \epsilon)\alpha$ and $g_0[(1 - \epsilon)\alpha] = g_{n+1}[(1 - \epsilon)\alpha] = 0$ since the starting point v_0 and the terminal point v_{n+1} do not need to be charged. $|v_i(k_i\Delta t)v_j(k_j\Delta t)| = d[v_i(k_i\Delta t), v_j(k_j\Delta t)]$ is the Euclidean distance between $v_i(k_i\Delta t)$ and $v_j(k_j\Delta t)$, i.e., the length of the line segment $\overrightarrow{v_i(k_i\Delta t)v_j(k_j\Delta t)}$. The inequality (5.1) implies that starting at $v_i(k_i\Delta t)$ at time $k_i\Delta t$, the charger can charge v_i to level $(1 - \epsilon)\alpha$ and then arrive at $v_j(k_j\Delta t)$ no later than $k_j\Delta t$. For any edge $\overrightarrow{v_i(k_i\Delta t)v_j(k_j\Delta t)}$, we set its cost $c_d[v_i(k_i\Delta t), v_j(k_j\Delta t)]$ as follows:

$$c_d[v_i(k_i\Delta t), v_j(k_j\Delta t)] \triangleq (k_j - k_i)\Delta t.$$

Note that there is no edge between $v_i(k_i^1\Delta t)$ and $v_i(k_i^2\Delta t)$ for any $v_i \in V$ and $0 \leq k_i^1, k_i^2 \leq K_i$.

The index upper-bound K_i ($1 \leq i \leq n$) is set as below:

$$\frac{c_d[v_0(0), v_i(K_i\Delta t)] + c_d[v_i(K_i\Delta t), v_{n+1}(K_{n+1}\Delta t)]}{r_s} + g_i[(1 - \epsilon)\alpha] \leq B, \quad (5.2)$$

where the left hand side of (5.2) is the minimum time required for the charger to go from the starting point to $v_i(K_i\Delta t)$ and charge v_i to $(1 - \epsilon)\alpha$ and then return to the terminal point. For any node $v_i(k_i\Delta t)$ with $k_i > K_i$, it is impossible for the charger to charge v_i at the position $v_i(k_i\Delta t)$ under the timespan budget B .

We next introduce two definitions that are useful in subsequent analysis.

Definition 5.6 (Clique). In G_d , a clique is defined as a set of nodes $v_i(k_i\Delta t)$ ($0 \leq k_i \leq K_i$) belonging to the trajectory of the same physical node v_i .

Definition 5.7 (Feasible Path). Let \mathcal{P}_d denote the set of all paths in G_d , a path $P \in \mathcal{P}_d$ is called a feasible path if P starts from $v_0(0)$, terminates at $v_{n+1}(K_{n+1}\Delta t)$, passes exactly one node of each clique it traverses, and $\Gamma(P) \leq B$. An optimum feasible path is a feasible path passing maximum number of cliques.

Example 7. Figure 5.7 illustrates an example of G_d and a feasible charging path P . There are two nodes to be charged with v_1 being stationary and v_2 being mobile. The coordinates of nodes and the trajectory of v_2 are shown in the left subfigure. $\Delta t = 1$. $B = 3$. The speed of the charger and v_2 are $r_s = 1$ unit length per unit time and $r_2 = 0.5$. Charging is immediate at any node. The corresponding discretized graph G_d is shown in the right subfigure with a feasible path P depicted in both subfigures.

⁵How to set K_i will be analysed later.

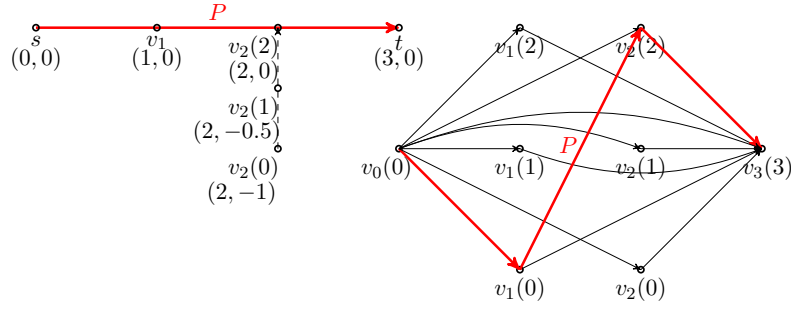


Figure 5.7: An example illustrating G_d and a feasible path P .

Theorem 5.7. *Any feasible path passing m cliques corresponds to a charging path of maximum timespan B that can charge m nodes to $(1 - \epsilon)\alpha$.*

We next show that an optimum charging path can be approximated arbitrarily close by a feasible path in terms of the number of charged nodes.

Theorem 5.8. *Given any $\epsilon > 0$, under the condition that $\Delta t \leq \frac{\alpha\epsilon}{3} \min_{1 \leq i \leq n} g'_i[(1 - \epsilon)\alpha]$, there exists a feasible path in G_d which is also an ϵ -optimum charging path.*

Theorem 5.8 demonstrates that the performance loss due to discretization can be controlled to arbitrarily small, at the price of increasing computation complexity in terms of the size of G_d . Given a tolerance level ϵ , Theorem 5.8 also quantifies the bound on the discretization granularity Δt to meet the performance requirement.

It follows from Theorem 5.7 and Theorem 5.8 that the problem of finding ϵ -optimum charging path can be transformed to the problem of finding the optimum feasible path, which is however APX-hard. The proof follows from the same deduction to the Orienteering problem as that in the proof of Theorem 5.6 [51]. Given its APX-hardness, we focus on approximation algorithm design in the next subsection.

5.3.3.2 Finding Optimum Feasible Path: Approximation Algorithm

This subsection presents our design of a quasi-polynomial time algorithm that achieves logarithmic approximation to the optimum feasible path. We first state the following property of G_d which is useful in later analysis.

Lemma 5.4. *For any pair of nodes $v_s, v_t \in V(G_d)$, if there exists a path from v_s to v_t , then there must exist an edge from v_s to v_t , i.e., $c(v_s, v_t) \neq \infty$. Equivalently, if $c(v_s, v_t) = \infty$, then there does not exist a path from v_s to v_t .*

To develop our algorithm, we assume that the edge costs of G_d are integers. If not, we can round them to integers by scaling each edge cost by a factor λ and rounding the scaled cost to its ceiling integer. The relative error incurred by the rounding process can be upper-bounded by λ , denoted by $\epsilon(\lambda)$, as follows:

$$\epsilon(\lambda) = \frac{\lceil c_d(e)\lambda \rceil - c_d(e)\lambda}{c_d(e)\lambda} \leq \frac{1}{c_d(e)\lambda} = O\left(\frac{1}{\lambda}\right).$$

The core idea of our algorithm, inspired by the idea of recursion in [36, 37], is summarized below:

- For each $m = [1..n_d]$ ($n_d \triangleq |V(G_d)|$) and each node $v \in V(G_d) - \{s, t\}$:
 - Recursively search a path P_1 from s to v of minimum timespan that charges m nodes, denote the timespan of P_1 by b_1 ;
 - Recursively search another path P_2 from v to t of timespan at most $B - b_1$ that charges the maximum number of nodes;
- Output the concatenated path $P = (P_1, P_2)$ that charges the maximum number of nodes;

In the recursion process, we need to carefully choose P_1 and P_2 such that the resulting concatenated path does not visit any clique more than once.

Formally, the pseudo-code of the algorithm is illustrated in Algorithm 12. The core part of Algorithm 12 is the recursive procedure OPF, which has the following inputs:

- G_d : the discretized graph;
- v_s, v_t : the starting and terminating nodes;
- V : the set of nodes to be charged; initially $V = V(G_d)$;
- l : the recursion level, upper-bounded by L ;
- b : the timespan budget of the charging journey.

OPF returns the optimum feasible path starting from v_s ending at v_t charging maximum number of nodes in V , whose timespan is upper-bounded by b , by invoking l recursions.

- OPF first checks the timespan budget b and returns $P = \emptyset$ if the budget is infeasible. OPF also returns \emptyset if $c_d(v_s, v_t) = \infty$ as it follows from Lemma 5.4 that $c_d(v_s, v_t) = \infty$ implies there does not exist a path from v_s to v_t . Otherwise P is initialized to $c_d(v_s, v_t)$.
- If $l = 0$, meaning that the current instance is the last recursion, then OPF returns P .
- Otherwise, OPF iterates on each node $v \in V(G_d) - C(v_s) - C(v_t)$ and $m = [1..n_d]$ to recursively find: (1) P_1 with minimum timespan (denoted by b_1) starting from v_s ending at v that charges m nodes, each in a distinct clique, and (2) P_2 which starts from v and ends at v_t with timespan $B - b_1$ that charges maximum number of nodes, each in a distinct clique and different to the cliques in P_1 . To find P_1 , it follows from Lemma 5.4 that only nodes in V_{v-} need to be searched. Symmetrically, only nodes in $V_{v+} - C(P_1)$ need to be searched to find P_2 .
- The output is the concatenation of P_1 and P_2 that charges the maximum number of nodes.

To find P_1 OPF calls the procedure BMIN, essentially a binary search function that returns the path with minimum timespan starting from v_s ending at v that charges m nodes, each in a distinct clique.

In the following three lemmas, we show that OPF indeed returns a feasible path (Lemma 5.5) and establish its time complexity (Lemma 5.6) and approximation ratio (Lemma 5.7). The core part of the proofs, especially that of Lemma 5.7, consists of decomposing P_f^* into two sub-paths charging $\lfloor \frac{m^*}{2} \rfloor$ and $\lceil \frac{m^*}{2} \rceil$ nodes and then proving the results by induction on L .

Lemma 5.5 (Correctness of OPF). *If $B \geq c_d(s, t)$, OPF returns a feasible path for any L , otherwise it returns $P = \emptyset$.*

Algorithm 12 Finding optimum feasible path (OFFP)**Input:** G_d, s, t, B, L **Output:** P_f

```

1: return  $P_f := \text{OPF}(G_d, s, t, V(G_d), L, B)$ 

2: procedure  $\text{OPF}(G_d, v_s, v_t, V, l, b)$ 
3:   if  $b < c_d(v_s, v_t)$  or  $c_d(v_s, v_t) = \infty$  then
4:     return  $P := \emptyset$ 
5:   else
6:      $P := (v_s, v_t)$ 
7:   end if
8:   if  $l = 0$  then
9:     return  $P$ 
10:  end if
11:  for all  $v \in V - C(v_s) - C(v_t)$  do
12:    for  $m := 1$  to  $n_d$  do
13:       $(P_1, b_1) := \text{BMIN}(G_d, v_s, v, V_{v-}, l - 1, b, m)$ 
14:      if  $P_1 = \emptyset$  then
15:        Break
16:      end if
17:       $P_2 := \text{OPF}(G_d, v, v_t, V_{v+} - C(P_1), l - 1, b - b_1)$ 
18:      if  $P_2 = \emptyset$  then
19:        Break
20:      end if
21:      if  $|\Lambda(P_1)| + |\Lambda(P_2)| > |\Lambda(P)|$  then
22:         $P := (P_1, P_2)$ 
23:      end if
24:    end for
25:  end for
26:  return  $P$ 
27: end procedure

28: procedure  $\text{BMIN}(G_d, v_s, v, V, l, b, m)$ 
29:    $b_{min} := 1, P_{max} := \text{OPF}(G_d, v_s, v, V, l, b)$ 
30:   if  $|\Lambda(P_{max})| < m$  then
31:     return  $(\emptyset, 0)$ 
32:   end if
33:    $b_{max} := b$ 
34:   while  $b_{max} - b_{min} > 1$  do
35:      $b_{mid} := \left\lceil \frac{b_{min} + b_{max}}{2} \right\rceil$ 
36:      $P := \text{OPF}(G_d, v_s, v, V, l, b_{mid})$ 
37:     if  $|\Lambda(P)| \leq m$  then
38:        $b_{min} := b_{mid}, P_{min} := P$ 
39:     else
40:        $b_{max} := b_{mid}, P_{max} := P$ 
41:     end if
42:   end while
43:   return  $(P, b_{mid})$ 
44: end procedure

```


Lemma 5.6 (Time complexity of OFP). *OFP terminates in $O\left((n_d \min(n_d, B) \log B)^L\right)$ time.*

Lemma 5.7 (Approximation ratio of OFP). *Let P_f^* denote the optimum feasible path and let $m^* = |\Lambda(P_f^*)|$. Under the condition $L \geq \lceil \log m^* \rceil + 1$ ⁶, it holds that*

$$|\Lambda(P)| \geq \frac{|\Lambda(P_f^*)|}{1 + \lceil \log m^* \rceil}, \text{ i.e., } |\Lambda(P)| = \Omega\left(\frac{|\Lambda(P_f^*)|}{\log m^*}\right).$$

Lemma 5.5, Lemma 5.6 and Lemma 5.7 together lead to the main theorem below on the performance of OFP.

Theorem 5.9. *By setting $L = 1 + \lceil \log m^* \rceil$, OFP finds an $O(\log m^*)$ -approximate optimum feasible path in quasi-polynomial time.*

In practice, to set L without knowing m^* , we need to estimate the upper-bound of m^* . Since the edge costs are integers, it holds that $m^* \leq B$. L can thus be set to B to ensure the $O(\log B)$ approximation.

5.3.4 Discussion on Variants and Extensions

We now discuss how our approximation algorithm can be adapted and extended to solve other charging path optimization and scheduling problems formulated in Section 5.3.2.

For the problem where the charger has a battery reservoir and aims at charging the maximum number of nodes before returning to its service station to replenish itself, we can set the edge cost between v_i and v_j ($v_i, v_j \in V(G_d)$) to the energy required to charge v_i to $(1 - \epsilon)\alpha$ plus the energy consumption to move from v_i to v_j . Our algorithm OFP can then be invoked to find the $O(\log m^*)$ -optimum solution. In a broader sense, our approach can be adopted to solve the first category of the charging path optimization and scheduling problems formulated in Section 5.3.2.

We next focus on the second class of charging path optimization problems where the charger has a number of nodes M to charge and it seeks a path with minimum cost (e.g. in terms of time, energy consumption) to accomplish the charging task. This class of problems are NP-hard because when nodes are stationary and charging is immediate and $M = n$, the problems degenerate to the classical TSP problem which is NP-hard. To solve these problems, we devise a recursive algorithm similar to OFP, summarized below:

- For each $m = [1..M]$ and each node $v \in V(G_d) - \{s, t\}$:
 - Recursively search a path P_1 from s to v of minimum timespan (or any form of budget) charging m nodes, denote the timespan of P_1 by b_1 ;
 - Recursively search another path P_2 from v to t of minimum timespan charging $M - m$ nodes;
- Output the concatenated path $P = (P_1, P_2)$ of minimum timespan;

⁶All logarithms in our analysis are to base 2. In the case where $m^* = 0$, meaning that the optimum feasible path cannot pass any node other than s and t , it holds that $|\Lambda(P_f^*)| = 0$, Lemma 5.7 holds trivially for any L .

Using the similar analysis as OFP, we can establish the logarithmic approximation ratio of the above algorithm.

In many practical scenarios, some nodes are more critical than others. Instead of seeking $P^* = \operatorname{argmin}_{P \in \mathcal{P}} |\Lambda(P)|$, it makes more sense to solve $P^* = \operatorname{argmin}_{P \in \mathcal{P}} \sum_{v_i \in \Lambda(P)} w_i$, where w_i is a weight for node v_i . Our algorithm OFP can be readily applied to solve such weighted version by attributing a reward w_i to v_i and by adjusting the objective to finding the path maximizing the collected reward.

Our algorithm can also be adapted to solve the charging path optimization with time windows where node v_i needs to be charged within a time window. Specifically, this can be done by redefining the feasible charging path such that only nodes charged within its time window is counted in $\Lambda(P)$.

5.4 Summary of Other Contributions

We have also considered the problem of realtime charging path optimization and scheduling as follows: given each recharge as a real-time task which is not known a priori, find the traveling path that allows the mobile robot to process these tasks so that the percentage of alive nodes is maximized. The main difficulty of this problem is that both deadlines and spatial distributions of nodes affect the selection of traveling path.

We first formulate the mobile charger scheduling problem, then prove its NP-completeness. The solution to this problem depends on the relative ratio between the traveling time of the charger and the battery lives of the sensor nodes. If the traveling time to traverse the network is much shorter than the battery lives of the sensor nodes, then no nodes require more than one recharge in each round. The scheduling problem is reduced to the TSP with Time Window, which was studied before. On the other hand, if the traveling time is larger than some or all of the sensor nodes' battery lives, this problem gets more complicated because each time a sensor node is charged, its deadline is reset. In this case, we aim at finding schedules responding to such dynamically generated deadlines, and maintain high percentage of alive nodes over time. To reveal the complexity of the problem, we investigate the performances of existing solutions, such as the Earliest Deadline First (EDF) and TSP types of algorithms. We show EDF scheduler rescues urgent nodes, but does not take spatial distribution of nodes into account. Whereas, TSP scheduler minimizes the traveling cost, but fails to respond to the most urgent node in time. Therefore, it is essential to consider spatial and temporal constraints together in this scenario.

To address this problem, we propose a spatial dependent task model to quantify the impact of scheduling proximate tasks on the other tasks. Intuitively, the traveling time for recharging nearby urgent nodes is less than faraway urgent nodes. Therefore, we can optimize the recharging schedule based on the proximity and density of nodes. With the spatial dependent task model, we design the Spatial Dependent Task scheduler. It first identifies clusters of nodes such that recharging them maximizes the network energy level, then computes the traveling path to reach these clusters, such that more nodes can be recharged without incurring much detour.

We have implemented a comprehensive simulation framework, and conducted extensive trace-driven experiments to test different scheduling algorithms under various settings. The simulation uses energy consumption data traces, which are collected from real sensor network deployments. Experimental results show that our SDT scheduler outperforms the classical shortest route scheduler (TSP) by up to 10% coverage ratio, and reduces the average tardiness by up to 85%.

5.5 Conclusion and Perspectives

In this chapter, we have studied a class of path optimization and the related scheduling problems arising from data harvesting and mobile charging. We have performed a generic analysis on these problems and designed polynomial or quasi-polynomial time algorithm achieving constant or logarithmic approximation to the optimum. Throughout this chapter, we have taken an analysis procedure from problem formulation and abstraction, to the analysis on the hardness of the formulated problem, and then to the design of approximation algorithms, completed by the discussion and extension to related and more generalized versions of the problems.

There are a number of research directions we plan to investigate following our work in this chapter. Take the charging path optimization as an example, the first direction is to study the case where the trajectories of nodes are partially known, or even unknown to the charger. In this context, the charger should make its decision based on only local information and interaction. Secondly, in some practical settings, mobile nodes can also adapt their trajectories to facilitate charging. A natural research question is to look for the optimum charging path by exploiting such adaptation, which may significantly improve charging performance. The third direction is to study more sophisticated variants of the problem, e.g., the case of multiple chargers with heterogeneous moving speed.

Chapter 6

Algorithm Design and Analysis in RFID Systems

6.1 Introduction

In this chapter, we focus on algorithm design and analysis in RFID (radio-frequency identification) systems, which are becoming ubiquitously available today in many domains ranging from warehouse management, object tracking to inventory control. Our focus is tag counting and monitoring, one of the most fundamental functionalities in RFID systems, particularly when the system scales. In this context, the major performance metric in the algorithm design is the time efficiency as an algorithm (e.g., tag population estimation, missing tag monitoring and detection) is likely to be executed frequently.

6.1.1 Stability Analysis of Frame Slotted Aloha Protocol

We start by studying the stability of the Frame Slotted Aloha (FSA) protocol. This is an important problem as FSA has been widely applied in RFID systems as the *de facto* standard in tag identification. However, very limited work has been done on the stability of FSA despite its fundamental importance both on the theoretical characterisation of FSA performance and its effective operation in practical systems. In order to bridge this gap, we devote our first analysis to investigating the stability properties of FSA by focusing on two physical layer models of practical importance, the models with single packet reception (SPR) and multipacket reception (MPR) capabilities.

Technically, we model the FSA system backlog as a Markov chain with its states being backlog size at the beginning of each frame. The objective is to analyze the ergodicity of the Markov chain and demonstrate its properties in different regions, particularly the instability region. By employing drift analysis, we obtain the closed-form conditions for the stability of FSA and show that the stability region is maximized when the frame length equals the backlog size in the SPR model and the upper bound of stability region is maximized when the backlog size equals the maximum multipacket reception capacity in the MPR model. Furthermore, to characterise system behavior in the instability region, we mathematically demonstrate the existence of transience of the backlog Markov chain.

6.1.2 Dynamic Tag Population Estimation Algorithm Design

We then proceed to study the problem of tag population estimation. Quickly and accurately estimating the number of tagged objects is crucial in establishing inventory reports for large retailers such as Wal-Mart [168]. Due to the paramount practical importance of tag population estimation, a large body of studies [110, 120, 160, 177, 239] have been devoted to the design of efficient estimation algorithms. Most of them, as reviewed in Sec. 6.3.1, are focused on the static scenario where the tag population is constant during the estimation process. However, many practical RFID applications, such as logistic control, are dynamic in the sense that tags may be activated or terminated as specialized in C1G2 standard [73], or the tagged objects may enter and/or leave the reader's covered area frequently, thus resulting in tag population variation. In such dynamic applications, a fundamental research question is how to design efficient algorithms to dynamically trace the tag population quickly and accurately.

We develop a generic framework of stable and accurate tag population estimation schemes for both static and dynamic RFID systems. By generic, we mean that our framework does not require any prior knowledge on the tag arrival and departure patterns. Our design is based on the extended Kalman filter (EKF) [180], a powerful tool in optimal estimation and system control. By performing Lyapunov drift analysis, we mathematically prove the efficiency and stability of our framework.

6.1.3 Missing Tag Detection Algorithm Design

We complete our work with a comprehensive analysis on the detection of missing tags, one of the most important RFID applications which has attracted extensive research attention (cf. Section 6.4.1 on related work on missing tag detection). We investigate a new problem motivated by the following practical settings.

- *Multiple groups of tags.* Tags are usually attached to objects belonging to different groups: e.g., different brands of the goods with the high-end brands order-of-magnitude more valuable than their low-end peers. Therefore, the missing tag events are characterized by asymmetrical threshold and reliability requirement across groups.
- *Multiple interrogation regions.* Tags may be unevenly located in multiple interrogation regions: e.g., tags may be located in several rooms or different corners or regions of a large warehouse. Hence, a reader may need to move several times to cover all monitored tags and complete the missing tag detection process.

The problem we consider is to devise missing tag detection algorithm with minimum execution time while guaranteeing the detection reliability requirement for each group of tags. We deliver a comprehensive analysis on the missing tag detection problem in the above multiple-group multiple-region environment and investigate how to devise optimum missing tag detection algorithms. Note that when there are only one group and all tags are with one interrogation region, our problem degenerates to the classical missing tag detection problem studied in the literature.

To design missing tag detection algorithms in the multiple-region multiple-group case, we leverage a powerful technique called *Bloom filter* which is a space-efficient probabilistic data structure for representing a set and supporting set membership queries [32] to detect a missing event. Specifically, we develop a suite of three missing tag detection algorithms, each decreasing the execution time compared to its predecessor by incorporating an improved version of the Bloom filter design and parameter tuning. By sequentially analysing the developed algorithms, we

gradually iron out an optimum detection algorithm that works in practice.

6.1.4 Chapter Organization

The rest of this chapter is structured as follows. Section 6.2 presents our work on the stability of FSA. Section 6.3 develops our work on tag population estimation. Section 6.4 focuses on our work on missing tag detection. The work of this chapter is in collaboration with my former Ph.D. student Jihong YU who has just defended his thesis and started his post-doc at Simon Fraser University in Canada. More details of our work on this topic including proofs and numerical analysis can be found in our publications [227–230].

6.2 Stability Analysis of Frame Slotted Aloha Protocol

6.2.1 Introduction

Since the introduction of Aloha protocol in 1970 [1], a variety of such protocols have been proposed to improve its performance, such as Slotted Aloha (SA) [167] and Frame Slotted Aloha (FSA) [154]. SA is a well-known random access scheme where the time is divided into identical slots of duration equal to the packet transmission time and the users contend to access the medium with a predefined access probability. As a variant of SA, FSA divides slots into *frames* and a user is allowed to transmit only a single packet per frame in a randomly chosen slot.

Due to their effectiveness to tackle collisions in wireless networks, SA- and FSA-based protocols have been applied extensively to various networked systems ranging from the satellite networks [153], wireless LANs [197, 232] to the emerging Machine-to-Machine (M2M) networks [198, 215]. Specifically, in RFID systems, FSA plays a fundamental role in the identification of tags [131, 241] and is standardized in the EPCGlobal Class-1 Generation-2 (C1G2) RFID standard [73]. In FSA-based protocols, all users with packets transmit in the selected slot of the frame respectively, but only packets experiencing no collisions are successful while the other packets are retransmitted in the subsequent frames.

Given the paramount importance of the stability for systems operating on top of Aloha-based protocols, a large body of studies have been devoted to stability analysis in a slotted collision channel [34, 74, 112] where a transmission is successful if and only if just a single user transmits in the selected slot, referred to as single packet reception (SPR). Differently with SPR, the emerging multipacket reception (MPR) technologies in wireless networks, such as Code Division Multiple Access (CDMA) and Multiple-Input and Multiple-Output (MIMO), make it possible to receive multiple packets in a slot simultaneously, which remarkably boosts system performance at the cost of the system complexity.

More recently, the application of FSA in RFID systems and M2M networks has received considerable research attention. However, very limited work has been done on the stability of FSA despite its fundamental importance both on the theoretical characterisation of FSA performance and its effective operation in practical systems. Motivated by the above observation, we argue that a systematic study on the stability properties of FSA incorporating the MPR capability is called for in order to lay the theoretical foundations for the design and optimization of FSA-based communication systems.

Motivated by the above analysis, we investigate the stability properties of FSA with SPR and MPR capabilities. The main contributions are articulated as follows. We model the packet transmission process in a frame as the bins and balls problem [104] and derive the number of successfully received packets under both SPR and MPR models. We formulate a homogeneous Markov chain to characterize the number of the backlogged packets and derive the one-step transition probability. By employing drift analysis, we obtain the closed-form conditions for the stability of FSA and derive conditions maximising the stability regions for both SPR and MPR models. To characterise system behavior in the instability region, we mathematically demonstrate the existence of transience of the backlog Markov chain.

6.2.2 Related Work

Aloha-based protocols are basic schemes for random medium access and are applied extensively in many communication systems. As a central property, the stability of Aloha protocols has received a lot of research attention, which we briefly review here.

Stability of slotted Aloha. Tsybakov and Mikhailov [195] initiated the stability analysis of finite-user slotted Aloha. They established sufficient conditions for stability of the queues in the system using the principle of stochastic dominance and derived the stability region for two users explicitly. For the case of more than two users, the inner bounds to the stability region were shown in [163]. Subsequently, Szpankowski [185] established necessary and sufficient (but not in closed-form) conditions for the stability under a fixed transmission probability vector for three-user case. In [34] an approximate stability region was derived for arbitrary number of users based on the mean-field asymptotics. The sufficient condition for the stability was further derived to be linear in arrival rates without the requirement on the knowledge of the stationary joint statistics of queue lengths in [112]. Recently, the stability region of SA with K-exponential backoff was derived in [74] by modeling the network as inter-related quasi-birth-death processes. We would like to point out that all the above stability analysis results were derived for the SPR model.

Stability of slotted Aloha with MPR. The first attempt of analyzing stability properties of SA with MPR was made by Ghez *et al.* in [81, 82] in an infinite-user single-buffer model. They demonstrated that the system could be stabilized under the symmetrical MPR model with a non-zero probability that all packets were transmitted successfully. Later, Sant and Sharma [171] studied a special case of the symmetrical MPR model for finite-user with an infinite buffer. They derived sufficient conditions on arrival rate for stability under the stationary ergodic arrival process. Subsequently, the effect of MPR on stability and delay was investigated in [150] and it was shown that stability region undergoes a phase transition and then reaches the maximization. More recently, Jeon and Ephremides [102] characterized the exact stability region of SA with stochastic energy harvesting and MPR for a pair of bursty users. These works are mostly, if not all, focused on the baseline SA, while our focus is FSA with both SPR and MPR.

Performance analysis of FSA. There exist several studies on the performance of FSA. Wieselthier and Anthony [213] introduced a combinational technique to analyse performance of FSA-MPR for the case of finite users. Schoute [175] investigated dynamic FSA and obtained the expected number of slots needed until the backlog becomes zero. Recently, the optimal frame setting for dynamic FSA was proved mathematically in [159] and [27]. However, these works did not address the stability of FSA, which is of fundamental importance.

In summary, only very limited work has been done on the stability of FSA despite its funda-

mental importance both on the theoretical characterisation of FSA performance and its effective operation in practical systems. In order to bridge this gap, we devote our work to investigating the stability properties of FSA under both SPR and MPR models.

6.2.3 System Model

6.2.3.1 Physical Layer and Random Access Model in FSA

We consider a system of infinite identical users operating on one frequency channel. In one slot, a node can complete a packet transmission. We investigate two physical layer models, SPR and MPR models.

- Under SPR, a packet suffers a collision if more than one packet is transmitted in the same slot.
- Under MPR, up to \bar{M} ($\bar{M} > 1$) concurrently transmitted packets can be received successfully with non-zero probabilities as specified by a stochastic matrix Ξ defined as follows:

$$\Xi \triangleq \begin{pmatrix} \hat{\xi}_{10} & \hat{\xi}_{11} & & & & & \mathbf{0} \\ \hat{\xi}_{20} & \hat{\xi}_{21} & \hat{\xi}_{22} & & & & \\ \vdots & \vdots & \vdots & \ddots & & & \\ \hat{\xi}_{x_0 0} & \hat{\xi}_{x_0 1} & \cdots & \cdots & \hat{\xi}_{x_0 x_0} & & \\ \vdots & \vdots & \vdots & \vdots & \ddots & & \\ \hat{\xi}_{\bar{M} 0} & \hat{\xi}_{\bar{M} 1} & \cdots & \cdots & \hat{\xi}_{\bar{M} \bar{M}} & & \\ \mathbf{1} & \mathbf{0} & \cdots & \cdots & \cdots & & \mathbf{0} \end{pmatrix} \quad (6.1)$$

where $\hat{\xi}_{x_0 k_0}$ ($k_0 \leq x_0 \leq \bar{M}$) is the probability of having k_0 successful packets among x_0 transmitted packets in one slot. Ξ is referred to as the reception matrix. The last two decades have witnessed an increasing prevalence of MPR technologies such as CDMA and MIMO. Mathematically, the SPR model can be regarded as a degenerated MPR model with $\bar{M} = 1$ and

$$\Xi = \begin{pmatrix} 0 & 1 & & \\ 1 & 0 & & \\ \vdots & \vdots & \mathbf{0} & \\ 1 & 0 & & \end{pmatrix}.$$

The random access process we consider is as follows: FSA organises time slots in frames, each containing a number of consecutive slots. Each user is allowed to randomly and independently choose a slot to send his packet at most once per frame. More specifically, Denote the length of frame t by L_t ; in the beginning of frame t each user generates a random number r and selects the $(r \bmod L_t)$ -th slot in frame t to transmit his packet. Note that unsuccessful packets in the current frame are retransmitted in the next frame with the constant persistence probability p while new generated packets are transmitted in the next frame following their arrivals with probability one.¹

¹To make our presentation concise, we focus on the case $p = 1$ in this chapter. The case with general p follows similar analysis and is detailed in [227].

For notation convenience, we use FSA-SPR and FSA-MPR to denote the FSA system operating on the SPR and MPR models, respectively.

6.2.3.2 Traffic Model

Let N_t denote the number of packets arrived during frame t and denote by A_{tl} the number of packets arrived in slot l of frame t where $l = 1, 2, \dots, L_t$. Assume that $\{A_{tl}\}$ are independent and identically Poisson distributed random variables with probability distribution:

$$P\{A_{tl} = u\} = \Lambda_u(u \geq 0) \quad (6.2)$$

such that the expected number of arrivals per slot $\Lambda = \sum_{u=1}^{\infty} u\Lambda_u$ is finite. Note that $N_t = \sum_{l=1}^{L_t} A_{tl}$, N_t also follows Poisson distribution with expectation $L_t\Lambda$.

6.2.4 Main Results

Aiming at studying the stability of FSA, we decompose our global objective into the following three questions:

- **Q1:** Under what condition(s) is FSA stable?
- **Q2:** When is the stability region maximized?
- **Q3:** How does FSA behave in the instability region?

Before answering the questions, we first introduce the formal definition of stability employed by Ghez *et al.* in [82]. Define by random variable X_t the backlog size in the system at the beginning of frame t . The discrete-time process $(X_t)_{t \geq 0}$ can be seen as a homogeneous Markov chain.

Definition 6.1. *An FSA system is stable if $(X_t)_{t \geq 0}$ is ergodic and unstable otherwise.*

By Definition 6.1, we can transform the study of stability of FSA into investigating the ergodicity of the backlog Markov chain. The rationale of this transformation is two-fold. The first interpretation is the property of ergodicity: there exists a unique stationary distribution of a Markov chain if it is ergodic. The second follows the nature of ergodicity: each state of the Markov chain can recur in finite time with probability 1 asymptotically. From an engineering perspective, if an FSA system is stable, then asymptotically the backlog size will not explode.

We next establish the following results characterizing the stability region and demonstrating the behavior of the Markov chain in non-ergodic regions under both SPR and MPR. The proof and analysis are detailed in [227], where the core technique used is drift analysis with the drift defined as $D_i = E[X_{t+1} - X_t | X_t = i]$.

6.2.4.1 Results for FSA-SPR

Let \hat{h} denote the backlog size at the beginning of the frame t and define $\alpha \triangleq \frac{\hat{h}}{L_t}$, where L_t is the size of frame t , we have the following theorem.

Theorem 6.1 (Stability of FSA-SPR). *Under FSA-SPR, the following results hold asymptotically.*

1. The system is stable if $\Lambda < \alpha e^{-\alpha}$ and $L_t = \Theta(\hat{h})$. Specially, $\alpha = 1$ maximizes the stability region² and the stable throughput.
2. The system is unstable under each of the following three conditions: (1) $L_t = o(\hat{h})$; (2) $\hat{h} = o(L_t)$; (3) $L_t = \Theta(\hat{h})$ and $\Lambda > \alpha e^{-\alpha}$.

Remark. Theorem 6.1 answers the first two questions and can be interpreted as follows:

- When $L_t = o(\hat{h})$, the number of packets sent during frame t is far larger than the frame length; a packet experiences collision w.h.p., thus increasing the backlog size and destabilising the system.
- When $\hat{h} = o(L_t)$, the number of packets sent is far smaller than the frame length; a packet is transmitted successfully w.h.p.; however, the expected number of successful packets, which is $o(L_t)$, is still order-of-magnitude less than that of new arrivals in the frame, which is $O(L_t)$ asymptotically; the system is thus unstable.
- When $L_t = \Theta(\hat{h})$, the number of packets sent is of the same order as the frame length; the system is stable when the expected arrival rate is less than the effective throughput.

It is well known that an irreducible aperiodic Markov chain falls into one of three mutually exclusive classes: positive recurrent, null recurrent and transient. So, our next step after deriving the stability conditions is to show whether the backlog Markov chain in the instability region is transient or recurrent, which answers the third question.

Theorem 6.2 (Behavior of FSA-SPR in instability region). $(X_t)_{t \geq 0}$ is always transient in the instability region under each of the following three conditions: (1) $L_t = o(\hat{h})$; (2) $L_t = \Theta(\hat{h})$ and $\Lambda > \alpha e^{-\alpha}$; (3) $\hat{h} = o(L_t)$.

Remark. If a state of a Markov chain is transient, then the probability of returning to itself for the first time in a finite time is less than 1 asymptotically. Hence, Theorem 6.2 implies that once out of the stability region, the system is not guaranteed to return to stable state in finite time, that is, the number of backlogs may increase persistently.

6.2.4.2 Results for FSA-MPR

Theorem 6.3 (Stability of FSA-MPR). Under FSA-MPR, the following results hold asymptotically.

1. The system is always stable if $L_t = \Theta(\hat{h})$ and $\Lambda < \sum_{x_0=1}^{\bar{M}} e^{-\alpha} \frac{\alpha^{x_0}}{x_0!} \sum_{k_0=1}^{x_0} k_0 \hat{\xi}_{x_0 k_0}$. Specially, let α^* denote the value of α that maximizes the upper bound of stability region, it holds that $\alpha^* = \Theta(\bar{M})$.
2. The system is unstable under each of the following conditions: (1) $L_t = o(\hat{h}^{1-\epsilon_1})$ for any $0 < \epsilon_1 \leq 1$; (2) $L_t = O(\hat{h})$; (3) $\Lambda > \alpha$ and $L_t = \Theta(\hat{h})$.

Comparing Theorem 6.3 to Theorem 6.1, we can quantify the performance gap between FSA-SPR and FSA-MPR in terms of stability. For example, when $\alpha = 1$, the stability region is maximized in FSA-SPR with $\Lambda < e^{-1}$, while the upper bound of the stability region in FSA-MPR is

²The ergodicity region of a Markov chain is referred to as stability region.

$e^{-1} \sum_{x_0=1}^{\bar{M}} \frac{1}{(x_0-1)!}$. Note that for $\bar{M} > 2$, it holds that

$$1 + 1 + \frac{1}{2} < \sum_{x_0=1}^{\bar{M}} \frac{1}{(x_0-1)!} < 1 + 1 + \sum_{x_0=1}^{\bar{M}} \frac{1}{x_0(x_0+1)} < 2 + \left(\sum_{x_0=1}^{\bar{M}} \frac{1}{x_0} - \frac{1}{x_0+1} \right) = 3 - \frac{1}{\bar{M}+1}.$$

The upper bound of the stability region of FSA-MPR when $\alpha = 1$ is thus between 2.5 and 3 times the maximum stability region of FSA-SPR. Hence the maximum upper bound of the stability region of FSA-MPR achieved when $\alpha^* = \Theta(\bar{M})$ is far larger than that of FSA-SPR.

Theorem 6.4 (Behavior of FSA-MPR in instability region). *With the same notations as in Theorem 6.3, $(X_t)_{t \geq 0}$ is transient under each of the following three conditions: (1) $L_t = o(\hat{h}^{1-\epsilon_1})$; (2) $\hat{h} = o(L_t)$; (3) $\Lambda > \alpha$ and $L_t = \Theta(\hat{h})$.*

Theorem 6.4 demonstrates that despite the gain on the stability region size of FSA-MPR over FSA-SPR, their behavior in the unstable region are essentially identical.

6.3 Tag Population Estimation Algorithm Design

6.3.1 Background and Related Work

Due to its fundamental importance, tag population estimation has received significant research attention, which we briefly review here.

6.3.1.1 Tag Population Estimation for Static RFID Systems

Most of existing works are focused on the static scenario where the tag population is constant during the estimation process. The central question there is to design efficient algorithms quickly and accurately estimating the static tag population. Kodialam *et al.* designed an estimator called PZE which uses the probabilistic properties of empty and collision slots to estimate the tag population size [109]. The authors then enhanced PZE by taking the average of the probability of idle slots in multiple frames as an estimator in order to eliminate the constant additive bias [110]. Han *et al.* exploited the average number of idle slots before the first non-empty slots to estimate the tag population size [88]. Later, Qian *et al.* developed the Lottery-Frame scheme by employing geometrically distributed hash function such that the j th slot is chosen with probability $\frac{1}{2^{j+1}}$ [160]. As a result, the first idle slot approaches around the logarithm of the tag population and the frame size can be reduced to the logarithm of the tag population, thus reducing the estimation time. Subsequently, a new estimation scheme called ART was proposed in [177] based on the average length of consecutive non-empty slots. The design rationale of ART is that the average length of consecutive non-empty slots is correlated to the tag population. ART admits smaller variance than prior schemes. More recently, Zheng *et al.* proposed another estimation algorithm, ZOE, where each frame just has a single slot and the random variable indicating whether a slot is idle follows Bernoulli distribution [239]. The average of multiple individual observations is used to estimate the tag population.

We note that the above research works do not consider the estimation problem for dynamic RFID systems and thus may fail to monitor the system dynamics in real time. Specifically, in typical

static tag population estimation schemes, the final estimation result is the average of the outputs of multi-round executions. When applied to dynamic tag population estimation, additional estimation error occurs due to the variation of the tag population size during the estimation process.

6.3.1.2 Tag Population Estimation for Dynamic RFID Systems

Only a few propositions have tackled the dynamic scenario. The works in [172] and [219] considered specific tag mobility patterns with tags moving along the conveyor with constant speed. Xiao *et al.* developed an estimation algorithm, ZDE, in dynamic RFID systems to estimate the number of arriving and removed tags [216]. More recently, they further generalized ZDE by taking into account the snapshots of variable frame sizes [217]. Though the algorithms in [216] and [217] can monitor the dynamic RFID systems, they may fail to estimate the tag population size accurately, because they use the same hash seed in the whole monitoring process. Using the same seed is required in tracing tag departure and arrival. However, it may significantly limit the estimation accuracy, even in the static case.

Besides the limitations above, prior works do not provide formal analysis on the stability and the convergence rate. Motivated by the above argument, we develop a generic framework for tag population estimation in dynamic RFID systems. By generic, we mean that our framework can estimate the number of tags accurately without any prior knowledge on the tag arrival and departure patterns. Moreover, the efficiency and stability of our framework is mathematically established.

6.3.2 Technical Preliminaries

We briefly introduce the extended Kalman filter and some fundamental concepts and results in stochastic process which are useful in the subsequent analysis. The main notations used are listed in Table 6.1.

6.3.2.1 Extended Kalman Filter

The extended Kalman filter is a powerful tool to estimate system state in nonlinear discrete-time systems. Formally, a nonlinear discrete-time system can be described as follows:

$$z_{k+1} = f(z_k, x_k) + w_k^* \quad (6.3)$$

$$y_k = h(z_k) + u_k^* \quad (6.4)$$

where $z_{k+1} \in \mathbb{R}^n$ denotes the state of the system, $x_k \in \mathbb{R}^d$ is the controlled inputs and $y_k \in \mathbb{R}^m$ stands for the measurement observed from the system. The uncorrelated stochastic variables $w_k^* \in \mathbb{R}^n$ and $u_k^* \in \mathbb{R}^m$ denote the process noise and the measurement noise, respectively. The functions f and h are assumed to be the continuously differentiable.

For the above system, we introduce an EKF-based state estimator in Definition 6.2.

Definition 6.2 (Extended Kalman filter [180]). *A two-step discrete-time extended Kalman filter consists of state prediction and measurement update, defined as follows:*

Table 6.1: Main Notations

z_k	System state in frame k : tag population
y_k	Measurement in frame k : idle slot frequency
$\hat{z}_{k+1 k}$	Priori prediction of z_{k+1}
$\hat{z}_{k k}$	Posteriori estimate of z_k
$P_{k+1 k}$	Priori pseudo estimate covariance
$P_{k k}$	Posteriori pseudo estimate covariance
v_k	Measurement residual in frame k
K_k	Kalman gain in frame k
Q_k, R_k	Two tunable parameters in frame k
$e_{k k-1}$	Estimation error in frame k
Φ_k	Normalization of v_k
L_k	The length of frame k
$Rs_k, h(\cdot)$	Random seed in frame k and Hash function
r_k	Persistence probability in frame k
N_k	The number of idle slots in frame k
$p(z_k)$	Probability of an idle slot in frame k
$u_k, Var[u_k]$	Gaussian random variable and Variance of u_k
ϕ_k	Controllable parameter
w_k	Random variable: variation of tag population
θ, Υ_k	CUSUM threshold and reference value
ϵ	Upper bound of initial estimation error
λ_k, δ_k	Upper bounds of $E[w_k]$ and $E[w_k^2]$

1) Time update (prediction)

$$\hat{z}_{k+1|k} = f(\hat{z}_{k|k}, x_k) \quad (6.5)$$

$$P_{k+1|k} = P_{k|k} + Q_k, \quad (6.6)$$

2) Measurement update (correction)

$$\hat{z}_{k+1|k+1} = f(\hat{z}_{k+1|k}, x_k) + K_{k+1}v_{k+1} \quad (6.7)$$

$$P_{k+1|k+1} = P_{k+1|k} (1 - K_{k+1}C_{k+1}) \quad (6.8)$$

$$K_{k+1} = \frac{P_{k+1|k}C_{k+1}}{P_{k+1|k}C_{k+1}^2 + R_{k+1}}, \quad (6.9)$$

where

$$v_{k+1} = y_{k+1} - h(\hat{z}_{k+1|k}) \quad (6.10)$$

$$C_{k+1} = \left. \frac{\partial h(z_{k+1})}{\partial z_{k+1}} \right|_{z_{k+1}=\hat{z}_{k+1|k}}. \quad (6.11)$$

Remark. In the above definition of extended Kalman filter, the parameters can be interpreted in our context as follows:

- $\hat{z}_{k+1|k}$ is the prediction of z_{k+1} at the beginning of frame $k + 1$ given by the previous state estimate, while $\hat{z}_{k+1|k+1}$ is the estimate of z_{k+1} after the adjustment based on the measure at the end of frame $k + 1$.
- v_{k+1} , called innovation, is the measurement residual in frame $k+1$. It represents the estimated error of the measure.
- K_{k+1} is the Kalman gain. In (6.7), it weighs the innovation v_{k+1} w.r.t. $f(\hat{z}_{k+1|k}, x_k)$.
- $P_{k+1|k}$ and $P_{k+1|k+1}$, in contrast to the linear case, are not equal to the covariance of estimation error of the system state. Here, we call them pseudo-covariance.
- Q_k and R_k are two tunable parameters which play the role as that of the covariance of the process and measurement noises in linear stochastic systems to achieve optimal filtering in the maximum likelihood sense. They also play an important role in improving the stability and convergence of our EKF-based estimators.

6.3.2.2 Boundedness of Stochastic Process

In order to analyse the stability of an estimation algorithm, we need to check the boundedness of the estimation error defined as follows:

$$e_{k|k-1} \triangleq z_k - \hat{z}_{k|k-1}. \quad (6.12)$$

We further introduce the following two mathematical definitions [147] [188] on the boundedness of stochastic process.

Definition 6.3 (Boundedness of Random Variable). *The stochastic process of the estimation error $e_{k|k-1}$ is said to be bounded w.p.o., if there exists $X > 0$ such that*

$$\lim_{k \rightarrow \infty} \sup_{k \geq 1} \mathbb{P}\{|e_{k|k-1}| > X\} = 0. \quad (6.13)$$

Definition 6.4 (Boundedness in Mean Square). *The stochastic process $e_{k|k-1}$ is said to be exponentially bounded in the mean square with exponent ζ , if there exist real numbers $\psi_1, \psi_2 > 0$ and $0 < \zeta < 1$ such that*

$$E[e_{k|k-1}^2] \leq \psi_1 e_{1|0}^2 \zeta^{k-1} + \psi_2. \quad (6.14)$$

To investigate the boundedness defined in Definition 6.3 and 6.4, we present the following lemma [165].

Lemma 6.1. *Given a stochastic process $V_k(e_{k|k-1})$ and real numbers $\underline{\beta}, \bar{\beta}, \tau > 0$ and $0 < \alpha \leq 1$ with the following properties:*

$$\underline{\beta} e_{k|k-1}^2 \leq V_k(e_{k|k-1}) \leq \bar{\beta} e_{k|k-1}^2, \quad (6.15)$$

$$E[V_{k+1}(e_{k+1|k})|e_{k|k-1}] - V_k(e_{k|k-1}) \leq -\alpha V_k(e_{k|k-1}) + \tau, \quad (6.16)$$

then for any $k \geq 1$ it holds that

- the stochastic process $e_{k|k-1}$ is exponentially bounded in the mean square, i.e.,

$$E[e_{k|k-1}^2] \leq \frac{\bar{\beta}}{\underline{\beta}} E[e_{1|0}^2] (1 - \alpha)^{k-1} + \frac{\tau}{\underline{\beta}} \sum_{j=1}^{k-2} (1 - \alpha)^j \leq \frac{\bar{\beta}}{\underline{\beta}} E[e_{1|0}^2] (1 - \alpha)^{k-1} + \frac{\tau}{\underline{\beta} \alpha}, \quad (6.17)$$

- the stochastic process $e_{k|k-1}$ is bounded w.p.o..

From (6.15) and (6.16), if we can construct a function $V_k(e_{k|k-1})$ such that both its drift and $\frac{V_k(e_{k|k-1})}{e_{k|k-1}^2}$ are bounded, then $e_{k|k-1}$ is also bounded. Besides, it can be noted that Lemma 6.1 can only be implemented offline. To address this limit, we adjust Lemma 6.1 to an online version with time-varying parameters, which can be proven by the same method as in [166, 188].

Lemma 6.2. *If there exist a stochastic process $V_k(e_{k|k-1})$ and parameters $\beta^*, \beta_k, \tau_k > 0$ and $0 < \alpha_k^* \leq 1$ with the following properties:*

$$V_1(e_{1|0}) \leq \beta^* e_{1|0}^2, \quad (6.18)$$

$$\beta_k e_{k|k-1}^2 \leq V_k(e_{k|k-1}), \quad (6.19)$$

$$E[V_{k+1}(e_{k+1|k})|e_{k|k-1}] - V_k(e_{k|k-1}) \leq -\alpha_k^* V_k(e_{k|k-1}) + \tau_k; \quad (6.20)$$

then for any $k \geq 1$ it holds that

- the stochastic process $e_{k|k-1}$ is exponentially bounded in the mean square, i.e.,

$$E[e_{k|k-1}^2] \leq \frac{\beta^*}{\beta_k} E[e_{1|0}^2] \prod_{i=1}^{k-1} (1 - \alpha_i^*) + \frac{1}{\beta_k} \sum_{i=1}^{k-2} \tau_{k-i-1} \prod_{j=1}^i (1 - \alpha_{k-j}^*), \quad (6.21)$$

- the stochastic process $e_{k|k-1}$ is bounded w.p.o..

Remark. *The conditions in Lemma 6.2 can be interpreted as follows: to establish the boundedness of $e_{k|k-1}$, it suffices to construct a function $V_k(e_{k|k-1})$ such that both its drift, i.e., (6.20), and $\frac{V_k(e_{k|k-1})}{e_{k|k-1}^2}$, i.e., (6.18), (6.19), are bounded.*

6.3.3 System Model and Problem Formulation

6.3.3.1 System Model

Consider a RFID system consisting of a reader and a mass of tags operating on one frequency channel. The number of tags is unknown *a priori* and can be constant or dynamic (time-varying), which we refer to as *static* and *dynamic* systems, respectively. The MAC protocol for the RFID system is the standard FSA protocol analyzed in 6.2, where the standard *Listen-before-Talk* mechanism is employed by the tags to respond the reader's interrogation [76].

Specifically, the reader initiates a series of frames indexed by $k \in \mathbb{Z}_+$. Each frame, referred to as a round, consists of a number of slots. The reader starts frame k by broadcasting a begin-round command with frame size L_k , persistence probability r_k and a random seed Rs_k . When a tag receives a begin-round command, it uses a hash function $h(\cdot)$, L_k , Rs_k , and its ID to generate a uniformly distributed random number $i \in [0, L_k - 1]$ and reply in slot i of frame k with probability r_k .

Since every tag picks its own response slot independently, there may be zero, one, or more than one tags transmitting in a slot, which are referred to as *idle*, *singleton*, and *collision* slots, respectively. The reader is not assumed to be able to distinguish between a singleton or a collision slot, but it can detect an idle slot. We term both singleton and collision slots as *occupied* slots.

By collecting all responses in a frame, the reader can generate a binary sequence B_k , where ‘0’ indicates an idle slot, and ‘1’ stands for an occupied one.

The reader then terminates the current frame by sending an end round command. Based on the number of idle slots, i.e., the number of ‘0’ in B_k , the reader runs the estimation algorithm, detailed in following analysis, to estimate and trace the tag population.

6.3.3.2 Tag Population Estimation Problem

Our objective is to design a stable and accurate tag population estimation algorithm for both static and dynamic systems. By stable and accurate we mean that

- the estimation error of our algorithm is bounded in mean square in the sense of Definition 6.3 and 6.4 and the relative estimation error tends to zero;
- the estimated population size converges exponentially to the real value.

Mathematically, we consider a large-scale RFID system of a reader and a set of tags with the unknown size z_k in frame k which can be static or dynamic. Denote by $\hat{z}_{k|k-1}$ the prior estimate of z_k in the beginning of frame k . At the end of frame k , the reader updates the estimate $\hat{z}_{k|k-1}$ to $\hat{z}_{k|k}$ by running the estimation algorithm. Our designed estimation scheme need to guarantee the following properties:

- $\lim_{z_k \rightarrow \infty} \left| \frac{\hat{z}_{k|k-1} - z_k}{z_k} \right| = 0$;
- the converges rate is exponential.

6.3.4 Tag Population Estimation: Static Systems

We begin with the baseline scenario of static systems where the tag population is constant during the estimation process. We first establish the discrete-time model for the system dynamics and the measurement model using the bit string B_k observed during frame k . We then present our EKF-based estimation algorithm.

6.3.4.1 System Dynamics and Measurement Model

Consider the static RFID systems where the tag population stays constant, the system state evolves as

$$z_{k+1} = z_k, \quad (6.22)$$

meaning that the number of tags z_{k+1} in the system in frame $k + 1$ equals that in frame k .

In order to estimate z_k , we leverage the measurement on the number of idle slots during a frame. To start, we study the stochastic characteristics of the number of idle slots.

Assume that the initial tag population z_0 falls in the interval $z_0 \in [z_0, \bar{z}_0]$, yet the exact value of z_0 is unknown and should be estimated. The range $[z_0, \bar{z}_0]$ can be a very coarse estimation that can be obtained by any existing population estimation method. Recall the system model that in frame k , the reader probes the tags with the frame size L_k . Denote by variable N_k the number of idle slots in frame k , that is, the number of ‘0’s in B_k , we have the following results on N_k according to [109, 111].

Lemma 6.3. *If each tag replies in a random slot among the L_k slots with probability r_k , then it holds that $N_k \sim \mathcal{N}[\mu, \sigma^2]$ for large L_k and z_k , where*

$$\mu = L_k \left(1 - \frac{r_k}{L_k}\right)^{z_k}, \quad \sigma^2 = L_k(L_k - 1) \left(1 - \frac{2r_k}{L_k}\right)^{z_k} + L_k \left(1 - \frac{r_k}{L_k}\right)^{z_k} - L_k^2 \left(1 - \frac{r_k}{L_k}\right)^{2z_k}.$$

Lemma 6.4. *For any $\epsilon^* > 0$, there exists some $M > 0$, such that if $z_k \geq M$ or $L_k = \hat{z}_{k|k-1} \geq M$, then it holds that*

$$|\mu - L_k e^{-r_k \rho}| \leq \epsilon^*, \quad (6.23)$$

$$|\sigma^2 - L_k(e^{-r_k \rho} - (1 + r_k^2 \rho)e^{-2r_k \rho})| \leq \epsilon^*, \quad (6.24)$$

where $\rho = \frac{z_k}{L_k}$ is referred to as the reader load factor.

Lemmas 6.3 and 6.4 imply that in large-scale RFID systems, we can use $L_k e^{-r_k \rho}$ and $L_k(e^{-r_k \rho} - (1 + r_k^2 \rho)e^{-2r_k \rho})$ to approximate μ and σ^2 .

At the end of each frame k , the reader gets a measure y_k of the idle slot frequency defined as

$$y_k = \frac{N_k}{L_k}. \quad (6.25)$$

Recall Lemma 6.3, it holds that y_k is a Normal distributed random variable specified as follows: $E[y_k] = e^{-r_k \rho}$ and $Var[y_k] = \frac{1}{L_k}(e^{-r_k \rho} - (1 + r_k^2 \rho)e^{-2r_k \rho})$. Since there are z_k tags reply in frame k with probability r_k , the probability that a slot is idle, denoted as $p(z_k)$, can be calculated as

$$p(z_k) = \left(1 - \frac{r_k}{L_k}\right)^{z_k} \approx e^{-\frac{r_k z_k}{L_k}}. \quad (6.26)$$

Notice that for large z_k , $p(z_k)$ can be regarded as a continuously differentiable function of z_k .

Using the notation in the Kalman filter, we can write y_k as follows:

$$y_k = p(z_k) + u_k, \quad (6.27)$$

where, based on the statistic characteristics of y_k , u_k is a Gaussian random variable with zero mean and variance

$$Var[u_k] = \frac{1}{L_k}(e^{-r_k \rho} - (1 + r_k^2 \rho)e^{-2r_k \rho}). \quad (6.28)$$

We note that u_k measures the uncertainty of y_k .

To summarize, the discrete-time model for static RFID systems is characterized by (6.22) and (6.27). We conclude this subsection by stating the following auxiliary lemma which is useful in our later analysis.

Lemma 6.5. *Denote the function*

$$\Lambda(r_k) \triangleq Var[u_k] = \frac{1}{L_k}(e^{-r_k \rho} - (1 + r_k^2 \rho)e^{-2r_k \rho}), \quad \rho > 0,$$

it holds that $\Lambda(r_k)$ has a unique minimizer $r_k^ = 1.59/\rho$ where $\rho \geq 1.59$, and $\Lambda(r_k^*) \leq \frac{e^{1.59} - 1}{e^{3.18} L_k}$.*

Note that the estimation algorithm with the small variance $Var[u_k]$ is more accurate, thus we set the persistence probability r_k following Lemma 6.5.

6.3.4.2 Tag Population Estimation Algorithm

Noticing that the system state characterized by (6.22) and (6.27) is a discrete-time nonlinear system, we thus leverage the two-step EKF described in Definition 6.2 to estimate the system state. In (6.9), the Kalman gain K_k increases with Q_k while decreases with R_k . As a result, Q_k and R_k can be used to tune the EKF such that increasing Q_k and/or decreasing R_k accelerates the convergence rate but leads to larger estimation error. In our design, we set Q_k to a constant $q > 0$ and introduce a parameter ϕ_k as follows to replace R_k to facilitate our demonstration:

$$R_k = \phi_k P_{k|k-1} C_k^2. \quad (6.29)$$

It can be noted from (6.9) and (6.29) that K_k is monotonously decreasing in ϕ_k , i.e., a small ϕ_k leads to quick convergence at the price of relatively high estimation error. Hence, choosing the appropriate value for ϕ_k consists of striking a balance between the convergence rate and the estimation error. In our work, we take a dynamic approach by setting ϕ_k to a small value $\underline{\phi}$ at the first few rounds (J rounds) of estimation to allow the system to act quickly since the estimation in the beginning phase can be very coarse. After that we set ϕ_k to a relatively high value $\bar{\phi}$ to achieve high estimation accuracy.

Algorithm 13 Tag population estimation (static case): executed by the reader

Input: $z_0, P_{0|0}, q, J, L, \underline{\phi}, \bar{\phi}$, maximum number of rounds k_{max}

Output: Estimated tag population set $S_z = \{\hat{z}_{k|k} : k \in [0, k_{max}]\}$

- 1: **Initialisation:** $\hat{z}_{0|0} \leftarrow z_0, Q_0 \leftarrow q, S_z = \{\hat{z}_{0|0}\}$
 - 2: **for** $k = 1$ to k_{max} **do**
 - 3: $\hat{z}_{k|k-1} \leftarrow \hat{z}_{k-1|k-1}, L_k \leftarrow L, r_k \leftarrow 1.59L_k/\hat{z}_{k|k-1}, P_{k|k-1} \leftarrow P_{k-1|k-1} + Q_{k-1}$
 - 4: Generate random seed RS_k , broadcast (L_k, r_k, RS_k) , and run *Listen-before-Talk* protocol
 - 5: Obtain the number of idle slots N_k , and compute y_k and v_k using (6.25) and (6.10)
 - 6: $Q_k \leftarrow q$
 - 7: **if** $k \leq J$ **then**
 - 8: $\phi_k \leftarrow \underline{\phi}$
 - 9: **else**
 - 10: $\phi_k \leftarrow \bar{\phi}$
 - 11: **end if**
 - 12: Calculate R_k and K_k using (6.29) and (6.9)
 - 13: Update $\hat{z}_{k|k}$ and $P_{k|k}$ using (6.7) and (6.8)
 - 14: $S_z \leftarrow S_z \cup \{\hat{z}_{k|k}\}$
 - 15: **end for**
-

Now, we are ready to present our tag population estimation algorithm as illustrated in Algorithm 13. The major procedures of our estimation algorithm can be summarized as:

1. *In the beginning of frame k : prediction (line 3).* The reader predicts the number of tags based on the estimation at the end of frame $k - 1$. The predicted value is defined as $\hat{z}_{k|k-1}$. Then the reader sets the persistence probability r_k following Lemma 6.5 where z_k is set to $\hat{z}_{k|k-1}$.
2. *Line 4-5.* The reader launches the *Listen-before-talk* protocol as introduced in 6.3.3.1 in order to receive the feedbacks from tags.

3. *At the end of frame k : correction (line 6-14).* The reader computes N_k based on B_k and further calculates y_k and v_k from N_k . It then updates the prediction with the corrected estimate $\hat{z}_{k|k}$ following (6.7).

We will theoretically establish the stability and accuracy of the estimation algorithm in Sec. 6.3.6.

6.3.5 Tag Population Estimation: Dynamic Systems

We now tackle the dynamic case where the tag population may vary during the estimation process. The objective for the dynamic systems is to promptly detect the global tag population change and accurately estimate the quantity of this change. To that end, we first establish the system model and then present our estimation algorithm.

6.3.5.1 System Dynamics and Measurement Model

In dynamic RFID systems, we can formulate the system dynamics as

$$z_{k+1} = z_k + w_k, \quad (6.30)$$

where the tag population z_{k+1} in frame $k+1$ consists of two parts: i) the tag population in frame k and ii) a random variable w_k which accounts for the stochastic variation of tag population resulting from the tag arrival/departure during frame k . Notice that w_k is referred to as process noise in Kalman filters and the appropriate characterisation of w_k is crucial in the design of stable Kalman filters, which will be investigated in detail later. Besides, the measurement model is the same as the static case. Hence, the discrete-time model for dynamic RFID systems can be characterized by (6.30) and (6.27).

6.3.5.2 Tag Population Estimation Algorithm

In the dynamic case, we leverage the two-step EKF to estimate the system state combined with the CUSUM test to further trace the tag population fluctuation.

Our main estimation algorithm is illustrated in Algorithm 14. The difference compared to the static scenario is that tag population variation needs to be detected by the CUSUM test presented in Algorithm 15 in the next subsection and the output of Algorithm 15 acts as a feedback to ϕ_k because due to the tag population variation, ϕ_k is no more a constant after the J th round as the static case. The overall structure of the estimation algorithm is illustrated in Fig. 6.1. We note that in the case where z_k is constant, Algorithm 14 degenerates to Algorithm 13.

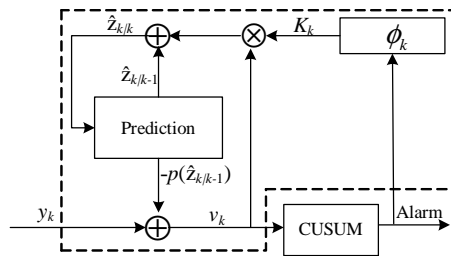


Figure 6.1: Estimation algorithm diagram: Dashed box indicates the EKF.

Algorithm 14 Tag population estimation (unified framework): executed by the reader

Input: $z_0, P_{0|0}, q, J, L, \underline{\phi}, \overline{\phi}$, maximum number of rounds k_{max}

Output: Estimation set $\hat{S}_z = \{\hat{z}_{k|k} : k \in [0, k_{max}]\}$

- 1: **Initialisation:** $\hat{z}_{0|0} \leftarrow z_0, Q_0 \leftarrow q, S_z = \{\hat{z}_{0|0}\}$
- 2: **for** $k = 1$ to k_{max} **do**
- 3: $\hat{z}_{k|k-1} \leftarrow \hat{z}_{k-1|k-1}, L_k \leftarrow L, r_k \leftarrow 1.59L_k/\hat{z}_{k|k-1}, P_{k|k-1} \leftarrow P_{k-1|k-1} + Q_{k-1}$
- 4: Generate random seed Rs_k , broadcast (L_k, Rs_k) , and run *Listen-before-Talk* protocol
- 5: Obtain the number of idle slots N_k , and compute y_k and v_k using (6.25) and (6.10)
- 6: $Q_k \leftarrow q$
- 7: **if** $k \leq J$ **then**
- 8: $\phi_k \leftarrow \underline{\phi}$
- 9: **else**
- 10: Execute Algorithm 15
- 11: $\phi_k \leftarrow \text{output of Algorithm 15}$
- 12: **end if**
- 13: Calculate R_k and K_k using (6.29) and (6.9)
- 14: Update $\hat{z}_{k|k}$ and $P_{k|k}$ using (6.7) and (6.8)
- 15: $S_z \leftarrow S_z \cup \{\hat{z}_{k|k}\}$
- 16: **end for**

6.3.5.3 Detecting Tag Population Change: CUSUM Test

The CUSUM Detection Framework. We leverage the CUSUM test to detect the change of tag population and further adjust ϕ_k . CUSUM test is a sequential analysis technique typically used for change detection [86]. It is shown to be asymptotically optimal in the sense of the minimum detection time subject to a fixed worst-case expected false alarm rate [35].

In the context of dynamic tag population detection, the reader monitors the innovation process $v_k = y_k - p(\hat{z}_{k|k-1})$. If the number of the tags population is constant, v_k equals to u_k which is a Gaussian process with zero mean. In contrast, upon the system state changes, i.e., tag population changes, v_k drifts away from the zero mean. In our design, we use Φ_k as a normalized input to the CUSUM test by normalising v_k with its estimated standard variance, specified as follows:

$$\Phi_k = \frac{v_k}{\sqrt{(P_{k|k-1} + Q_{k-1})C_k^2 + \text{Var}[u_k]_{z_k=\hat{z}_{k|k-1}}}}. \quad (6.31)$$

The reader further updates the CUSUM statistics g_k^+ and g_k^- as follows:

$$g_k^+ = \max\{0, g_{k-1}^+ + \Phi_k - \Upsilon\}, \quad (6.32)$$

$$g_k^- = \min\{0, g_{k-1}^- + \Phi_k + \Upsilon\}, \quad (6.33)$$

$$g_k^+ = g_k^- = 0, \text{ if } \delta = 1, \quad (6.34)$$

where $g_0^+ = 0$ and $g_0^- = 0$. And $\Upsilon \geq 0$, referred to as reference value, is a filter design parameter indicating the sensitivity of the CUSUM test to the fluctuation of Φ_k . Moreover, by δ we define an indicator flag indicating tag population change:

$$\delta = \begin{cases} 1 & \text{if } g_k^+ > \theta \text{ or } g_k^- < -\theta, \\ 0 & \text{otherwise,} \end{cases} \quad (6.35)$$

where $\theta > 0$ is a pre-specified CUSUM threshold.

The detailed procedure of the change detection is illustrated in Algorithm 15.

Algorithm 15 CUSUM test: executed by the reader in frame k

Input: Υ, θ

Output: ϕ_k

- 1: **Initialisation:** $g_0^+ \leftarrow 0, g_0^- \leftarrow 0$
 - 2: Compute Φ_k using equation (6.31)
 - 3: $g_k^+ \leftarrow (6.32), g_k^- \leftarrow (6.33)$
 - 4: **if** $g_k^+ > \theta$ or $g_k^- < -\theta$ **then**
 - 5: $\delta \leftarrow 1, \phi_k \leftarrow \varphi_1(\delta), g_k^+ \leftarrow 0, g_k^- \leftarrow 0$
 - 6: **else**
 - 7: $\delta \leftarrow 0, \phi_k \leftarrow \varphi_1(\delta)$
 - 8: **end if**
 - 9: Return ϕ_k
-

Parameter tuning in CUSUM test. The choice of the threshold θ and the drift parameter Υ has a directly impact on the performance of the CUSUM test in terms of detection delay and false alarm rate. Formally, the average running length (ARL) $L(\mu^*)$ is used to denote the duration between two actions [28]. For a large θ , $L(\mu^*)$ can be approximated as ³

$$L(\mu^*) = \begin{cases} \Theta(\theta), & \text{if } \mu^* \neq 0, \\ \Theta(\theta^2), & \text{if } \mu^* = 0, \end{cases} \quad (6.36)$$

where μ^* denotes the mean of the process Φ_k .

In our context, ARL corresponds to the mean time between two false alarms in the static case and the mean detection delay of the tag population change in the dynamic case. It is easy to see from (6.36) that a higher value of θ leads to lower false alarm rate at the price of longer detection delay. Therefore, the choices of θ and Υ consists of a tradeoff between the false alarm rate and the detection delay.

Recall that Φ_k can be approximated to a white noise process, i.e., $\Phi_k \sim \mathcal{N}[\mu^*, \sigma^{*2}]$ with $\mu^* = 0$, $\sigma^* = 1$ if the system state does not change. Generically, as recommended in [181], setting θ and Υ as follows achieves good ARL from the engineering perspective.

$$\theta = 4\sigma^*, \quad (6.37)$$

$$\Upsilon = \mu^* + 0.5\sigma^*. \quad (6.38)$$

In the CUSUM framework, we set ϕ_k by $\varphi_1(\delta)$ as follows:

$$\varphi_1(\delta) = \begin{cases} \underline{\phi}, & \text{if } \delta = 1, \\ \bar{\phi}, & \text{if } \delta = 0. \end{cases} \quad (6.39)$$

The rationale is that once a change on the tag population is detected in frame k , ϕ_k is set to $\underline{\phi}$ to quickly react to the change, while ϕ_k sticks to $\bar{\phi}$ when no system change is detected.

³For two variables X, Y , asymptotic notation $X = \Theta(Y)$ implies that there exist positives c_1, c_2 and x_0 such that for $\forall X > x_0$, it follows that $c_1X \leq Y \leq c_2X$.

6.3.6 Performance Analysis

We now establish the stability and the accuracy of our estimation algorithms for both static and dynamic cases.

6.3.6.1 Static Case

Our analysis is composed of two steps. We first derive the estimation error and then establish the stability and the accuracy of Algorithm 13 in terms of the boundedness of estimation error.

Computing Estimation Error. We first approximate the non-linear discrete system by a linear one. To that end, as the function $p(z_k)$ is continuously differentiable at $z_k = \hat{z}_{k|k-1}$, using the Taylor expansion and the fact that $r_k = 1.59L_k/\hat{z}_{k|k-1}$, we have

$$p(z_k) = p(\hat{z}_{k|k-1}) + C_k(z_k - \hat{z}_{k|k-1}) + \chi(z_k, \hat{z}_{k|k-1}), \quad (6.40)$$

where

$$C_k = -\frac{1.59}{e^{1.59\hat{z}_{k|k-1}}}, \quad (6.41)$$

$$\chi(z_k, \hat{z}_{k|k-1}) = \sum_{j=2}^{\infty} \frac{1}{e^{1.59j!}} (1.59 - \frac{1.59z_k}{\hat{z}_{k|k-1}})^j. \quad (6.42)$$

Regarding the convergence of $\chi(z_k, \hat{z}_{k|k-1})$ in (6.42), assume that

$$z_k = a'_k \hat{z}_{k|k-1}, 0.37 < a'_k < 1.63, \quad (6.43)$$

we can obtain the following boundedness of the residual for the case $0.61z_k < \hat{z}_{k|k-1} < 2.7z_k$:

$$\begin{aligned} |\chi(z_k, \hat{z}_{k|k-1})| &= \frac{1.59^2(\hat{z}_{k|k-1} - z_k)^2}{e^{1.59\hat{z}_{k|k-1}^2}} \sum_{j=0}^{\infty} \frac{1.59^j}{(j+2)!} \left|1 - \frac{z_k}{\hat{z}_{k|k-1}}\right|^j \\ &\leq \frac{1.59^2(\hat{z}_{k|k-1} - z_k)^2}{2e^{1.59\hat{z}_{k|k-1}^2}} \sum_{j=0}^{\infty} 1.59^j \left|1 - \frac{z_k}{\hat{z}_{k|k-1}}\right|^j \\ &\leq \frac{1.59^2(\hat{z}_{k|k-1} - z_k)^2}{2e^{1.59\hat{z}_{k|k-1}^2} [1 - |1.59(1 - \frac{z_k}{\hat{z}_{k|k-1}})|]} \leq \frac{1.59^2(\hat{z}_{k|k-1} - z_k)^2}{2e^{1.59} a_k \hat{z}_{k|k-1}^2}, \end{aligned} \quad (6.44)$$

where

$$a_k = 1 - 1.59|1 - a'_k|. \quad (6.45)$$

Recall the definition of the estimation error in (6.12) and using (6.22), (6.5) and (6.7), we can derive the estimation error $e_{k+1|k}$ as follows:

$$\begin{aligned} e_{k+1|k} &= z_{k+1} - \hat{z}_{k+1|k} = z_k - \hat{z}_{k|k} = z_k - \hat{z}_{k|k-1} - K_k [C_k(z_k - \hat{z}_{k|k-1}) + \chi(z_k, \hat{z}_{k|k-1}) + u_k] \\ &= (1 - K_k C_k) e_{k|k-1} + s_k + m_k, \end{aligned} \quad (6.46)$$

where s_k and m_k are defined as

$$s_k = -K_k u_k, \quad (6.47)$$

$$m_k = -K_k \chi(z_k, \hat{z}_{k|k-1}). \quad (6.48)$$

Boundedness of Estimation Error. Having derived the dynamics of the estimation error, we now state the main result on the stochastic stability and accuracy of Algorithm 13. The core technique in the proof, detailed in our publication [229], consists of setting an appropriate Lyapunov function, $V_k(e_{k|k-1}) \triangleq \frac{e_{k|k-1}^2}{P_{k|k-1}}$ in our problem, and then employ Lyapunov drift analysis to prove the conditions in Lemma 6.2 are satisfied.

Theorem 6.5. Consider the discrete-time stochastic system given by (6.22) and (6.27) and Algorithm 13, the estimation error $e_{k|k-1}$ defined by (6.12) is exponentially bounded in mean square and bounded w.p.o., if the following conditions hold:

1. there are positive numbers \underline{q} , \bar{q} , $\underline{\phi}$ and $\bar{\phi}$ such that for every $k \geq 0$:

$$\underline{q} \leq Q_k \leq \bar{q}, \quad (6.49)$$

$$\underline{\phi} \leq \phi_k \leq \bar{\phi}, \quad (6.50)$$

2. The initialization must follow the rules:

$$P_{0|0} > 0, \quad (6.51)$$

$$|e_{1|0}| \leq \epsilon \quad (6.52)$$

with positive real number $\epsilon > 0$.

Remark. By referring to the design objective posed in Section 6.3.3, Theorem 6.5 proves the following properties of our estimation algorithm:

- the estimation error of our algorithm is bounded in mean square and the relative estimation error tends to zero;
- the estimated population size converges to the real population with exponential rate.

Moreover, the conditions in Theorem 6.5 can be interpreted as follows:

1. The inequalities (6.49) and (6.50) can be satisfied by the configuring the correspondent parameters in Algorithm 13, which guarantees the boundedness of the pseudo-covariance $P_{k|k-1}$.
2. The inequality (6.51) consists of establishing positive $P_{k|k-1}$ for every $k \geq 1$.
3. As a sufficient condition for stability, the upper bound ϵ may be too stringent. As shown in the simulation results in [229], stability is still ensured even with a relatively large ϵ .

6.3.6.2 Dynamic Case

Our analysis on the stability of Algorithm 14 for the dynamic case is also composed of two steps. First, we derive the estimation error. Second, we establish the stability and the accuracy of Algorithm 14 in terms of the boundedness of estimation error.

We first derive the dynamics of the estimation error as follows:

$$e_{k+1|k} = (1 - K_k C_k) e_{k|k-1} + s_k + m_k, \quad (6.53)$$

which differs from the static case (6.46) in s_k . In the dynamic case, we have

$$s_k = w_k - K_k u_k \quad (6.54)$$

Next, we show the boundedness of the estimation error in Theorem 6.6.

Theorem 6.6. *Under the conditions of Theorem 6.5, consider the discrete-time stochastic system given by (6.30) and (6.27) and Algorithm 14, if there exist time-varying positive real number λ_k , $\sigma_k > 0$ such that $E[w_k] \leq \lambda_k$ and $E[w_k^2] \leq \sigma_k$, then the estimation error $e_{k|k-1}$ defined by (6.12) is exponentially bounded in mean square and bounded w.p.o..*

The closed-form formulas of λ_k and σ_k are detailed in [229]. As in the static case, the conditions may be too stringent such that the results still hold even if the conditions are not satisfied, as illustrated in the simulations.

6.4 Missing Tag Detection Algorithm Design

6.4.1 Background and Related Work

Detecting missing tags is one of the most important RFID applications. According to the statistics presented in [75], inventory shrinkage, a combination of shoplifting, internal theft, administrative and paperwork error, and vendor fraud, resulted in 44 billion dollars in loss for retailers in 2014. Hence, missing tag detection has attracted extensive research attention. Existing missing tag detection algorithms can be classified into probabilistic algorithms and deterministic algorithms, summarized as below.

Probabilistic algorithms detect a missing tag event with a predefined probability. Tan *et al.* initiated the study of probabilistic detection and propose a solution called Trusted Reader Protocol (TRP) in [187]. TRP detects a missing tag event by comparing the pre-computed slots with those picked by the tags in the population. Follow-up works [134] [133] employ multiple seeds to increase the probability of the singleton slot. The latest probabilistic algorithm called RUN is proposed in [178]. Different from previous works, RUN considers the influence of unexpected tags and can work in the environment with unexpected tags.

Deterministic algorithms, on the other hand, is able to exactly identify which tags are absent. Li *et al.* developed a series of deterministic algorithms in [119] to reduce the radio collision and enable tag identification at the bit level. Subsequently, Zhang *et al.* proposed another suite of determine algorithms in [234] by storing the bitmap of tag responses in all rounds and comparing them to determine the present and absent tags. But how to configure the algorithm parameters is not theoretically analyzed. More recently, Liu *et al.* [130] enhanced the work by reconciling both 2-collision and 3-collision slots and filtering the empty slots to improve time efficiency.

In a broader context, tag identification and tag population estimation algorithms sometimes can also be used to detect missing tags. Specifically, tag identification algorithms (e.g., [113, 179]) identify all tags in the interrogation region. To detect missing tags, they can be executed to

obtain the IDs of the tags present in the population and then the missing tags can be found out by comparing the collected IDs with those recorded in the database. However, tag identification algorithms are usually time-consuming [119] as they are designed to identify all tags. Moreover, they fail to work when it is not allowed to read the IDs of tags due to privacy concern. Tag estimation algorithms (e.g., [39, 177, 239]), on the other hand, estimate the number of tags in the interrogation region. If more than a certain number of tags are absent in RFID systems, a missing tag event can be detected by comparing the estimation and the number of expected tags stored in the database. However, estimation error may be misinterpreted as missing tags and cause detection error, especially when there are only a few missing tags.

Compared to the state-of-the-art development, we formulate the missing tag detection problem in the multiple-group multiple-region scenario, which has not been addressed before. We provide a comprehensive analysis on this new problem and investigate how to devise optimum missing tag detection algorithms.

6.4.2 System Model and Problem Formulation

6.4.2.1 System Model

We consider a grouped RFID system composed of a mobile reader and G groups of tags distributed in R ($R \geq 1$) interrogation regions (e.g., R rooms), concisely referred to as regions. In case where a tag may be physically located in two regions, i.e., regions may overlap one with another, the tag only responds to reader queries regarding to the first region when it is interrogated. In this sense, we can treat the regions as non-overlapping ones.

We use \mathbb{E} to denote the set of the tags which are expected to be present and we denote its cardinality (i.e., the number of expected tags) by $|\mathbb{E}|$. The reader knows the IDs of all tags in \mathbb{E} but does not know the set of tags in each region. For presentation conciseness, we set the ID of group g ($1 \leq g \leq G$) to its index g . We assume every tag knows its group ID through a grouping algorithm, e.g. [124]. We also assume the reader knows the approximate number of tags of each group g actually present in each region r ($1 \leq r \leq R$), denoted by n_{gr} . The estimation of n_{gr} can be achieved by the reader by deactivating all tags not belonging to group g (using the ID of group g) and then using any state-of-the-art tag population estimation algorithm.

To make our analysis generic, we do not impose any physical constraints on tags, which can be either battery-powered active tags or lightweight passive ones energized by radio waves emitted by the reader. We follow the standard Listen-before-talk communication algorithm [88] between the reader and tags: the reader initiates communication first by sending commands and broadcasting the parameters to tags, such as the frame size, random seeds, and then each tag responds in its chosen time slot. Consider an arbitrary time slot, if no tag replies in this slot, it is called an *empty slot*; otherwise, it is called a *nonempty slot*. Only one bit is needed to distinguish an empty slot from a nonempty slot: 0 for an empty slot and 1 for a nonempty slot. During the communication, the tag-to-reader transmission rate and the reader-to-tag transmission rate may differ with each other. In practice, the former is either 40 – 640kb/s in the FMO encoding format or 5 – 320kb/s in the modulated subcarrier encoding format, while the later is normally 26.7 – 128kb/s [73].

Table 6.2 summarizes the main notations.

Table 6.2: Main Notations

Symbols	Descriptions
G	Number of groups
g	Group index and group ID
R	Number of interrogation regions
r	Region index
\mathbb{E}	Set of target tags that need to be monitored
n_{gr}	Number of tags of group g in region r
m_g	Number of missing tags in group g
M_g	Threshold of group g
P_{dg}	Probability of detecting a missing event of group g
α_g	System requirement on the detection reliability for group g
f	Length of Bloom filter in B-detect
k	Number of hash functions in B-detect
s	Hash function seed
P_{fp}	False positive rate of Bloom filter in B-detect
T_B	Execution time of B-detect
f_r	Bloom filter vector size in region r in AB-detect
k_g	Number of hash functions for group g in AB/GAB-detect
$P_{fp,g}$	False positive rate of Bloom filter for g in AB/GAB-detect
T_{AB}	Execution time of AB-detect
f_{gr}	Bloom filter vector length for group g in r in GAB-detect
T_{GAB}	Execution time of GAB-detect

6.4.2.2 Problem Formulation

We are interested in detecting missing tag event for each group g . Let m_g denote the number of missing tags in group g which is of course not known by the reader. Let M_g denote the threshold of group g . A missing event of group g denotes the event where there are at least M_g tags of group g missing in the system. Let P_{dg} denote the probability that the reader can detect a missing event of group g , we formulate the optimum missing tag detection problem as follows.

Definition 6.5 (Optimum missing tag detection problem). *The optimum missing tag detection problem is to devise an algorithm of minimum execution time which can detect a missing event for each group g with probability $P_{dg} \geq \alpha_g$ if $m_g \geq M_g$, where α_g is the requirement on the detection reliability for group g . When there is only one group in the system, the problem degenerates to the classical missing event detection problem.*

6.4.2.3 Design Rationale

To design missing tag detection algorithms in the multiple-region multiple-group case, we leverage a powerful technique called *Bloom filter* which is a space-efficient probabilistic data structure for representing a set and supporting set membership queries [32]. Let $h_1(\cdot), \dots, h_k(\cdot)$ denote k independent hash functions with uniformly distributed outputs. Given a set of elements S , a Bloom filter first constructs an array B of m bits, where each bit is initialized to 0, and for each item $e \in S$, it sets the k bits $B[h_1(e)\%m], \dots, B[h_k(e)\%m]$ to 1. To process a membership query

of whether item e is in set S , the Bloom filter returns true if all corresponding k bits are 1 (i.e., it returns $\bigwedge_{i=1}^k B[h_i(e)\%m]$). Bloom filters admit no false negatives but have false positives.

In our design, we explore the following three natural ideas, each corresponding to a proposed missing tag detection algorithm detailed in the next three subsections.

Baseline approach. To enable missing tag detection in the multiple-region multiple-group case, we let the reader use the same Bloom filter parameters in each region for each group of tags and construct the Bloom filter based on the responses from the tags to perform missing event detection. This approach, termed as *B-detect*, is a direct application of Bloom filter to solve our problem.

Adaptive approach. In the baseline approach B-detect, the reader uses the same parameters in each region, which may not be optimum in the case when tags are not evenly distributed across regions. Motivated by this observation, we develop an adaptive approach, named *AB-detect*, which enables the reader to use different parameters based on the number of tags in the interrogation region the reader queries. Specifically, for each region r , the reader executes one query, to which tags of all the groups in the region respond. The reader constructs a Bloom filter B_r for each region containing the response and aggregates B_r ($1 \leq r \leq R$) to form a virtual Bloom filter B^{AB} , based on which it detects missing event for each group.

Group-wise approach. We further develop a group-wise approach, referred to as *GAB-detect*. In GAB-detect, the reader executes G group-wise queries for each region r . Only tags of group g ($1 \leq g \leq G$) in the interrogation region respond to the g -th query. The reader then constructs a Bloom filter B_{gr} for each group g and aggregates B_{gr} ($1 \leq r \leq R$) to form a virtual Bloom filter B_{g*} using the technique in AB-detect, based on which it detects missing event for group g .

Table 6.3: Comparison of three algorithms

Baseline approach	Same Bloom filter parameters for all regions and groups
Adaptive approach	Different parameters across regions, same parameters for all groups in the same region
Group-wise approach	Different parameters across regions and groups

By sequentially analysing the above three approaches and mathematically comparing their performance, we gradually iron out an optimum detection algorithm that works in practice.

6.4.3 The Baseline approach

In the B-detect design to enable missing tag detection in the multiple-region case, we let the reader use the same parameters in each region and construct the Bloom filter based on the responses from the tags to perform missing event detection. Specifically, B-detect consists of two phases, detailed as below.

6.4.3.1 Protocol Description

Phase 1: Query and feedback collection. The reader performs a query in each region r with the same parameter setting (f, k, s) , where f is the length of the Bloom filter vector, k is the number of independent hash functions used to construct the Bloom filter vector, and s is the seed of the hash functions which is identical for all groups and regions. How their values are chosen is analysed in Sec. 6.4.3.2 on parameter optimisation. Upon receiving the request, each tag in region

r , regardless of the group to which it belongs, selects k slots $(h_v(ID) \bmod f)$ ($1 \leq v \leq k$) in the frame of f slots and replies in these slots. The reader then constructs a Bloom filter vector B_r with the responses from the tags in each region r as follows. Note there are two types of slots: empty slots and nonempty slots. According to the responses from tags, if slot i ($1 \leq i \leq f$) is empty, the reader sets $B_r(i) = 0$, otherwise it sets $B_r(i) = 1$.

Phase 2: Virtual Bloom filter construction and missing event detection. After interrogating all R regions, the reader combines the Bloom filter vectors B_r ($1 \leq r \leq R$) to a virtual Bloom filter B by XORing each bit of them, i.e., $B(i) = B_1(i) \oplus \dots \oplus B_R(i)$. The reader then performs membership test. For each tag in \mathbb{E} , the reader maps its ID into k bits at positions $(h_v(ID) \bmod f)$ ($1 \leq v \leq k$). If all the corresponding bits in B are 1, then the tag is regarded as present. Otherwise, the tag is considered to be missing. The reader reports a missing event in group g if the number of missing tags is at least M_g and no missing event otherwise.

6.4.3.2 Performance Optimisation and Parameter Tuning

The execution time of B-detect, defined as T_B in number of slots, can be written as

$$T_B = R(t_1 + f\delta) \simeq Rf\delta, \quad (6.55)$$

where t_1 denotes the time for the reader to broadcast the query parameters and δ denotes the slot duration which we normalize to 1 for notation conciseness. In a large RFID system, it holds that $f \gg t_1$, so we ignore t_1 . In this subsection, we derive the optimum value of f that minimizes T_B .

It is well-known that there is no false negative in the Bloom filter membership test and the false positive rate P_{fp} for an arbitrary group g can be calculated as follows [32]:

$$P_{fp} = \left[1 - \left(1 - \frac{1}{f} \right)^{(|\mathbb{E}| - m)k} \right]^k \approx (1 - e^{-(|\mathbb{E}| - m)k/f})^k, \quad (6.56)$$

where $m = \sum_{g=1}^G m_g$ denotes the total number of missing tags in all groups.

By rearranging (6.56), we can express the Bloom filter size as

$$f = \frac{-(|\mathbb{E}| - m)k}{\ln(1 - P_{fp}^{1/k})}. \quad (6.57)$$

The following theorem derives the optimal values of f and k in the sense of minimising the execution time.

Theorem 6.7. *The optimum size of the Bloom filter and the optimum number of hash functions in B-detect, denoted by f^* and k^* respectively, that minimize the execution time while satisfying the detection reliability requirement for each group g regardless of m_g , are as follows:*

$$f^* = (|\mathbb{E}| - M) \cdot \frac{k^*}{-\ln(1 - X_{g^*}^{1/k^*})}, \quad k^* = \frac{\ln\left(1 - \alpha_{g^*}^{1/M_{g^*}}\right)}{\ln \frac{1}{2}}, \quad (6.58)$$

where $M = \sum_{g=1}^G M_g$, $X_g \triangleq 1 - \alpha_g^{1/M_g}$, and $g^* = \arg \min_g X_g$.

Given the practical meaning of k^* and f^* , both of them should be further rounded to the smallest integers not smaller than themselves.

6.4.4 The Adaptive Approach

In B-detect, the reader uses the same parameters in each region, particularly the length of the Bloom filter, which may not be optimum in the case when the tags are not evenly distributed across interrogation regions. Motivated by this observation, we develop another missing tag detection algorithm, named *AB-detect*, which enables the reader to use different parameters based on the number of tags in the region the reader queries.

6.4.4.1 Protocol Description

Phase 1: Query and feedback collection. The reader performs a query in each region r with the parameter $(f_r, \{k_g\}_{g=1}^G, s)$ where f_r is the length of the Bloom filter vector used in region r , k_g is the number of hash functions used by tags in group g , s is the hash seed which is identical for all groups and regions. There are two differences compared to the baseline approach. First, f_r may be different across different regions but are identical across groups; Second, k_g may be different across different groups but are identical across regions. We require f_r to be a power-multiple of two, i.e., $f_r = 2^{b_r}$, ($b_r \in \mathbb{N}$). As in B-detect, the reader constructs an f_r -bit Bloom filter vector B_r with the responses from the tags in each region r . Without loss of generality, we assume that $f_1 \leq f_2 \leq \dots \leq f_R$.

Phase 2: Virtual Bloom filter construction and missing event detection. After interrogating all R regions, the reader first expand B_r to an f_R -bit padded Bloom filter by repeating B_r $\frac{f_R}{f_r}$ times. Denote the padded Bloom filter as PB_r . The reader then combines PB_r ($1 \leq r \leq R-1$) and B_R to a virtual Bloom filter B^{AB} by XORing each bit of them, i.e., $B^{AB}(i) = PB_1(i) \oplus \dots \oplus PB_{R-1}(i) \oplus B_R(i)$ ($1 \leq i \leq f_R$), as illustrated in Fig. 6.2. The reader then performs membership test. For each tag in group g , the reader maps its ID into k_g bits at positions $(h_v(ID) \bmod f_R)$ ($1 \leq v \leq k_g$). If all the corresponding bits in B^{AB} are 1, then the tag is regarded as present. Otherwise, the tag is considered to be missing. The reader reports a missing event for group g if the number of missing tags in the group g is at least M_g and no missing event otherwise.

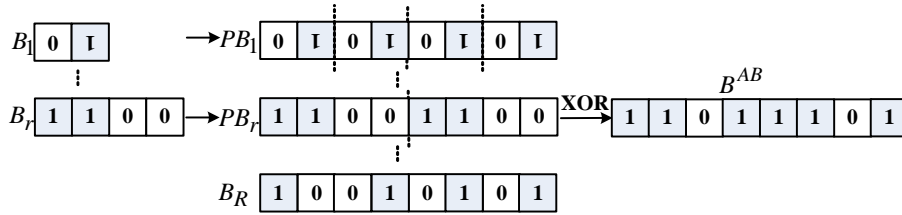


Figure 6.2: An illustrative example of constructing virtual Bloom filter.

Lemma 6.6. *There is no false negative in AB-detect.*

6.4.4.2 Performance Optimisation and Parameter Tuning

We investigate how to tune the parameters in AB-detect to minimize the execution time while ensuring the reliability requirement of each group. We first formulate the false positive rate for each group g , defined as $P_{fp,g}$. Recall the construction of B^{AB} in AB-detect, the probability that

any bit in B^{AB} is zero is $\prod_{r=1}^g \left(1 - \frac{1}{f_r}\right)^{\sum_{g=1}^G k_g n_{gr}}$. The false positive rate for group g can then be derived as

$$P_{fp,g} = \left[1 - \prod_{r=1}^R \left(1 - \frac{1}{f_r}\right)^{\sum_{g=1}^G k_g n_{gr}}\right]^{k_g} \approx \left(1 - e^{-\sum_{r=1}^R \sum_{g=1}^G \frac{k_g n_{gr}}{f_r}}\right)^{k_g}. \quad (6.59)$$

The following theorem derives the optimal values of f_r and k_g that minimize the execution time while ensuring the group-wise reliability requirement.

Theorem 6.8. *The optimum Bloom filter vector size for the region r and the number of hash functions for the group g , denoted as f_r^* and k_g^* , that minimize the execution time while satisfying the detection reliability requirement for each group g regardless of m_g , are as follows:*

$$f_r^* = \frac{\sqrt{\sum_{g=1}^G k_g^* n_{gr}} \cdot \sum_{r=1}^R \sqrt{\sum_{g=1}^G k_g^* n_{gr}}}{\min_g Y_g^*}, \quad k_g^* = \frac{\ln(1 - \alpha_g^{\frac{1}{M_g}})}{\ln \frac{1}{2}}, \quad (6.60)$$

where $Y_g^* \triangleq -\ln[1 - (1 - \alpha_g^{\frac{1}{M_g}})^{\frac{1}{k_g^*}}]$. The minimum execution time under the above setting, defined as T_{AB}^* , is:

$$T_{AB}^* = \frac{1}{\min_{1 \leq g \leq G} Y_g^*} \left(\sum_{r=1}^R \sqrt{\sum_{g=1}^G k_g^* n_{gr}} \right)^2. \quad (6.61)$$

As k_g^* needs to be an integer and f_r a power-multiple of two, they need to be rounded to the smallest integer and power-multiple of two not smaller than themselves.

6.4.4.3 Performance Comparison: B-detect vs. AB-detect

Theorem 6.9. *Given the optimum parameters in B-detect and AB-detect, the following relationship between the minimum execution time of B-detect T_B^* and AB-detect T_{AB}^* holds: $\frac{1}{R} \leq \frac{T_{AB}^*}{T_B^*} \leq 2$.*

Theorem 6.9 leads to the following engineering implications.

- In the worst case, AB-detect doubles the execution time compared to B-detect;
- In a large asymmetric system where the number of regions R is large, AB-detect can achieve significant performance gain.

6.4.5 The Group-wise Approach

In AB-detect, the reader constructs one Bloom filter that contains the response bits of tags of all groups in the interrogation region. Mixing responses from tags of different group may cause "interference" among groups and thus may increase the detection time for certain groups. Motivated by this observation, we develop a group-wise approach, termed as *GAB-detect*, in which the reader queries one group each time and constructs group-wise Bloom filters to eliminate the inter-group interference.

6.4.5.1 Protocol Description

Phase 1: Query and feedback collection. The reader performs G queries in each region r . In the g -th query ($1 \leq g \leq G$), the reader broadcasts a tetrad (g, k_g, f_{gr}, s) where g is the group ID of group g , k_g is the number of hash functions used by group g tags, f_{gr} is the Bloom filter size used in region r for group g , s is the hash seed which is identical for all regions and groups. Again, we require f_{gr} to be a power-multiple of two. Without loss of generality, we assume that $f_{g1} \leq f_{g2} \leq \dots \leq f_{gR}$. When receiving the query, each tag compares its group ID with g . If the tag does not belong to the group being queried, it keeps silent and waits for the next query. Otherwise, the tag selects k_g positions $(h_v(ID) \bmod f_{gr})$ ($1 \leq v \leq k_g$) in the frame of f_{gr} slots and transmits a short response at each of the k_g slots. The reader then constructs a Bloom filter for each group g and each region r , denoted by B_{gr}^{GAB} .

Phase 2: Virtual Bloom filter construction and missing event detection. After interrogating all R regions, the reader combines B_{gr}^{GAB} ($1 \leq r \leq R - 1$) to a virtual Bloom filter B_{g*}^{GAB} for each group g by using the expansion and combination technique in AB-detect. The reader then performs membership test for each group g by using B_{g*}^{GAB} .

6.4.5.2 Performance Optimisation and Parameter Tuning

We investigate how to tune algorithm parameters in GAB-detect to minimize the execution time while ensuring the reliability requirement of each group. We first derive the false positive rate of GAB-detect for any group g , defined as $P_{fp,g}$. Recall the construction of B_{g*}^{GAB} , the probability that any bit in B_{g*}^{GAB} is zero is $\prod_{r=1}^R \left(1 - \frac{1}{f_{gr}}\right)^{k_g n_{gr}}$. Hence, the false positive rate for group g can be derived as

$$P_{fp,g} = \left[1 - \prod_{r=1}^R \left(1 - \frac{1}{f_{gr}}\right)^{k_g n_{gr}}\right]^{k_g} \approx \left(1 - e^{-\sum_{r=1}^R \frac{k_g n_{gr}}{f_{gr}}}\right)^{k_g}. \quad (6.62)$$

The following theorem derives the optimal values of f_{gr} and k_g that minimize the execution time while ensuring the group-wise reliability requirement.

Theorem 6.10. *The optimum Bloom filter vector size and number of hash functions for group g in region r , denoted as f_{gr}^* and k_g^* , that minimize the execution time while satisfying the detection reliability requirement for each group g regardless of m_g , are:*

$$f_{gr}^* = \frac{\sqrt{n_{gr}} \cdot \sum_{r=1}^R \sqrt{n_{gr}}}{Z_g^*}, \quad k_g^* = \frac{\ln(1 - \alpha_g^{\frac{1}{M_g}})}{\ln \frac{1}{2}}, \quad (6.63)$$

The minimum execution time under the above setting, defined as T_{GAB}^* , is:

$$T_{GAB}^* = \sum_{g=1}^G \frac{\left(\sum_{r=1}^R \sqrt{n_{gr}}\right)^2}{Z_g^*}, \quad (6.64)$$

where $Z_g^* \triangleq \frac{\ln[1 - (1 - \alpha_g^{\frac{1}{M_g}})^{\frac{1}{k_g^*}}]}{-k_g^*}$.

6.4.5.3 Performance Comparison: AB-detect vs. GAB-detect

Theorem 6.11. *When f_r^* and f_{gr}^* are both power-multiples of two, it holds that $T_{AB}^* \geq T_{GAB}^*$.*

6.4.6 Discussion

We discuss some implementation issues of our proposed missing tag detection algorithms.

6.4.6.1 Estimating Tag Population

In our algorithms, the reader needs to estimate the number of tags in n_{gr} in each region and for each group. This may lead to extra overhead prior to missing tag detection. However, this overhead can be limited as the estimation can be achieved in $O(\log n_{gr})$ time using state-of-the-art estimation approaches. Specifically, we can apply two types of methods to estimate n_{gr} : single-group estimator and multi-group estimator. In the single-group estimator, when staying at region r the reader queries with the group ID g and only the tags from g respond. Then it operates like a single-group system. n_{gr} can be estimated by the methods in [39]. On the other hand, multi-group estimator estimates multiple group sizes simultaneously by employing the maximum likelihood estimation method as in [132], which is time-efficient.

Despite the extra overhead due to estimation of n_{gr} , this estimation phase enables the pre-detection of missing tags if the number of missing tags is important (e.g., due to unexpected loss or accidents). More specifically, the reader can achieve pre-detection by comparing the bitmaps constructed by the tag feedbacks and computed a priori by the reader. If a bit that is 1 in the pre-calculated bitmap by reader but turns out to be 0 in the bitmap of the feedbacks, the reader can identify the absence of tags mapped into this slot. If the number of missing tag for a given group exceeds the threshold, a missing event is reported for the group. Consequently, the reader may not need to execute the fine-grained detection algorithms since missing tag events have already been detected in the estimation phase, thus reducing the time cost.

6.4.6.2 Presence of Unknown/Unexpected Tags

Unknown and unexpected tags can be interpreted as the tags that have not been identified by the reader [129], such as newly arrived products, on which the reader does not have any knowledge. During the interrogation, the unknown tags will respond together with the known tags, which results in the interference to the detection of missing known tags and thus degrades the performance [178, 228].

Fortunately, two of our proposed algorithms, AB-detect and GAB-detect, are resistant to the interference caused by unknown tags. The reason is as follows. The unknown tags have not been identified by the reader, so they do not have their individual group IDs [124] such that no group ID in the interrogation messages matches with theirs. Therefore, unknown tags stay silent during the whole detection process.

6.5 Conclusion and Perspectives

In this chapter, we have presented a systematic research on a number of research problems related to tag counting and monitoring, one of the most fundamental component in RFID systems, particularly when the system scales. Specifically, we have addressed the following problems ranging from theoretical modeling and analysis, to algorithm design and optimization: stability analysis of the FSA, tag population estimation in dynamic RFID systems, and missing tag detection.

We conclude this chapter by outlining two research perspectives, the first one related to infrastructural RFID systems and the second one in a broader and more generical context related to big data networks. Today, an emerging trend is to treat universally-deployed RFID tags as a new wireless infrastructure, on which novel applications can be developed. However, existing research works on RFID are mostly focused on inventory control and supply chain management and treat tags simply as ID carriers to identify the object they are attached to. The ubiquitous deployment of RFID systems and the emerging trend of building RFID systems as a new wireless infrastructure (instead of individual tags and readers) bring many new research problems that underlie many useful applications, many of which have never been studied before. Hence, significant research efforts need to be devoted to the design of efficient management and monitoring algorithms in this context. In this regard, we have just started a CNRS PEPS research project MIRFID (Management of Infrastructural RFID systems: algorithm design and implementation), of which I am the coordinator. The MIRFID project aims at designing efficient and privacy-preserving tag management and monitoring algorithms in large-scale infrastructural RFID systems.

In a broader context, we plan to adapt and extend the methodology in this chapter to develop efficient algorithms for measuring, monitoring, and analyzing big data transferred over the emerging big data networks, which presents clear analogy and similarity to the counting and monitoring problems we address in large-scale RFID systems. A natural starting point we are currently looking at is cardinality estimation (i.e., flow counting and monitoring) in the networks with big data.

Chapter 7

Conclusion and Perspectives

This habilitation thesis presents our works on some algorithmic problems of both fundamental and practical importance in wireless networks. Chapter 2 investigates channel rendezvous and neighbor discovery and presents a series of distributed channel rendezvous and neighbor discovery algorithms enabling users to meet and discover each other within bounded and order-optimum delay. Chapter 3 addresses the problem of opportunistic channel access by providing a generic analysis to cast the problem into the RMAB problem and conducting a systematic analysis on a class of myopic policies of both theoretical and practical importance. Chapter 4 considers distributed learning in wireless networks and presents the design of distributed algorithms allowing users to gradually converge to a stable and desirable system state based on purely location information and interactions. Chapter 5 studies a class of path optimization and the related scheduling problems arising from data harvesting and mobile charging and presents a generic analysis on these problems and designs polynomial or quasi-polynomial time algorithm achieving constant or poly-logarithmic approximation to the optimum. Chapter 6 focuses on algorithm design and analysis in RFID systems by designing a number of basic algorithms including tag population estimation and missing tag detection.

Despite the variety of problems we address, our work is driven by the common and fundamental quest for algorithms that can scale elegantly, act efficiently in terms of computation and communication, while keeping operations as local and distributed as possible. Each of the chapters has stated the corresponding optimization problem and established either a methodology for finding its exact solution or, in some cases, efficient algorithms for its approximation. This is also accompanied by an analysis of the properties of the resulting system optimum, including, in particular, the establishment of analytic and asymptotic bounds on their performance, as well as a study of the mathematical problems hinging behind and its generalization.

In this concluding chapter, we first provide a brief description of other works we have done that are related to the central thread of the thesis. We then take a broader and further view to discuss more general perspectives and directions for future research.

7.1 A Glimpse on other Works

7.1.1 Auction-based Resource Allocation

Due to their perceived fairness and allocation efficiency, auctions are among the best-known market-based mechanisms to allocate resources. However, conventional auctions cannot be directly applied in wireless networks because radio resources are by nature different to conventional goods due to radio interference and reuse. Radio resource auction is essentially a problem of interference-constrained resource allocation. Motivated by this analysis, we have studied a number of problems arising from radio resource auctions summarized below.

- *Spectrum auction* [44, 47]. We propose an auction framework for cognitive radio networks to allow unlicensed secondary users (SU) to share the available spectrum of licensed primary users (PU) fairly and efficiently, subject to the interference temperature constraint at each PU. To study the competition among SUs, we formulate a non-cooperative multiple-PU multiple-SU auction game and study the structure of the resulting equilibrium by solving a non-continuous two-dimensional optimization problem, including the existence, uniqueness of the equilibrium and the convergence to the equilibrium in the two auctions. A distributed algorithm is developed in which each SU updates its strategy based on local information to converge to the equilibrium. We also analyze the revenue allocation among PUs and propose an algorithm to set the prices under the guideline that the revenue of each PU should be proportional to its resource. We then extend the proposed auction framework to the more challenging scenario with free spectrum bands. We develop an algorithm based on the no-regret learning to reach a correlated equilibrium of the auction game. The proposed algorithm, which can be implemented distributedly based on local observation, is especially tailored in decentralized adaptive learning environments as cognitive radio networks.
- *Truthful auction in mesh community networks* [138]. Nowadays, the maintenance costs of wireless devices represent one of the main limitations to the deployment of Wireless Mesh Networks (WMN) as a means to provide Internet access in urban and rural areas. A promising solution to this issue is to let the WMN operator lease its available bandwidth to a subset of customers, forming a Wireless Mesh Community Network, in order to increase network coverage and the number of residential users it can serve. Motivated by the above analysis, we design an economically efficient and resilient auction-based bandwidth allocation in WMNs. Our particular emphasis is on the resilience of the proposed mechanism against any actions of selfish customers that manipulate the bandwidth marketplace of the network scenario described above to obtain extra benefit. To tackle this problem, we design an optimum truthful auction that forces each customer interested in leasing the available bandwidth to bid its real valuation of the required bandwidth demand. More specifically, the approach consists of finding the optimum set of customers to be accepted by the operator (auction winners), whose traffic demands can be routed through the WMN, and the corresponding prices they have to pay for the leased service, which constitute the operator revenue. The optimum allocation and the pricing together ensure the truthfulness of the proposed auction scheme.
- *Auction-based mobile data offloading* [155, 156, 204, 208]. The opportunistic utilization of third party WiFi access devices to offload customer traffic from the mobile network has recently gained momentum as a promising approach to increase the network capacity and

simultaneously reduce the energy consumption of the radio access network (RAN) infrastructure. To foster the opportunistic utilization of unexploited Internet connections, we propose a new and open market where a mobile operator can lease the bandwidth made available by third parties (residential users or private companies) through their access points to increase dynamically (and adaptively) the network capacity. Specifically, we propose and analyze a combinatorial reverse auction to implement a market both for selecting the cheapest third party access devices and offloading the maximum amount of data traffic from the radio access networks. We show that a payment rule that considers only the variation of the objective function solving the problem with and without the winner does not always ensure the individual rationality of the participants for the analyzed mobile data offloading problem. We present a novel payment rule based on the Vickrey-Clarke-Groves (VCG) scheme and demonstrate that it guarantees both individual rationality and truthfulness. Since the optimum reverse auction problem is NP-hard, we further propose three greedy algorithms that solve the allocation problem in polynomial time while preserving the truthfulness property.

7.1.2 Towards Privacy-preserving Resource Allocation Algorithms

In a typical resource allocation algorithm, users submit their demands (e.g., bids in auctions) to the resource manager who allocates the resource to users based on their demands. In such context, revealing one's demand naturally opens the door for many security vulnerabilities. For example, a malicious auctioneer can exploit such information for future auctions, as historical data can be used to evaluate the willingness to pay of users. Hence, a critical research issue is how to design privacy-preserving resource allocation algorithms that do not leak any information to any entity other than the outcome of the allocation. To address this problem, we develop a generic framework based on garbled circuits and secret sharing and apply the framework in the following context.

- *Privacy-preserving spectrum auction* [56, 58]. We design an *information-theoretically secure* framework, termed as ITSEC, for truthful spectrum auctions, which ensures the privacy of bidders' bid information against any adversary with *arbitrary computation power*. We would like to put the emphasis on cryptographical security, where a protocol is said to be secure if no participating party can learn any information beyond the output of the protocol. Using this formal security criteria, existing approaches indeed reveal certain information that cannot be inferred from the output. Moreover, ITSEC brings almost no extra computation overhead to the underlining spectrum auction and incurs only limited communication overhead. Technically, ITSEC introduces two separate entities, a seller agent and a buyer agent, to cooperate with the auctioneer to securely run the auction. Note that none of the three parties need to be a trusted party, but any two of them are assumed not to collude. As a distinguished feature, ITSEC reveals nothing about the bids to any adversary with unbounded computation power, except for the auction result. We also design circuits for spectrum auction mechanisms implementing ITSEC, and optimize the circuits to further improve performance.
- *Privacy-preserving cloud auction* [57]. We develop a privacy-preserving cloud auction framework that runs without disclosing any bid information. Specifically, we focus on a truthful cloud auction scenario, where a cloud provider provides various computing resources to a large number of heterogeneous users. Being a multi-unit combinatorial auction problem, finding its optimum is NP-hard. A greedy resource allocation scheme was then proposed to achieve reasonable economic and computational efficiency. Our work is to design a privacy-preserving mechanism that prevent the auction from disclosing any information on user bids.

Compared with other combinatory auctions, privacy preserving design in cloud auctions has the following specific challenges. The first one is the heterogeneity in both computing resources (VM instances) and user preferences. It is not clear how privacy-preserving auction can be performed under such heterogeneous environment. Secondly, cloud users may not stay online during the entire auction duration. In other words, a user may switch offline after having submitted its bid. Last but not least, privacy-preserving auctions should be efficient, and scale nicely in both computation and communication so as to support auctions with a large number of users. The challenges in cloud auction privacy require our privacy-preserving framework to be data-oblivious, meaning that its execution path should not depend on the input. We thus develop a privacy-preserving algorithm uniquely composed of data-oblivious operation blocks. We then leverage tools in garbled circuits and homomorphic encryption to further preserve privacy of the whole auction algorithm. The complexity of our privacy-preserving cloud auction is $O(n \log^2 n)$ which is within a logarithmic factor of the complexity of the original auction algorithm $O(n \log n)$.

7.1.3 Game Theoretical Analysis on Smart Grid Security

Another research thrust that I have been developing is the application of game theory in the security analysis and defense strategy design in smart grids. In this regard, we employ game theoretical techniques to optimize the deployment of defense resources by focusing in particular on the impact of attacks on equipment. By analyzing the interactions between the attacker and the defender, we find the optimum choice of security modes to enable on each equipment in the Advanced Metering Infrastructure (AMI) to protect the confidentiality of customers' data. In addition, we find the minimum defense resources needed to thwart any attack attempt to compromise customers' data in the AMI. In the smart grid, the interdependency between the communication and the electric infrastructures also renders the management of the overall security risk a challenging task. We address this issue by presenting an analytical model for identifying and hardening the most critical communication equipment used in the power system. Using non-cooperative game theory, we model the interactions between an attacker and a defender, and derive the minimum defense resources required and the optimum strategy of the defender that minimizes the risk on the power system. We validate our model via a case study based on the polish electric power transmission system. These works (cf. [97–100] for details) are conducted in collaboration with Ziad Ismail (formal Ph.D. student co-supervised with Pr. Jean Leneutre at Telecom ParisTech and Dr. David Bateman at EDF). We are now extending our research to generic critical infrastructures.

7.2 Perspectives and Future Works

Due to our limited version and capacity, our works presented in this thesis have a limited scope and coverage in both width and depth, and only touches a tiny subset of aspects in the algorithm design and analysis in wireless networks. Below we try our best to outline some potential perspectives in a much broader and more generical context compared to the perspectives given in the concluding section of each chapter.

7.2.1 Orchestrating Randomness and Determinism

It is well-known that randomized algorithms are often conceptually easy to implement, produce good average results, and have better time complexity compared to deterministic algorithms. Randomness is even indispensable for many algorithms to produce good results, such as the distributed learning algorithms in load balancing. However, random algorithms fail to bound the worst-case performance. Take the neighbor discovery problem we address as an example, probabilistic algorithms perform well in the average case by limiting the expected discovery delay. The main drawback of them is the lack of performance guarantee in terms of discovery delay. Deterministic algorithms, on the other hand, have good worst-case performance while usually have longer expected discovery delay.

A natural question is how to combine the advantage of both while limiting their side-effects. In our work, we propose an approach to interleave the probabilistic slots, where the operation of the algorithm is randomized, with the deterministic ones. As a result, we can tradeoff the worst-case performance with the average performance. Another example is the very recent result on the rendezvous problem where the authors beat the quadratic theoretical barrier on the worst-case rendezvous delay by utilizing a public source of randomness in conjunction with a Markovian hitter [54]. Generally speaking, how to systematically orchestrate randomness with determinism is an important research axe, especially in emerging networks where we often need to tradeoff worst-case and average performance.

7.2.2 Exploring Information Sharing

In wireless networks and other distributed systems, it is usually desirable to keep operations as local and distributed as possible. However, sometimes allowing sharing of information, even small quantity of information, may bring significant performance gain. Our work on the imitation-based spectrum access actually exploits a very simple form of information sharing, pair-wise imitation. In many distributed systems like wireless networks, information sharing requires communication, which may be expensive. It is therefore natural to quantify the benefits brought by information sharing and how much information to share in a distributed setting.

To instantiate this point, we can take the RMAB problem as an example. In the classical RMAB setting, a user chooses an action at each time from a set of actions. Since the rewards from each action are unknown before actually activating it, each player needs to balance between exploiting the action that yields the best payoffs so far and exploring new actions that may give even higher payoffs later. In this setting, sharing information among users can greatly expedite the exploration component in a distributed system. Each user can benefit from the information shared by other users to infer statistical properties of the actions to make better decisions. However, information sharing raises the issue of cooperative exploration among players. In addition, it also incurs a communication cost, which needs to be managed at an acceptable level. Therefore, how to integrate the information sharing and quantify the related tradeoffs deserve a comprehensive treatment.

7.2.3 Balancing Optimality and Complexity

A central objective in algorithm design is optimality, i.e., the algorithm should produce good results, ideally the optimum ones. However, many problems we encounter in practice are NP-hard

and thus do not admit efficient algorithms finding the optimum solution. By efficient we mean that the complexity is polynomial or quasi-polynomial in both time and space. Examples of these problems encountered in our thesis include the RMAB problem studied in Chapter 3 which is PSPACE-hard and the path optimization problems invoked in Chapter 6 which are either NP-hard or APX-hard. Given the impossibility of finding efficient algorithms in many cases, we argue that it is natural to take the following alternatives which we have used in our works.

- The first one is to seek sufficient conditions for simple and robust algorithms under which the optimality of such algorithms is guaranteed. We have used this methodology in Chapter 3 where we study myopic policies and derive the optimality conditions.
- The second one is to develop approximation algorithms with bounded efficiency loss compare to the system optimum. We have adopted this methodology in Chapter 6 in solving the path optimization problems.

The above methods should also be carefully tuned by taking into account specific constraints posed by the problems we address. For example, in some cases, even algorithms with polynomial complexity are too expensive in terms of practical operations and are thus impractical to implement. On the other hand, simple and implementable algorithms often implicate significant efficiency degradation. Therefore, a balance need to be carefully tuned. As there is no universal recipe in solving all the problems. A reasonable approach is to design algorithms with good efficiency-implementability trade-off for a class of problems which are sufficiently generic and extensible to solve a variety of other problems.

7.2.4 Towards Online Algorithm Design

One important omission in this thesis is perhaps the design of online algorithms to respond to each new input upon arrival. The key challenge is that previous decisions of the online algorithms cannot be revoked. We are currently extending our work on the charging path optimization to the online case where the charging requests are not known *a priori*. In stead, they are sent to the charger sequentially in an on-demand fashion. The key problem we are facing is to design an online charging scheduling algorithm with good performance in the face of uncertainty, since the future is unknown to the algorithm. Formally, we seek to design an algorithm with bounded and constant *competitive ratio*, which compares the performance of an online algorithm to that of an offline algorithm which is given the whole input sequence beforehand. We plan to model the situation as a two-person game between an online algorithm and a malicious adversary. The online algorithm's strategy is to minimize its cost (in terms of time or energy), while the adversary's strategy is to construct the worst possible input for the algorithm in terms of the cost. This formulation allows us to use the rich body of theories in game theory to design efficient online algorithms, random or deterministic.

We hope that by developing and analysis a set of online algorithms and comparing their efficiency with that of offline algorithms, we can get more insight on the structure properties of the problem and the guidelines in designing efficient algorithms.

In the coming era of big data and massively connected heterogeneous devices, it is evident that we need efficient algorithms more than ever, algorithms that can operate rapidly and adaptively, while generating good results. There is still a long way to go towards this ambitious objective. This thesis is only a starting point of a long and fascinating journey.

Bibliography

- [1] N. ABRAMSON. “The Aloha System: Another Alternative for Computer Communications”. *Proc. AFIPS Fall Joint Computer Conference*. 1970 (see p. 108)
- [2] H. ACKERMANN, P. BERENBRINK, S. FISCHER, and M. HOEFER. “Concurrent Imitation Dynamics in Congestion Games”. *Proc. PODC*. 2009 (see pp. 63, 67)
- [3] V. AGGARWAL, Y. ANEJA, and K. NAIR. Minimal spanning tree subject to a side constraint. *Computers and Operations Research*, 9:4 (1982), 287–296 (see p. 85)
- [4] R. AGRAWAL. Sample Mean Based Index Policies with $O(\log n)$ Regret for the Multi-Armed Bandit Problem. *Advances in Applied Probability*, 27:4 (1995), 1054–1078 (see p. 45)
- [5] R. AGRAWAL, D. TENEKETZIS, and V. ANANTHARAM. Asymptotically Efficient Adaptive Allocation Rules for the Multiarmed Bandit Problem with Switching. *IEEE Transactions on Automatic Control*, 33:10 (1988), 899–906 (see p. 60)
- [6] S. AHMAND, M. LIU, T. JAVIDI, Q. ZHAO, and B. KRISHNAMACHARI. Optimality of Myopic Sensing in Multichannel Opportunistic Access. *IEEE Transactions on Information Theory*, 55:9 (2009), 4040–4050 (see pp. 41, 44, 46–48)
- [7] C. ALOS-FERRER and K. H. SCHLAG. *The Handbook of Rational and Social Choice*. Oxford University Press, 2009. Chap. Imitation and Learning (see p. 63)
- [8] C. ALOS-FERRER and F. SHI. Imitation with asymmetric memory. *Economic Theory*, 49:1 (2012), 193–215 (see pp. 63, 74)
- [9] S. ALPERN. Hide and Seek Games. *Seminar, Institut fur Hoehere Studien, Wien*, (1976) (see p. 14)
- [10] S. ALPERN and S. GAL. The Theory of Search Games and Rendezvous and Discrimination for Resource Allocation in Shared Computer Systems. *International Series in Operations Research and Management Science*, (2002) (see pp. 14, 16)
- [11] S. ALPERN, R. FOKKINK, L. GASNIENIEC, R. LINDELAUF, and V. S. SUBRAHMANIAN. *Search Theory: A Game Theoretic Perspective*. Springer, 2013 (see p. 14)
- [12] A. ANANDKUMAR, N. MICHAEL, and A. TANG. “Index-based sampling policies for tracking dynamic networks under sampling constraints”. *Proc. INFOCOM*. 2010 (see pp. 45, 63)
- [13] V. ANANTHARAM, P. VARAIYA, and J. WALRAND. Asymptotically Efficient Adaptive Allocation Rules for the Multiarmed Bandit Problem with Switching. *IEEE Transactions on Automatic Control*, 32:11 (1987), 968–976 (see p. 45)
- [14] G. ANASTASI, M. CONTI, M. DI FRANCESCO, and A. PASSARELLA. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Networks*, 7:3 (2009), 537–568 (see p. 28)

-
- [15] E. J. ANDERSON and R. R. WEBER. The rendezvous problem on discrete locations. *Journal Applied Probability*, **27**:4 (1990), 839–851 (see pp. [14–16](#), [23](#))
- [16] D. L. APPLGATE, R. E. BIXBY, V. CHVATAL, and W. J. COOK. *The Traveling Salesman Problem: A Computational Study: A Computational Study*. Princeton University Press, 2011 (see pp. [84](#), [94](#))
- [17] N. ATANASOV, J. LE NY, and G. J. PAPPAS. Distributed Algorithms for Stochastic Source Seeking with Mobile Robot Networks. *ASME Journal of Dynamic Systems, Measurement, Control*, **137**:3 (2014), 1521–1533 (see p. [83](#))
- [18] P. AUER, N. CESA-BIANCHI, and P. FISCHER. Finite-time Analysis of the Multiarmed Bandit Problem. *Machine Learning*, **47**:2 (2002), 235–256 (see p. [45](#))
- [19] Y. AZAR, O. GUREL-GUREVICH, E. LUBETZKY, and T. MOSCIBRODA. “Optimal discovery strategies in white space networks”. *Proc. ESA*. 2011 (see p. [16](#))
- [20] M. BAKHT, M. TROWER, and R. H. KRAVETS. “Searchlight: won’t you be my neighbor”. *Proc. Mobicom*. 2012 (see p. [29](#))
- [21] N. BANSAL, A. BLUM, S. CHAWLA, and A. MEYERSON. “Approximation algorithms for deadline-TSP and vehicle routing with time-windows”. *Proc. STOC*. 2004 (see p. [94](#))
- [22] R. BAR-YEHUDA, G. EVEN, and S. M. SHAHAR. On approximating a geometric prize-collecting traveling salesman problem with time windows. *Journal of Algorithms*, **55**:1 (2005), 76–92 (see p. [94](#))
- [23] A. BARBATO, A. CAPONE, L. CHEN, F. MARTIGNON, and S. PARIS. “A Power Scheduling Game for Reducing the Peak Demand of Residential Users”. *Proc. GreenComm*. 2013 (see pp. [9](#), [10](#), [65](#))
- [24] A. BARBATO, A. CAPONE, L. CHEN, F. MARTIGNON, and S. PARIS. “Distributed Demand-Side Management in Smart Grid: How Imitation improves power scheduling”. *Proc. ICC*. 2015 (see pp. [9](#), [10](#), [65](#))
- [25] A. BARBATO, A. CAPONE, L. CHEN, F. MARTIGNON, and S. PARIS. “Distributed Learning Algorithms for Scheduling Games in the Future Smart Grid (Extended abstract)”. *Proc. NetGCoop*. 2014 (see pp. [9](#), [10](#), [65](#))
- [26] A. BARBATO, A. CAPONE, L. CHEN, F. MARTIGNON, and S. PARIS. A distributed demand-side management framework for the smart grid. *Computer Communications*, **57**:2 (2015), 13–24 (see pp. [9](#), [10](#), [65](#))
- [27] L. BARLETTA, F. BORGONOVO, and M. CESANA. A formal proof of the optimal frame setting for Dynamic-Frame Aloha with known population size. *IEEE Transactions on Information Theory*, **60**:11 (2014), 7221–7230 (see p. [109](#))
- [28] M. BASSEVILLE and I. V. NIKIFOROV. *Detection of abrupt changes: theory and application*. Prentice Hall Englewood Cliffs, 1993 (see p. [123](#))
- [29] P. BERENBRIK, T. FRIEDETZKY, L. GOLDBERG, and P. GOLDBERG. “Distributed Selfish Load Balancing”. *Proc. SODA*. 2011 (see p. [63](#))
- [30] K. BIAN, J.-M. PARK, and R. CHEN. “A quorum-based framework for establishing control channels in dynamic spectrum access networks”. *Proc. Mobicom*. 2009 (see p. [16](#))
- [31] K. BIAN, J.-M. PARK, and R. CHEN. “Asynchronous channel hopping for establishing rendezvous in CRNs”. *Proc. INFOCOM*. 2011 (see p. [16](#))

-
- [32] B. H. BLOOM. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, **13**:7 (1970), 422–426 (see pp. [107](#), [128](#), [130](#))
- [33] A. BLUM, S. CHAWLA, D. KARGER, T. LANE, A. MEYERSON, and M. MINKOFF. Approximation Algorithms for Orienteering and Discounted-Reward TSP. *SIAM Journal of Computing*, **37**:2 (2007), 653–670 (see pp. [85](#), [90](#), [91](#), [93](#), [94](#))
- [34] C. BORDENAVE, D. McDONALD, and A. PROUTIERE. Asymptotic stability region of slotted Aloha. *IEEE Transactions on Information Theory*, **58**:9 (2012), 5841–5855 (see pp. [108](#), [109](#))
- [35] E. BRODSKY and B. S. DARKHOVSKY. *Nonparametric methods in change point problems*. Springer Science & Business Media, 1993 (see p. [122](#))
- [36] C. CHEKURI and M. PAL. “A recursive greedy algorithm for walks in directed graphs”. *Proc. FOCS*. 2005 (see p. [101](#))
- [37] C. CHEKURI, G. EVEN, and G. KORTSARZ. A greedy approximation algorithm for the group steiner problem. *Discrete Applied Mathematics*, **154**:1 (2006), 15–34 (see p. [101](#))
- [38] C. CHEKURI, N. KORULA, and M. PÁL. Improved algorithms for orienteering and related problems. *ACM Transactions on Algorithms*, **8**:3 (2012), 1–27 (see p. [94](#))
- [39] B. CHEN, Z. ZHOU, and H. YU. “Understanding RFID counting protocols”. *Proc. Mobicom*. 2013 (see pp. [127](#), [134](#))
- [40] L. CHEN and K. BIAN. The Telephone Coordination Game Revisited: From Random to Deterministic Algorithms. *IEEE Transactions on Computers*, **64**:10 (2015), 2968–2980 (see pp. [8](#), [9](#), [14](#), [19](#), [22](#), [27](#))
- [41] L. CHEN, K. BIAN, and M. ZHENG. Never Live Without Neighbors: From Single- to Multi-Channel Neighbor Discovery for Mobile Sensing Applications. *IEEE/ACM Transactions on Networking*, **24**:5 (2016), 3148–3161 (see pp. [8](#), [9](#), [14](#))
- [42] L. CHEN, S. IELLAMO, and M. COUPECHOUX. “Opportunistic Spectrum Access with Channel Switching Cost for CRNs”. *Proc. ICC*. 2011 (see pp. [9](#), [42](#))
- [43] L. CHEN, Y. LI, and A. V. VASILAKOS. “Oblivious neighbor discovery for wireless devices with directional antennas”. *Proc. INFOCOM (extended version to appear in IEEE/ACM Transactions on Networking)*. 2016 (see pp. [8](#), [9](#), [14](#))
- [44] L. CHEN, S. IELLAMO, M. COUPECHOUX, and P. GODLEWSKI. “An Auction Framework for Spectrum Allocation with Interference Constraint in CRNs”. *Proc. INFOCOM*. 2010 (see p. [137](#))
- [45] L. CHEN, R. FAN, K. BIAN, L. CHEN, M. GERLA, T. WANG, and X. LI. “On heterogeneous neighbor discovery in WSNs”. *Proc. INFOCOM*. 2015 (see pp. [8](#), [9](#), [14](#))
- [46] L. CHEN, W. WANG, H. HUANG, and S. LIN. On Time-Constrained Data Harvesting in WSNs: Approximation Algorithm Design. *IEEE/ACM Transactions on Networking*, **24**:5 (2016), 3123–3135 (see pp. [10](#), [84](#), [88](#), [89](#), [92](#))
- [47] L. CHEN, S. IELLAMO, M. COUPECHOUX, and P. GODLEWSKI. Spectrum auction with interference constraint for CRNs with multiple primary and secondary users. *Wireless Networks*, **17**:5 (2011), 1355–1371 (see p. [137](#))
- [48] L. CHEN, W. WANG, H. HUANG, and S. LIN. “Time-constrained data harvesting in WSNs: Theoretical foundation and algorithm design”. *Proc. INFOCOM*. 2015 (see pp. [10](#), [84](#))

-
- [49] L. CHEN and K. BIAN. Neighbor Discovery in Mobile Sensing Applications. *Ad Hoc Networks*, **48**:9 (2016), 38–52 (see pp. 8, 9, 14)
- [50] L. CHEN, K. BIAN, and M. ZHENG. “Heterogeneous Multi-channel Neighbor Discovery Formobile Sensing Applications: Theoretical Foundationand and Protocol Design”. *Proc. ACM MobiHoc*. 2014 (see pp. 8, 9, 14, 34)
- [51] L. CHEN, S. LIN, and H. HUANG. “Charge Me if You Can: Charging Path Optimization and Scheduling in Mobile Networks”. *Proc. MobiHoc*. 2016 (see pp. 10, 84, 98, 100)
- [52] L. CHEN, K. BIAN, L. CHEN, C. LIU, J.-M. PARK, and X. LI. “A Group-theoretic Framework for Rendezvous in Heterogeneous CRNs”. *Proc. ACM MobiHoc*. 2014 (see pp. 8, 9, 14)
- [53] L. CHEN, W. WANG, H. HUANG, and S. LIN. “Time-constrained data harvesting in WSNs: Theoretical foundation and algorithm design”. *Proc. INFOCOM*. 2015 (see p. 94)
- [54] S. CHEN, M. DIPPEL, A. RUSSELL, A. SAMANTA, and R. SUNDARAM. “Markovian Hitters and the Complexity of Blind Rendezvous”. *Proc. SODA*. 2016 (see pp. 39, 140)
- [55] X. CHEN and J. HUANG. “Spatial spectrum access game: Nash equilibria and distributed learning”. *Proc. MobiHoc*. 2012 (see p. 63)
- [56] Z. CHEN, L. HUANG, and L. CHEN. “ITSEC: An information-theoretically secure framework for truthful spectrum auctions”. *Proc. INFOCOM*. 2015 (see p. 138)
- [57] Z. CHEN, L. CHEN, L. HUANG, and H. ZHONG. “On Privacy-Preserving Cloud Auction”. *Proc. SRDS*. 2016 (see p. 138)
- [58] Z. CHEN, L. CHEN, L. HUANG, and H. ZHONG. “Towards Secure Spectrum Auction: Both Bids and Bidder Locations Matter (Extended abstract)”. *Proc. MobiHoc*. 2016 (see p. 138)
- [59] D. CIULLO, G. CELIK, and E. MODIANO. “Minimizing transmission energy in sensor networks via trajectory control”. *Proc. WiOpt*. 2010 (see pp. 84, 94)
- [60] B. COLTIN and M. VELOSO. “Mobile robot task allocation in hybrid WSNs”. *Proc. IROS*. 2010 (see p. 83)
- [61] J. CZYZOWICZ, A. LABOUREL, and A. PELC. “How to Meet Asynchronously (Almost) Everywhere”. *Proc. SODA*. 2010 (see p. 16)
- [62] H. DAI, G. CHEN, C. WANG, S. WANG, X. WU, and F. WU. Quality of Energy Provisioning for Wireless Power Transfer. *IEEE Transactions on Parallel and Distributed Systems*, **26**:2 (2015), 527–537 (see p. 94)
- [63] G. DE MARCO, L. GARGANO, E. KRANAKIS, D. KRIZANC, A. PELC, and U. VACCARO. Asynchronous deterministic rendezvous in graphs. *Theoretical Computer Science.*, **355**:3 (2006), 315–326 (see p. 16)
- [64] A. DESSMARK, P. FRAIGNIAUD, D. R. KOWALSKI, and A. PELC. Deterministic Rendezvous in Graphs. *Algorithmica*, **46**:1 (2006), 69–96 (see p. 16)
- [65] L. DI PUGLIA PUGLIESE, F. GUERRIERO, D. ZORBAS, and T. M. RAZAFINDRALAMBO. Modelling the mobile target covering problem using flying drones. *Optimization Letters*, **10**:5 (2016), 1021–1052 (see p. 83)
- [66] A. DUMITRESCU and J. MITCHELL. “Approximation algorithms for TSP with neighborhoods in the plane”. *Proc. SODA*. 2001 (see p. 84)

-
- [67] P. DUTTA and D. CULLER. “Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications”. *Proc. SenSys*. 2008 (see pp. 29, 32)
- [68] N. EHSAN and M. LIU. “On the optimality of an index policy for bandwidth allocation with delayed state observation and differentiated services”. *Proc. INFOCOM*. 2004 (see p. 44)
- [69] E. EKICI, Y. GU, and D. BOZDAG. Mobility-based communication in WSNs. *IEEE Communications Magazine*, **44**:7 (2006), 56–62 (see pp. 84, 94)
- [70] J. ELIAS, F. MARTIGNON, L. CHEN, and M. KRUNZ. Distributed Spectrum Management in TV White Space Networks. *IEEE Transactions on Vehicular Technology (to appear)*, (2016) (see pp. 9, 10, 65)
- [71] J. ELIAS, F. MARTIGNON, L. CHEN, and E. ALTMAN. Joint Operator Pricing and Network Selection Game in CRNs: Equilibrium, System Dynamics and Price of Anarchy. *IEEE Transactions on Vehicular Technology*, **62**:9 (2013), 4576–4589 (see pp. 9, 10, 65)
- [72] G. ELLISON. Basins of Attraction, Long-Run Stochastic Stability, and the Speed of Step-by-Step Evolution. *Review of Economic Studies*, **67**:1 (2000), 17–45 (see pp. 64, 75–78)
- [73] EPCGLOBAL INC. Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz Version 1.0.9 (2005) (see pp. 107, 108, 127)
- [74] F. FARHADI and F. ASHTIANI. Stability Region of a Slotted Aloha Network with K-Exponential Backoff. *arXiv preprint:1406.4448*, (2014) (see pp. 108, 109)
- [75] N. R. FEDERATION. *National Retail Security Survey*. 2015 (see p. 126)
- [76] K. FINKENZELLE. *RFID handbook: Radio frequency identification fundamentals and applications*. John Wiley & Sons, 2000 (see p. 117)
- [77] D. FOSTER and P. YOUNG. Stochastic evolutionary game dynamics. *Theoretical Population Biology*, **38**:2 (Oct. 1990), 219–232 (see pp. 64, 75)
- [78] D. P. FOSTER and H. P. YOUNG. Regret testing: learning to play Nash equilibrium without knowing you have an opponent. *Theoretical Economics*, **1**:3 (2006), 341–367 (see p. 64)
- [79] J. FRIEDMAN and C. MEZZETTI. Learning in Games by Random Sampling. *Journal of Economic Theory*, **98**:1 (May 2001), 55–84 (see pp. 64, 78)
- [80] A. GANESH, S. LILIENTHAL, D. MANJUNATH, A. PROUTIERE, and F. SIMATOS. “Load Balancing via Random Local Search in Closed and Open Systems”. *Proc. SIGMETRICS*. 2010 (see p. 63)
- [81] S. GHEZ, S. VERDU, and S. C. SCHWARTZ. Optimal decentralized control in the random access multipacket channel. *IEEE Transactions on Automatic Control*, **34**:11 (1989), 1153–1163 (see p. 109)
- [82] S. GHEZ, S. VERDU, and S. C. SCHWARTZ. Stability properties of slotted Aloha with multipacket reception capability. *IEEE Transactions on Automatic Control*, **33**:7 (1988), 640–649 (see pp. 109, 111)
- [83] J. C. GITTINS and D. JONES. A Dynamic Allocation Index For the Sequential Design of Experiments. *Progress in Statistics*, (1974), 241–266 (see p. 43)
- [84] S. GUHA and K. MUNAGALA. “Approximation algorithms for partial-information based stochastic control with markovian rewards”. *Proc. FOCS*. 2007 (see p. 44)

-
- [85] S. GUHA and K. MUNAGALA. “Approximation algorithms for restless bandit problems”. *Proc. SODA*. 2009 (see p. 44)
- [86] F. GUSTAFSSON. *Adaptive filtering and change detection*. Wiley New York, 2000 (see p. 122)
- [87] A. HAGHANI and S. JUNG. A Dynamic Vehicle Routing Problem with Time-dependent Travel Times. *Computers and Operations Research*, **32**:11 (2005), 2959–2986 (see p. 94)
- [88] H. HAN, B. SHENG, C. C. TAN, Q. LI, W. MAO, and S. LU. “Counting RFID tags efficiently and anonymously”. *Proc. INFOCOM*. 2010 (see pp. 113, 127)
- [89] K. HAN, J. LUO, Y. LIU, and A. VASILAKOS. Algorithm design for data communications in duty-cycled wireless sensor networks: a survey. *IEEE Communications Magazine*, **51**:7 (2013), 107–113 (see p. 28)
- [90] L. HE, L. FU, L. ZHENG, Y. GU, P. CHENG, J. CHEN, and J. PAN. “Esync: An energy synchronized charging protocol for rechargeable WSNs”. *Proc. MobiHoc*. 2014 (see p. 94)
- [91] C. HELVIG, G. ROBINS, and A. ZELIKOVSKY. The moving-target traveling salesman problem. *Journal of Algorithms*, **49**:1 (2003), 153–174 (see p. 95)
- [92] Y. HU, X. WANG, and X. GAN. “Critical sensing range for mobile heterogeneous camera sensor networks”. *Proc. INFOCOM*. 2014 (see p. 94)
- [93] H. HUANG, S. LIN, L. CHEN, J. GAO, A. MAMAT, and J. WU. “Dynamic Mobile Charger Scheduling in Heterogeneous WSNs”. *Proc. MASS*. 2015 (see pp. 10, 84)
- [94] S. IELLAMO, L. CHEN, and M. COUPECHOUX. “Retrospective spectrum access protocol: A payoff-based learning algorithm for CRNs”. *Proc. ICC*. 2014 (see pp. 9, 10, 65)
- [95] S. IELLAMO, L. CHEN, M. COUPECHOUX, and A. V. VASILAKOS. “Imitation-based spectrum access policy for CRNs”. *Proc. ISWCS*. 2012 (see pp. 9, 10, 65)
- [96] S. IELLAMO, L. CHEN, and M. COUPECHOUX. Proportional and double imitation rules for spectrum access in CRNs. *Computer Networks*, **57**:8 (2013), 1863–1879 (see pp. 9, 10, 64, 65, 70, 72, 74)
- [97] Z. ISMAIL, J. LENEUTRE, D. BATEMAN, and L. CHEN. A Game Theoretical Analysis of Data Confidentiality Attacks on Smart-Grid AMI. *IEEE Journal on Selected Areas in Communications*, **32**:7 (2014), 1486–1499 (see p. 139)
- [98] Z. ISMAIL, J. LENEUTRE, D. BATEMAN, and L. CHEN. “A Game-Theoretical Model for Security Risk Management of Interdependent ICT and Electrical Infrastructures”. *Proc. HASE*. 2015 (see p. 139)
- [99] Z. ISMAIL, J. LENEUTRE, D. BATEMAN, and L. CHEN. “A Methodology to Apply a Game Theoretic Model of Security Risks Interdependencies Between ICT and Electric Infrastructures”. *Proc. GameSec*. 2016 (see p. 139)
- [100] Z. ISMAIL, C. KIENNERT, J. LENEUTRE, and L. CHEN. Auditing a Cloud Provider’s Compliance With Data Backup Requirements: a Game Theoretical Analysis. *IEEE Transactions on Information Forensics and Security*, **11**:8 (2016), 1685–1699 (see p. 139)
- [101] P. J. Value of information in optimal flow-level scheduling of users with Markovian time-varying channels. *Performance Evaluation*, **68**:11 (2011), 1022–1036 (see p. 44)
- [102] J. JEON and A. EPHREMIDES. On the Stability of Random Multiple Access With Stochastic Energy Harvesting. *IEEE Journal on Selected Areas in Communications*, **33**:3 (2015), 571–584 (see p. 109)

-
- [103] J.-R. JIANG. Expected quorum overlap sizes of quorum systems for asynchronous power-saving in mobile ad hoc networks. *Computer Networks*, **52**:17 (2008), 3296–3306 (see p. 29)
- [104] N. JOHNSON and S. KOTZ. *Urn models and their application: an approach to modern discrete probability theory*. Wiley, 1977 (see p. 109)
- [105] O. JONATHAN. A Continuous-Time Markov Decision Process for Infrastructure Surveillance. *Proc. Operations Research*, (2010), 327–332 (see p. 44)
- [106] A. KANDHALU, K. LAKSHMANAN, and R. RAJKUMAR. “U-connect: a low-latency energy-efficient asynchronous neighbor discovery protocol”. *Proc. IPSN*. 2010 (see pp. 29, 32)
- [107] N. KAROWSKI, A. C. VIANA, and A. WOLISZ. Optimized Asynchronous Multi-channel Discovery of IEEE 802.15.4-based Wireless Personal Area Networks. *IEEE Transactions on Mobile Computing*, **12**:10 (2013), 1972–1985 (see p. 28)
- [108] D. KIM, B. ABAY, R. N. UMA, W. WU, W. WANG, and A. TOKUTA. “Minimizing data collection latency in WSNs with multiple mobile elements”. *Proc. INFOCOM*. 2012 (see pp. 84, 91–93)
- [109] M. KODIALAM and T. NANDAGOPAL. “Fast and reliable estimation schemes in RFID systems”. *Proc. Mobicom*. 2006 (see pp. 113, 118)
- [110] M. KODIALAM, T. NANDAGOPAL, and W. C. LAU. “Anonymous tracking using RFID tags”. *Proc. INFOCOM*. 2007 (see pp. 107, 113)
- [111] V. F. KOLCHIN, B. A. SEVASTYANOV, and V. P. CHISTYAKOV. *Random allocation*. Wiley New York, 1978 (see p. 118)
- [112] S. C. KOMPALLI and R. R. MAZUMDAR. On the stability of finite queue slotted Aloha protocol. *IEEE Transactions on Information Theory*, **59**:10 (2013), 6357–6366 (see pp. 108, 109)
- [113] T. F. LA PORTA, G. MASELLI, and C. PETRIOLI. Anticollision protocols for single-reader RFID systems: temporal analysis and optimization. *IEEE Transactions on Mobile Computing*, **10**:2 (2011), 267–279 (see p. 126)
- [114] S. LAI. *Heterogenous Quorum-based Wakeup Scheduling for Duty-Cycled Wireless Sensor Networks*. PhD thesis. Virginia Polytechnic Institute and State University, 2009 (see p. 29)
- [115] T. L. LAI and H. ROBBINS. Asymptotically Efficient Adaptive Allocation Rules. *Advances in Applied Probability*, **6**:1 (1985), 4–22 (see p. 45)
- [116] F. E. LAPICCIRELLA, K. LIU, and Z. DING. “Multi-Channel Opportunistic Access Based on Primary ARQ Messages Overhearing”. *Proc. ICC*. 2011 (see pp. 41, 44)
- [117] G. LAPORTE. The traveling salesman problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, **59**:2 (1992), 231–247 (see p. 94)
- [118] J. LE NY, M. DAHLEH, and E. FERON. “Multi-UAV dynamic routing with partial observations using restless bandit allocation indices”. *Proc. ACC*. 2008 (see p. 44)
- [119] T. LI, S. CHEN, and Y. LING. “Identifying the missing tags in a large RFID system”. *Proc. MobiHoc*. 2010 (see pp. 126, 127)
- [120] T. LI, S. WU, S. CHEN, and M. YANG. “Energy efficient algorithms for the RFID estimation problem”. *Proc. INFOCOM*. 2010 (see p. 107)

-
- [121] Z. Y. LIN, H. LIU, X. W. CHU, and Y.-W. LEUNG. “Jump-stay based channel hopping algorithm with guaranteed rendezvous for CRNs”. *Proc. INFOCOM*. 2011 (see p. 16)
- [122] H. LIU, K. LIU, and Q. ZHAO. “Learning and Sharing in A Changing World: Non-Bayesian Restless Bandit with Multiple Players”. *Proc. ITA*. 2011 (see p. 45)
- [123] H. LIU, K. LIU, and Q. ZHAO. “Logarithmic Weak Regret of Non-Bayesian Restless Multi-Armed Bandit”. *Proc. ICASSP*. 2011 (see p. 45)
- [124] J. LIU, B. XIAO, S. CHEN, F. ZHU, and L. CHEN. “Fast RFID grouping protocols”. *Proc. INFOCOM*. 2015 (see pp. 127, 134)
- [125] K. LIU and Q. ZHAO. Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Wireless Communications*, **58**:11 (2010), 5667–5681 (see pp. 41, 44, 45)
- [126] K. LIU and Q. ZHAO. Indexability of Restless Bandit Problems and Optimality of Whittle Index for Dynamic Multichannel Access. *IEEE Transactions on Information Theory*, **56**:11 (2000), 5547–5567 (see p. 44)
- [127] K. LIU, Q. ZHAO, and B. KRISHNAMACHARI. Dynamic Multichannel Access With Imperfect Channel State Detection. *IEEE Transactions on Signal Processing*, **58**:5 (2010), 2795–2807 (see pp. 51, 52)
- [128] Q. LIU, K. WANG, and L. CHEN. On Optimality of Greedy Policy for a Class of Standard Reward Function of Restless Multi-armed Bandit Problem. *IET Transactions on Signal Processing*, **6**:6 (2012), 584–593 (see p. 52)
- [129] X. LIU, B. XIAO, S. ZHANG, and K. BU. Unknown tag identification in large RFID systems: An efficient and complete solution. *IEEE Transactions on Parallel and Distributed Systems*, **26**:6 (2015), 1775–1788 (see p. 134)
- [130] X. LIU, K. LI, G. MIN, Y. SHEN, A. X. LIU, and W. QU. Completely pinpointing the missing RFID Tags in a time-efficient way. *IEEE Transactions on Computers*, **64**:1 (2015), 87–96 (see p. 126)
- [131] X. LIU, K. LI, G. MIN, K. LIN, B. XIAO, Y. SHEN, and W. QU. Efficient unknown tag identification protocols in large-scale RFID systems. *IEEE Transactions on Parallel and Distributed Systems*, **25**:12 (2014), 3145–3155 (see p. 108)
- [132] W. LUO, Y. QIAO, and S. CHEN. “An efficient protocol for RFID multigroup threshold-based classification”. *Proc. INFOCOM*. 2013 (see p. 134)
- [133] W. LUO, S. CHEN, Y. QIAO, and T. LI. Missing-Tag Detection and Energy-Time Tradeoff in Large-Scale RFID Systems With Unreliable Channels. *IEEE/ACM Transactions on Networking*, **22**:4 (2014), 1079–1091 (see p. 126)
- [134] W. LUO, S. CHEN, T. LI, and Y. QIAO. “Probabilistic missing-tag detection and energy-time tradeoff in large-scale RFID systems”. *Proc. MobiHoc*. 2012 (see p. 126)
- [135] M. MA and Y. YANG. SenCar: An Energy-Efficient Data Gathering Mechanism for Large-Scale Multihop Sensor Networks. *IEEE Transactions on Parallel and Distributed Systems*, **18**:10 (2007), 1476–1488 (see p. 84)
- [136] J. R. MARDEN, H. P. YOUNG, G. ARSLAN, and J. S. SHAMMA. Payoff-Based Dynamics for Multiplayer Weakly Acyclic Games. *SIAM Journal on Control and Optimization*, **48**:1 (2009), 373–396 (see p. 64)

-
- [137] J. MARDEN, H. YOUNG, and L. PAO. “Achieving Pareto-optimality through distributed learning”. *Proc. CDC*. 2012 (see p. 64)
- [138] F. MARTIGNON, S. PARIS, I. FILIPPINI, L. CHEN, and A. CAPONE. Efficient and Truthful Bandwidth Allocation in Wireless Mesh Community Networks. *IEEE/ACM Transactions on Networking*, **23**:1 (2015), 161–174 (see p. 137)
- [139] M. J. MCGLYNN and S. A. BORBASH. “Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks”. *Proc. MobiHoc*. 2001 (see p. 28)
- [140] T. MENG, F. WU, and G. CHEN. “On designing neighbor discovery protocols: A code-based approach”. *Proc. INFOCOM*. 2014 (see p. 29)
- [141] P. MERTIKOPOULOS and A. MOUSTAKAS. “Learning in the presence of noise”. *Proc. GameNets*. 2009 (see p. 64)
- [142] F. MIAO, S. HAN, S. LIN, and G. J. PAPPAS. “Taxi Dispatch under Model Uncertainties.” *Proc. CDC*. 2015 (see p. 94)
- [143] K. MICHIHIRO, M. G. J., and R. RAFAEL. Learning, Mutation, and Long Run Equilibria in Games. *Econometrica*, **61**:1 (1993), 29–56 (see p. 64)
- [144] K. MICHIHIRO and R. RAFAEL. Evolution of Equilibria in the Long Run: A General Theory and Applications. *Journal of Economic Theory*, **65**:2 (1995), 383–414 (see p. 64)
- [145] I. MILCHTAICH. Congestion Games with Player-Specific Payoff Functions. *Games and Economic Behavior*, **13**: (1996), 111–124 (see pp. 74, 78)
- [146] D. MONDERER and L. S. SHAPLEY. Potential Games. *Games and Economic Behavior*, **14**:1 (1996), 124–143 (see p. 79)
- [147] T. MOROZAN. “Boundedness properties for stochastic systems”. *Proc. Stability of Stochastic Dynamical Systems*. 1972 (see p. 116)
- [148] S. MURUGESAN, P. SCHNITER, and N. B. SHROFF. Multi-user scheduling in markov-modeled downlink using randomly delayed ARQ feedback. *IEEE Transactions on Information theory*, **2**:58 (2012), 1025 –1042 (see pp. 41, 44)
- [149] R. B. MYERSON. *Game Theory: Analysis of Conflict*. Harvard Univ. Press, 1991 (see p. 66)
- [150] V. NAWARE, G. MERGEN, and L. TONG. Stability and delay of finite-user slotted Aloha with multipacket reception. *IEEE Transactions on Information Theory*, **51**:7 (2005), 2636–2656 (see p. 109)
- [151] P. NI, H. T. VO, D. DAHLMEIER, W. CAI, J. IVANCHEV, and H. AYDT. “DEPART: Dynamic Route Planning in Stochastic Time-Dependent Public Transit Networks”. *Proc. ITSC*. 2015 (see p. 94)
- [152] I. NIVEN, H. S. ZUCKERMAN, and H. L. MONGOMERY. *An Introduction to the Theory of Numbers*. John Wiley & Sons, 1991 (see p. 32)
- [153] H.-J. NOH, J.-K. LEE, and J.-S. LIM. ANC-Aloha: Analog network coding Aloha for satellite networks. *IEEE Communications Letters*, **18**:6 (2014), 957–960 (see p. 108)
- [154] H. OKADA, Y. IGARASHI, and Y. NAKANISHI. Analysis and application of framed Aloha channel in satellite packet switching networks-FADRA method. *Electronics and Communications in Japan*, **60**: (1977), 72–80 (see p. 108)

-
- [155] S. PARIS, F. MARTIGNON, I. FILIPPINI, and L. CHEN. “A bandwidth trading marketplace for mobile data offloading”. *Proc. INFOCOM*. 2013 (see p. 137)
- [156] S. PARIS, F. MARTIGNON, I. FILIPPINI, and L. CHEN. An Efficient Auction-based Mechanism for Mobile Data Offloading. *IEEE Transactions on Mobile Computing*, **14**:8 (2015), 1573–1586 (see p. 137)
- [157] Y. PENG, Z. LI, W. ZHANG, and D. QIAO. “Prolonging sensor network lifetime through wireless charging”. *Proc. RTSS*. 2010 (see p. 94)
- [158] B. S. PRADELSKI and H. P. YOUNG. Learning efficient Nash equilibria in distributed systems. *Games and Economic Behavior*, **75**:2 (2012), 882–897 (see p. 64)
- [159] Z. G. PRODANOFF. Optimal frame size analysis for framed slotted Aloha based rfid networks. *Computer Communications*, **33**:5 (2010), 648–653 (see p. 109)
- [160] C. QIAN, H. NGAN, Y. LIU, and L. M. NI. Cardinality estimation for large-scale RFID systems. *IEEE Transactions on Parallel and Distributed Systems*, **22**:9 (2011), 1441–1454 (see pp. 107, 113)
- [161] H. QIN and W. ZHANG. “Charging Scheduling with Minimal Waiting in a Network of Electric Vehicles and Charging Stations”. *Proc. VANET*. 2011 (see p. 83)
- [162] V. RAGHUNATHAN, V. BORKAR, C. MIN, and P. KUMAR. “Index Policies for Real-Time Multicast Scheduling for Wireless Broadcast Systems”. *Proc. INFOCOM*. 2008 (see p. 44)
- [163] R. R. RAO and A. EPHREMIDES. On the stability of interacting queues in a multiple-access system. *IEEE Transactions on Information Theory*, **34**:5 (1988), 918–930 (see p. 109)
- [164] J. REICH, V. MISRA, D. RUBENSTEIN, and G. ZUSSMAN. Connectivity maintenance in mobile wireless networks via constrained mobility. *IEEE Journal on Selected Areas on Communications*, **30**:5 (2012), 935–950 (see p. 94)
- [165] K. REIF, S. GÜNTHER, E. YAZ SR., and R. UNBEHAUEN. Stochastic stability of the discrete-time extended Kalman filter. *IEEE Transactions on Automatic Control*, **44**:4 (1999), 714–728 (see p. 116)
- [166] M. B. RHUDY and Y. GU. Online Stochastic Convergence Analysis of the Kalman Filter. *International Journal of Stochastic Analysis*, (2013) (see p. 117)
- [167] L. G. ROBERT. Aloha packet system with and without slots and capture. *ACM SIGCOMM Computer Communication Review*, **5**:2 (1975), 28–42 (see p. 108)
- [168] M. ROBERTI. Wal-Mart begins RFID process changes. *RFID Journal*, (2005) (see p. 107)
- [169] L. SAMMUELSON and J. ZHANG. Evolutionary Stability in Asymmetric Games. *Journal of Economic Theory*, **57**:2 (1992), 363–391 (see p. 68)
- [170] W. H. SANDHOLM. Local Stability under Evolutionary Game Dynamics. *Theoretical Economics*, **5**:1 (2010), 27–50 (see p. 67)
- [171] J. SANT and V. SHARMA. Performance analysis of a slotted-Aloha protocol on a capture channel with fading. *Queueing Systems*, **34**:1 (2000), 1–35 (see p. 109)
- [172] V. SARANGAN, M. DEVARAPALLI, and S. RADHAKRISHNAN. A framework for fast RFID tag reading in static and mobile environments. *Computer Networks*, **52**:5 (2008), 1058–1073 (see p. 114)

-
- [173] K. H. SCHLAG. Which One Should I Imitate ? *Journal of Mathematical Economics*, **31**:4 (1999), 493–522 (see p. 68)
- [174] K. H. SCHLAG. Why Imitate, and if so, How ? A Boundedly Rational Approach to Multi-Armed Bandits. *Journal of Economic Theory*, **78**:1 (1998), 130–156 (see p. 66)
- [175] F. C. SCHOUTE. Dynamic frame length Aloha. *IEEE Transactions on Communications*, **31**:4 (1983), 565–568 (see p. 109)
- [176] R. SHAH, S. ROY, S. JAIN, and W. BRUNETTE. “Data MULEs: Modeling a Three-tier Architecture for Sparse Sensor Networks”. *Proc. SNPA Workshop*. 2003 (see p. 82)
- [177] M. SHAHZAD and A. X. LIU. “Every bit counts: fast and scalable RFID estimation”. *Proc. Mobicom*. 2012 (see pp. 107, 113, 127)
- [178] M. SHAHZAD and A. X. LIU. “Expecting the unexpected: Fast and reliable detection of missing RFID tags in the wild”. *Proc. INFOCOM*. 2015 (see pp. 126, 134)
- [179] M. SHAHZAD and A. X. LIU. “Probabilistic optimal tree hopping for RFID identification”. *Proc. SIGMETRICS*. 2013 (see p. 126)
- [180] Y. SONG and J. W. GRIZZLE. “The extended Kalman filter as a local asymptotic observer for nonlinear discrete-time systems”. *Proc. ACC*. 1992 (see pp. 107, 114)
- [181] F. SPIRING. Introduction to Statistical Quality Control. *Technometrics*, **49**:1 (2007), 108–109 (see p. 123)
- [182] R. SUGIHARA and R. K. GUPTA. Speed Control and Scheduling of Data Mules in Sensor Networks. *ACM Transactions on Sensor Networks*, **7**:1 (2010), 1–29 (see pp. 84, 94)
- [183] G. SUN, F. WU, and G. CHEN. “Neighbor Discovery in Low-Duty-Cycle Wireless Sensor Networks with Multipacket Reception”. *Proc. ICPADS*. 2012 (see p. 28)
- [184] W. SUN, Z. YANG, K. WANG, and Y. LIU. “Hello: A generic flexible protocol for neighbor discovery”. *Proc. INFOCOM*. 2014 (see p. 29)
- [185] W. SZPANKOWSKI. Stability conditions for some distributed systems: Buffered random access systems. *Advances in Applied Probability*, **26**:2 (1994), 498–515 (see p. 109)
- [186] A. TA-SHMA and U. ZWICK. “Deterministic rendezvous, treasure hunts and strongly universal exploration sequences”. *Proc. SODA*. 2007 (see p. 16)
- [187] C. C. TAN, B. SHENG, and Q. LI. “How to monitor for missing RFID tags”. *Proc. ICDCS*. 2008 (see p. 126)
- [188] T.-J. TARN and Y. RASIS. Observers for nonlinear stochastic systems. *IEEE Transactions on Automatic Control*, **21**:4 (1976), 441–448 (see pp. 116, 117)
- [189] C. TEKIN and M. LIU. “Online learning in opportunistic spectrum access: a restless bandit approach”. *Proc. INFOCOM*. 2011 (see p. 45)
- [190] N. THEIS, R. THOMAS, and L. DASILVA. Rendezvous for cognitive radios. *IEEE Transactions on Mobile Computing*, **10**:2 (2011), 216–227 (see p. 16)
- [191] W. R. THOMPSON. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika*, **25**: (1933), 275–294 (see p. 43)
- [192] D. TONE. The evolution of conventions with mobile players. *Journal of Economic Behavior & Organization*, **38**:1 (1999), 93–111 (see p. 64)

-
- [193] P. TOTH and D. VIGO, eds. *The Vehicle Routing Problem*. MOS/SIAM Series on Optimization, 2001 (see p. 94)
- [194] Y.-C. TSENG, C.-S. HSU, and T.-Y. HSIEH. “Power-saving protocols for ieee 802.11-based multi-hop ad hoc networks”. *Proc. INFOCOM*. 2002 (see p. 29)
- [195] B. S. TSYBAKOV and V. A. MIKHAILOV. Ergodicity of a slotted Aloha system. *Problemy Peredachi Informatsii*, 15:4 (1979), 73–87 (see p. 109)
- [196] I. VASILESCU, K. KOTAY, D. RUS, M. DUNBABIN, and P. CORKE. “Data Collection, Storage, and Retrieval with an Underwater Sensor Network”. *Proc. SenSys*. 2005 (see p. 82)
- [197] S. VASUDEVAN, D. TOWSLEY, D. GOECKEL, and R. KHALILI. “Neighbor discovery in wireless networks and the coupon collector’s problem”. *Proc. MobiCom*. 2009 (see pp. 28, 108)
- [198] F. VÁZQUEZ GALLEGO, J. ALONSO-ZARATE, and L. ALONSO. “Energy and delay analysis of contention resolution mechanisms for machine-to-machine networks based on low-power WiFi”. *Proc. ICC*. 2013 (see p. 108)
- [199] P.-J. WAN, K. ALZOUBI, and O. FRIEDER. “Distributed construction of connected dominating set in wireless ad hoc networks”. *Proc. INFOCOM*. 2002 (see p. 91)
- [200] K. WANG and L. CHEN. On Optimality of Myopic Policy for Restless Multi-Armed Bandit Problem: An Axiomatic Approach. *IEEE Transactions on Signal Processing*, 60:1 (2012), 300–309 (see pp. 9, 42, 51)
- [201] K. WANG, L. CHEN, and Q. LIU. On Optimality of Myopic Policy for Opportunistic Access With Nonidentical Channels and Imperfect Sensing. *IEEE Transactions on Vehicular Technology*, 63:5 (2014), 2478–2483 (see pp. 9, 42, 48)
- [202] K. WANG, L. CHEN, and Q. LIU. Opportunistic Spectrum Access by Exploiting Primary User Feedbacks in Underlay Cognitive Radio Systems: An Optimality Analysis. *IEEE Journal of Selected Topics in Signal Processing*, 7:5 (2013), 869–882 (see pp. 9, 42)
- [203] K. WANG, L. CHEN, and J. YU. “On optimality of myopic policy in multi-channel opportunistic access”. *Proc. ICC*. 2016 (see pp. 9, 42)
- [204] K. WANG, F. C. M. LAU, L. CHEN, and R. SCHOBER. “A distributed market framework for mobile data offloading”. *Proc. ICC*. 2015 (see p. 137)
- [205] K. WANG, L. CHEN, Q. LIU, J. YU, Q. FAN, and Q. AI. On Optimality of Myopic Policy in Multi-channel Opportunistic Access. *IEEE Transactions on Communications (to appear)*, (2016) (see pp. 9, 42)
- [206] K. WANG, L. CHEN, Q. LIU, and K. A. AGHA. On Optimality of Myopic Sensing Policy with Imperfect Sensing in Multi-Channel Opportunistic Access. *IEEE Transactions on Communications*, 61:9 (2013), 3854–3862 (see pp. 9, 42, 51)
- [207] K. WANG, L. CHEN, Q. LIU, W. WANG, and F. LI. One Step Beyond Myopic Probing Policy: A Heuristic Lookahead Policy for Multi-Channel Opportunistic Access. *IEEE Transactions on Wireless Communications*, 14:2 (2015), 759–769 (see pp. 9, 42, 57, 59)
- [208] K. WANG, F. C. M. LAU, L. CHEN, and R. SCHOBER. Pricing Mobile Data Offloading: A Distributed Market Framework. *IEEE Transactions on Wireless Communications*, 15:2 (2016), 913–927 (see p. 137)

-
- [209] K. WANG, L. CHEN, K. A. AGHA, and Q. LIU. On Optimality of Myopic Policy in Opportunistic Spectrum Access: The Case of Sensing Multiple Channels and Accessing One Channel. *IEEE Wireless Communications Letters*, **1**:5 (2012), 452–455 (see pp. 47, 48, 52)
- [210] K. WANG, X. MAO, and Y. LIU. “BlindDate: A Neighbor Discovery Protocol”. *Proc. ICPP*. 2013 (see p. 29)
- [211] R. WEBER. Optimal Symmetric Rendezvous Search on Three Locations. *Mathematics of Operations Research*, **37**:1 (2012), 111–122 (see p. 16)
- [212] P. WHITTLE. Restless bandits: activity allocation in a changing world. *Journal of Applied Probability*, Special Vol. 25A (1988), 287–298 (see pp. 43, 44)
- [213] J. E. WIESELTHIER, A. EPHREMIDES, and A LARRY. An exact analysis and performance evaluation of framed Aloha with capture. *IEEE Transactions on Communications*, **37**:2 (1989), 125–137 (see p. 109)
- [214] C. WU and S. WU. “On bridging the gap between homogeneous and heterogeneous rendezvous schemes for cognitive radios”. *Proc. MobiHoc*. 2013 (see p. 16)
- [215] H. WU, C. ZHU, R. J. LA, and X. LIU. FASA: Accelerated S-Aloha Using Access History for Event-Driven M2M Communications. *IEEE/ACM Transactions on Networking*, **21**:6 (2013), 1904–1917 (see p. 108)
- [216] Q. XIAO, B. XIAO, and S. CHEN. “Differential estimation in dynamic RFID systems”. *Proc. INFOCOM*. 2013 (see p. 114)
- [217] Q. XIAO, M. C. ZHOU, S. CHEN, and Y. QIAO. “Temporally or Spatially Dispersed Joint RFID Estimation Using Snapshots of Variable Lengths”. *ACM MobiHoc*. 2015 (see p. 114)
- [218] L. XIE, Y. SHI, Y. T. HOU, and H. D. SHERALI. Making Sensor Networks Immortal: An Energy-Renewal Approach With Wireless Power Transfer. *IEEE/ACM Transactions on Networking*, **20**:6 (2012), 1748–1761 (see p. 84)
- [219] L. XIE, B. SHENG, C. C. TAN, H. HAN, Q. LI, and D. CHEN. “Efficient tag identification in mobile RFID systems”. *Proc. INFOCOM*. 2010 (see p. 114)
- [220] L. XIE, Y. SHI, Y. T. HOU, W. LOU, H. D. SHERALI, H. ZHOU, and S. F. MIDKIFF. A Mobile Platform for Wireless Charging and Data Collection in Sensor Networks. *IEEE Journal of Selected Areas on Communications*, **33**:8 (2015), 1521–1533 (see p. 83)
- [221] L. XIE, Y. SHI, Y. T. HOU, W. LOU, and H. D. SHERALI. “On Traveling Path and Related Problems for a Mobile Station in a Rechargeable Sensor Network”. *Proc. MobiHoc*. 2013 (see pp. 84, 94)
- [222] G. XING, T. WANG, W. JIA, and M. LI. “Rendezvous Design Algorithms for WSNs with a Mobile Base Station”. *Proc. MobiHoc*. 2008 (see pp. 84, 94)
- [223] L. XUE, D. KIM, Y. ZHU, D. LI, W. WANG, and A. O. TOKUTA. “Multiple heterogeneous data ferry trajectory planning in WSNs”. *Proc. INFOCOM*. 2014 (see pp. 82, 84)
- [224] J. YANG, P. JAILLET, and H. MAHMASSANI. Real-Time Multivehicle Truckload Pickup and Delivery Problems. *Transportation Science*, **38**: (2004), 135–148 (see p. 94)
- [225] H. P. YOUNG. The Evolution of Conventions. *Econometrica*, **61**:1 (1993), 57–84 (see pp. 64, 76, 79)
- [226] H. P. YOUNG. Learning by trial and error. *Games and Economic Behavior*, **65**:2 (2009), 626–643 (see p. 64)

-
- [227] J. YU and L. CHEN. “Stability Analysis of Frame Slotted Aloha Protocol”. *Proc. IWQoS (extended version to appear in IEEE Transactions on Mobile Computing)*. 2016 (see pp. 10, 11, 108, 110, 111)
- [228] J. YU, L. CHEN, R. ZHANG, and K. WANG. Finding Needles in a Haystack: Missing Tag Detection in Large RFID Systems. *IEEE Transactions on Communications (to appear)*, (2017) (see pp. 10, 11, 108, 134)
- [229] J. YU, L. CHEN, R. ZHANG, and K. WANG. From Static to Dynamic Tag Population Estimation: An Extended Kalman Filter Perspective. *IEEE Transactions on Communications*, **64**:11 (2016), 4706–4719 (see pp. 10, 11, 108, 125, 126)
- [230] J. YU, L. CHEN, R. ZHANG, and K. WANG. On Missing Tag Detection in Multiple-group Multiple-region RFID Systems. *IEEE Transactions on Mobile Computing (to appear)*, (2016) (see pp. 10, 11, 108)
- [231] B. YUAN, M. ORLOWSKA, and S. SADIQ. On the Optimal Robot Routing Problem in WSNs. *IEEE Transactions on Knowledge and Data Engineering*, **19**:9 (2007), 1252–1261 (see p. 84)
- [232] W. ZENG, S. VASUDEVAN, X. CHEN, B. WANG, A. RUSSELL, and W. WEI. “Neighbor discovery in wireless networks with multipacket reception”. *Proc. MobiHoc*. 2011 (see pp. 28, 108)
- [233] D. ZHANG, T. HE, S. LIN, S. MUNIR, and J. STANKOVIC. Online Cruising Mile Reduction in Large-Scale Taxicab Networks. *IEEE Transactions on Parallel and Distributed Systems*, **26**:11 (2015), 3122–3135 (see p. 94)
- [234] R. ZHANG, Y. LIU, Y. ZHANG, and J. SUN. “Fast identification of the missing tags in a large RFID system”. *Proc. SECON*. 2011 (see p. 126)
- [235] S. ZHANG, J. WU, and S. LU. Collaborative Mobile Charging. *IEEE Transactions on Computers*, **64**:3 (2015), 654–667 (see pp. 84, 94)
- [236] Y. ZHANG, Q. LI, G. YU, and B. WANG. “Etch: efficient channel hopping for communication rendezvous in dynamic spectrum access networks”. *Proc. INFOCOM*. 2011 (see p. 16)
- [237] Q. ZHAO, B. KRISHNAMACHARI, and K. LIU. On Myopic Sensing for Multi-Channel Opportunistic Access: Structure, Optimality, and Performance. *IEEE Transactions Wireless Communication*, **7**:3 (2008), 5413–5440 (see pp. 41, 44, 47)
- [238] R. ZHENG, J. C. HOU, and L. SHA. “Asynchronous wakeup for ad hoc networks”. *Proc. MobiHoc*. 2003 (see p. 29)
- [239] Y. ZHENG and M. LI. “ZOE: Fast cardinality estimation for large-scale RFID systems”. *Proc. INFOCOM*. 2013 (see pp. 107, 113, 127)
- [240] M. ZHU and S. MARTÍNEZ. Distributed Coverage Games for Energy-Aware Mobile Sensor Networks. *SIAM Journal on Control and Optimization*, **51**:1 (2013), 1–27 (see p. 64)
- [241] Y. ZHU, W. JIANG, Q. ZHANG, and H. GUAN. Energy-efficient identification in large-scale RFID systems with handheld reader. *IEEE Transactions on Parallel and Distributed Systems*, **25**:5 (2014), 1211–1222 (see p. 108)

Publications

Articles in Peer-reviewed Journals

1. D. Zhang, Q. Quan, **L. Chen**, W. Xu, K. Wang, On-demand Ecology-inspired Spectrum Allocation for Heterogeneous CRNs, accepted in *Telecommunication Systems (TELS)*, 2017.
2. J. Yu, **L. Chen**, R. Zhang, K. Wang, Finding Needles in a Haystack: Missing Tag Detection in Large RFID Systems, accepted in *IEEE Transactions on Communications (TCOM)*, 2017.
3. **L. Chen**, Y. Li, A. V. Vasilakos, On Oblivious Neighbor Discovery in Distributed Wireless Networks with Directional Antennas: Theoretical Foundation and Algorithm Design, accepted in *IEEE/ACM Transactions on Networking (ToN)*, 2017.
4. M. Zheng, **L. Chen**, W. Liang, H. Yu, J. Wu, Energy-efficiency Maximization for Cooperative Spectrum Sensing in Cognitive Sensor Networks, accepted in *IEEE Transactions on Green Communications and Networking (TGCN)*, 2017.
5. M. Koseoglu, E. Karasan, **L. Chen**, Cross-layer Energy Minimization for Underwater Aloha Networks, accepted in *IEEE Systems Journal (ISJ)*, 2017.
6. D. Zhang, Q. Liu, **L. Chen**, K. Wang, Multi-layer Based Multi-path Routing Algorithm for Maximizing Spectrum Availability, accepted in *Springer Wireless Networks (WINET)*, 2017.
7. X. Zhou, W. Wang, Y. Wang, **L. Chen**, Z. Zhang, Moderate Incentive Design for Delay-constrained D2D Relaying, accepted in *ACM/Springer Mobile Networks and Applications (MONET)*, 2017.
8. J. Elias, F. Martignon, **L. Chen**, M. Krunz, Distributed Spectrum Management in TV White Space Networks, accepted in *IEEE Transactions on Vehicular Technology (TVT)*, 2017.
9. J. Yu, **L. Chen**, R. Zhang, K. Wang, On Missing Tag Detection in Multiple-group Multiple-region RFID Systems, accepted in *IEEE Transactions on Mobile Computing (TMC)*, 2017.
10. J. Yu, **L. Chen**, Stability Analysis of Frame Slotted Aloha Protocol, accepted in *IEEE Transactions on Mobile Computing (TMC)*, 2017.
11. C. Rottondi, A. Barbato, **L. Chen**, G. Verticale, Enabling Privacy in a Distributed Game-theoretical Scheduling System for Domestic Appliances, *IEEE Transactions on Smart Grids (TSG)*, 8(3):1220–1230, May 2017.
12. K. Wang, **L. Chen**, J. Yu, On Optimality of Myopic Policy in Multi-channel Opportunistic Access, *IEEE Transactions on Communications (TCOM)*, 65(2):677–690, February 2017.
13. M. Zheng, C. Xu, W. Liang, H. Yu, **L. Chen**, Time-efficient Cooperative Spectrum Sensing via Analog Computation over Multiple-access Channel, *Elsevier Computer Networks (ComNet)*, 112:84–94, January 2017.
14. J. Yu, **L. Chen**, R. Zhang, K. Wang, From Static to Dynamic Tag Population Estimation: An Extended Kalman Filter Perspective, *IEEE Transactions on Communications (TCOM)*, 64(11):4706–4719, November 2016.
15. **L. Chen**, K. Bian, Neighbor Discovery in Mobile Sensing Applications: A Comprehensive

- Survey, *Elsevier Ad Hoc Networks*, 48(9):38–52, September 2016.
16. Z. Ismail, C. Kiennert, J. Leneutre, D. Bateman, **L. Chen**, Auditing a Cloud Provider’s Compliance with Data Backup Requirements: A Game Theoretical Analysis, *IEEE Transactions on Information Forensics and Security (TIFS)*, 11(8):1685–1699, August 2016.
 17. **L. Chen**, W. Wang, H. Huang, S. Lin, On Time-constrained Data Harvesting in WSNs: Approximation Algorithm Design, *IEEE/ACM Transactions on Networking (ToN)*, 24(5):3123–3135, October 2016.
 18. **L. Chen**, K. Bian, M. Zheng, Never Live without Neighbors: From Single- to Multi-channel Neighbor Discovery for Mobile Sensing Applications, *IEEE/ACM Transactions on Networking (ToN)*, 24(5):3123–3135, October 2016.
 19. Y. Lu, W. Wang, **L. Chen**, N. Ming, Z. Zhang, A. Huang, Distance-based Energy-efficient Opportunistic Broadcast Forwarding in Mobile DTNs, *IEEE Transactions on Vehicular Technology (TVT)*, 65(7):5512–5524, July 2016.
 20. K. Wang, **L. Chen**, F. Liu, J. Yu, Optimality of Myopic Policy for Multi-state Channel Access, *IEEE Communications Letters (COMML)*, 20(2):300–303, February 2016.
 21. K. Wang, F. Liu, **L. Chen**, R. Schober, Pricing Mobile Data Offloading: A Distributed Market Framework, *IEEE Transactions on Wireless Communications (TWC)*, 15(2):913–927, February 2016.
 22. W. Wang, L. Wu, Z. Zhang, **L. Chen**, Joint Spectrum Sensing and Access for Stable Dynamic Spectrum Aggregation, *EURASIP Journal on Wireless Communications and Networking (JWCN)*, December 2015.
 23. W. Wang, **L. Chen**, K. G. Shin, L. Duan, Thwarting Intelligent Malicious Behaviors in Cooperative Spectrum Sensing, *IEEE Transactions on Mobile Computing (TMC)*, 14(11):2392–2405, November 2015.
 24. **L. Chen**, K. Bian, The Telephone Coordination Game Revisited: From Random to Deterministic Algorithms, *IEEE Transactions on Computers (TC)*, 64(10):2968–2980, October 2015.
 25. S. D’oro, L. Galluccio, G. Morabito, S. Palazzo, **L. Chen**, F. Martignon, Defeating Jamming with the Power of Silence: A Game-theoretic Analysis, *IEEE Transactions on Wireless Communications (TWC)*, 14(5):2337–2352, May 2015, **Newom# project Best Student Paper**.
 26. K. Wang, Q. Liu, F. Li, **L. Chen**, X. Ma, Myopic Policy for Opportunistic Access in CRNs by Exploiting Primary User Feedbacks, *IET Communications*, 9(7):1017–1025, May 2015.
 27. S. Iellamo, E. Alekseeva, **L. Chen**, M. Coupechoux and Y.A. Kochetov, Competitive Location in CRNs, *A Quarterly Journal of Operations Research (4QR)*, 13(1):81–110, March 2015.
 28. F. Martignon, S. Paris, I. Filippini, **L. Chen**, A. Capone, Efficient and Truthful Bandwidth Allocation in Wireless Mesh Community Networks, *IEEE/ACM Transactions on Networking (ToN)*, 23(1):161–174, February 2015.
 29. S. Paris, F. Martignon, I. Filippini, **L. Chen**, An Efficient Auction-based Mechanism for Mobile Data Offloading, *IEEE Transactions on Mobile Computing (TMC)*, 14(8):1573–1586, August 2015.
 30. A. Barbato, A. Capone, **L. Chen**, F. Martignon, S. Paris, A Distributed Demand-side Management Framework for the Smart Grid, *Elsevier Computer Communications (ComCom)*, 57(2):13–24, February 2015.
 31. K. Wang, **L. Chen**, Q. Liu, W. Wang, One Step Beyond Myopic Probing Policy: A Heuris-

- tic Lookahead Policy for Multi-channel Opportunistic Access, *IEEE Transactions on Wireless Communications (TWC)*, 14(2):759–769, February 2015.
32. Z. Ismail, J. Leneutre, D. Bateman, **L. Chen**, A Game Theoretical Analysis of Data Confidentiality Attacks on Smart Grid AMI, *IEEE Journal on Selected Areas in Communications (JSAC)*, 32(7):1486–1499, August 2014.
 33. K. Wang, **L. Chen**, Q. Liu, On Optimality of Myopic Policy for Opportunistic Access with Non-identical Channels and Imperfect Sensing, *IEEE Transactions on Vehicular Technology (TVT)*, 63(5):2478–2483, June 2014.
 34. M. Youssef, M. Ibrahim, M. Abdelatif, **L. Chen**, A. V. Vasilakos, Routing Metrics of CRNs: A Survey, *IEEE Communications Surveys and Tutorials (CST)*, 14(1):92–109, February 2014, **ESI Highly Cited Paper: total number of citations 200+**.
 35. J. Elias, F. Martignon, **L. Chen**, E. Altman, Joint Operator Pricing and Network Selection Game in CRNs: Equilibrium, System Dynamics and Price of Anarchy, *IEEE Transactions on Vehicular Technology (TVT)*, 62(9):1–14, November 2013.
 36. K. Wang, **L. Chen**, Q. Liu, K. Al Agha, On Optimality of Myopic Sensing Policy with Imperfect Sensing in Multi-channel Opportunistic Access, *IEEE Transactions on Communications (TCOM)*, 61(9):3854–3862, September 2013.
 37. K. Wang, **L. Chen**, Q. Liu, Opportunistic Spectrum Access by Exploiting Primary User Feedbacks in Underlay CRNs: An Optimality Analysis, *IEEE Journal of Selected Topics in Signal Processing (JSTSP)*, 7(5):869–882, October 2013.
 38. S. Iellamo, **L. Chen** and M. Coupechoux, Proportional and Double Imitation Rules for Spectrum Access in CRNs, *Elsevier Computer Networks (ComNet)*, 57(8):1863–1879, June 2013.
 39. **L. Chen**, W. Wang, A. S. Anpalagan, A. V. Vasilakos, K. Illanko, H. Wang, M. Naeem, Green Cooperative Cognitive Communication and Networking: A New Paradigm for Wireless Networks, *ACM/Springer Mobile Networks and Applications (MONET)*, 18(4):524–534, May 2013.
 40. K. Wang, Q. Liu, **L. Chen**, Hierarchical Reversible Data Hiding Based on Statistical Information: Preventing Embedding Unbalance, *Elsevier Signal Processing (SigPro)*, 92(12):2888–2900, December 2012.
 41. K. Wang, **L. Chen**, K. Al Agha, Q. Liu, On Optimality of Myopic Policy in Opportunistic Spectrum Access: The Case of Sensing Multiple Channels and Accessing One Channel, *IEEE Wireless Communications Letters (WCL)*, 1(5):452–455, October 2012.
 42. K. Wang, Q. Liu, **L. Chen**, Optimality of Greedy Policy for a Class of Standard Reward Function of Restless Multi-armed Bandit Problem, *IET Signal Processing*, 6(6):584–593, August 2012.
 43. M. A. Awal, L. Boukhatem, **L. Chen**, An Integrated Cross-layer Framework of Adaptive Feedback Resource Allocation and Prediction for OFDMA Systems, *Elsevier Computer Networks (ComNet)*, 56(7), pp. 1863–1875, May 2012.
 44. K. Wang, **L. Chen**, On Optimality of Myopic Policy for Restless Multi-armed Bandit Problem: An Axiomatic Approach, *IEEE Transactions on Signal Processing (TSP)*, 60(1):300–309, January 2012.
 45. **L. Chen**, J. Leneutre, Fight Jamming with Jamming: A Game Theoretic Analysis of Jamming Attack in Wireless Networks and Defense Strategy, *Elsevier Computer Networks (ComNet)*, 55(9):2259–2270, June 2011.
 46. **L. Chen**, S. Iellamo, M. Coupechoux, Ph. Godlewski, Spectrum Auction with Interference

- Constraint for CRNs with Multiple Primary and Secondary Users, *Springer Wireless Networks (WINET)*, 17(5):1355–1371, May 2011.
47. **L. Chen**, L. Libman, J. Leneutre, Conflicts and Incentives in Wireless Cooperative Relaying: A Distributed Market Pricing Framework, *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 22(5):758–772, May 2011.
 48. **L. Chen**, J. Leneutre, A Game Theoretic Framework of Intrusion Detection in Heterogeneous Networks, *IEEE Transactions on Information Forensics and Security (TIFS)*, 4(2):165–178, June 2009.
 49. **L. Chen**, J. Leneutre, On Multipath Routing in Multihop Wireless Networks: Security, Performance and Their Tradeoff, *EURASIP Journal on Wireless Communications and Networking (JWCN)*, Vol. 2009, Article ID 946493.
 50. **L. Chen**, J. Leneutre, A Game Theoretic Framework of Distributed Power and Rate Control in IEEE 802.11 WLANs, *IEEE Journal on Selected Areas in Communications (JSAC)*, 26(7):1128–1137, September 2008.
 51. **L. Chen**, J. Leneutre, Toward Secure and Scalable Time Synchronization in Ad Hoc Networks, *Elsevier Computer Communication (ComCom)*, 30(11–12):2453–2467, September 2007.
 52. X. Xue, J. Leneutre, **L. Chen**, J. Ben-Othman, SWAN: A Secured Watchdog for Ad Hoc Networks, *International Journal of Computer Science and Network Security (IJCSNS)*, 6(2):209–218, June 2006

Articles in Peer-reviewed Conferences

1. Z. Chen, **L. Chen**, H. Zhong, Towards Secure and Verifiable Database-driven Spectrum Sharing, *IEEE/IFIP DSN 2017*, accepted.
2. J. Yu, J. Liu, **L. Chen**, Y. Zhu, Efficient Group Labeling for Multi-group RFID Systems, extended abstract, *ACM/IEEE IWQoS*, 2015, accepted.
3. R. Zhang, H. Moun gla, J. Yu, **L. Chen**, A. Mehaoua, Multi-channel Broadcast in Asymmetric Duty Cycling Wireless Body Area Networks, *IEEE ICC 2017*, accepted.
4. M. Morcos, T. Chaled, **L. Chen**, J. Elias, F. Martignon, A Two-level Auction for C-RAN Resource Allocation, *MT5GNet, IEEE ICC Workshop 2017*, accepted
5. Y. Wang, W. Wang, **L. Chen**, Z. Zhang, Lexicographic Relay Selection and Channel Allocation for Multichannel Cooperative Multicast, *IEEE WCNC*, San Francisco, CA, USA, March 2017.
6. Z. Ismail, J. Leneutre, D. Bateman, **L. Chen**, A Methodology to Apply a Game Theoretic Model of Security Risks Interdependencies Between ICT and Electric Infrastructures, *GameSec*, New York, USA November 2016.
7. Y. Wang, W. Wang, **L. Chen**, Z. Zhang, Energy Efficient Scheduling for Delay-constrained Spectrum Aggregation, *IEEE Globecom*, Washington, USA, December 2016.
8. Z. Chen, **L. Chen**, L. Huang, Z. Zhong, On Privacy-preserving Cloud Auction, *IEEE SRDS*, Budapest, Hungary, September 2016.
9. **L. Chen**, S. Lin, H. Huang, Charge Me If You Can: Charging Path Optimization and Scheduling in Mobile Networks, *ACM MobiHoc*, Paderborn, Germany, July 2016.
10. Z. Chen, **L. Chen**, L. Huang, H. Zhong, Towards Secure Spectrum Auction: Both Bids and Bidder Locations Matter, extended abstract, *ACM MobiHoc*, Paderborn, Germany, July 2016.
11. K. Wang, **L. Chen**, J. Yu, On Optimality of Myopic Policy in Multichannel Opportunistic Access,

- IEEE ICC*, Kuala Lumpur, Malaysia, May 2016.
12. **L. Chen**, Y. Li, A. V. Vasilakos, Oblivious Neighbor Discovery for Wireless Devices with Directional Antennas, *IEEE INFOCOM*, San Francisco, USA, April 2016.
 13. D. Zhang, **L. Chen**, Q. Liu, W. Xu, Ecology-based Coexistence Mechanism in Heterogeneous CRNs, *IEEE Globecom*, San Diego, USA, December 2015.
 14. H. Huang, S. Lin, **L. Chen**, J. Gao, A. Mamat and J. Wu, Dynamic Mobile Charger Scheduling in Heterogeneous WSNs, *IEEE MASS*, Dallas, USA, October 2015.
 15. J. Yu, **L. Chen**, Stability Analysis of Frame Slotted Aloha Protocol, *ACM/IEEE IWQoS*, Portland, USA, June 2015.
 16. **L. Chen**, W. Wang, H. Huang and S. Lin, Time-constrained Data Harvesting in WSNs: Theoretical Foundation and Algorithm Design, *IEEE INFOCOM*, Hong Kong, China, May 2015.
 17. Z. Chen, L. Huang, **L. Chen**, ITSEC: An Information-theoretically Secure Framework for Truthful Spectrum Auctions, *IEEE INFOCOM*, Hong Kong, China, May 2015.
 18. L. Chen, R. Fan, K. Bian, **L. Chen**, M. Gerla, T. Wang, X. Li, On Heterogeneous Neighbor Discovery in WSNs, *IEEE INFOCOM*, Hong Kong, China, May 2015.
 19. K. Wang, F. C. M. Lau, **L. Chen**, R. Schober, A Distributed Market Framework for Mobile Data Offloading, *IEEE ICC*, London, UK, June 2015.
 20. A. Barbato, A. Capone, **L. Chen**, F. Martignon, S. Paris, Distributed Demand-side Management in Smart Grid: How Imitation improves Power Scheduling, *IEEE ICC*, London, UK, June 2015.
 21. M. Zheng, C. Xu, W. Liang, H. Yu, **L. Chen**, A Novel CoMAC-based Cooperative Spectrum Sensing Scheme in CRNs, *COCONET, IEEE ICC workshop*, London, UK, June 2015.
 22. Z. Ismail, J. Leneutre, D. Bateman, L. Chen, A Game-theoretical Model for Security Risk Management of Interdependent ICT and Electrical Infrastructures, *IEEE HASE*, Daytona Beach, USA, January 2015.
 23. B. Djebaili, C. Kiennert, J. Leneutre, **L. Chen**, Data Integrity and Availability Verification Game in Untrusted Cloud Storage, *GameSec*, Los Angeles, USA, November 2014.
 24. A. Barbato, **L. Chen**, F. Martignon, S. Paris, A Multi-armed Bandit Formulation for Distributed Appliances Scheduling in Smart Grids, *IEEE Online GreenComm*, October 2014.
 25. A. Barbato, A. Capone, **L. Chen**, F. Martignon, S. Paris, Distributed Learning Algorithms for Scheduling Games in the Future Smart Grid, *NetGCoop*, Trento, Italy, October 2014.
 26. **L. Chen**, K. Bian, **L. Chen**, C. Liu, J.-M. Park, X. Li, A Group-theoretic Framework for Rendezvous in Heterogeneous CRNs, *ACM MobiHoc*, Philadelphia, USA, August 2014.
 27. **L. Chen**, K. Bian, M. Zheng, Heterogeneous Multi-channel Neighbor Discovery for Mobile Sensing Applications: Theoretical Foundation and Protocol Design, *ACM MobiHoc*, Philadelphia, USA, August 2014.
 28. W. Wang, **L. Chen**, K. G. Shin, L. Duan, Secure Cooperative Spectrum Sensing and Access Against Intelligent Malicious Behaviors, *IEEE INFOCOM*, Toronto, Canada, May 2014.
 29. S. Iellamo, **L. Chen**, M. Coupechoux, Retrospective Spectrum Access Protocol: A Payoff-based Learning Algorithm for CRNs, *IEEE ICC*, Sydney, Australia, June 2014.
 30. Y. Lu, W. Wang, **L. Chen**, Z. Zhang, A. Huang, Distance-based Energy-efficient Opportunistic Forwarding in Mobile DTNs, *IEEE ICC*, Sydney, Australia, June 2014.
 31. Y. Lu, W. Wang, **L. Chen**, Z. Zhang, A. Huang, Opportunistic Forwarding in Energy Harvesting

- Mobile DTNs, *IEEE ICC*, Sydney, Australia, June 2014.
32. A. Barbato, A. Capone, **L. Chen**, F. Martignon, S. Paris, A Power Scheduling Game for Reducing the Peak Demand of Residential Users, *IEEE Online GreenComm*, October 2013.
 33. L. Chen, K. Bian, **L. Chen**, W. Yan, X. Li. On the Cascading Spectrum Contention Problem in Self-coexistence of CRNs. *CRAB, ACM MobiCom workshop*, Miami, USA, September 2013.
 34. S. Paris, F. Martignon, I. Filippini, **L. Chen**, A Bandwidth Trading Marketplace for Mobile Data Offloading, *IEEE INFOCOM (Mini-conference)*, Turin, Italy, April 2013.
 35. S. Iellamo, E. Alekseeva, **L. Chen**, M. Coupechoux, Y. A. Kochetov, A Bi-level Competitive Facility Location Game for the Design of CRNs, *European Conference on Operations Research (EURO MMXIII)*, Rome, Italy, July 2013.
 36. L. Wu, W. Wang, Z. Zang, **L. Chen**, A Rollout-based Joint Spectrum Sensing and Access Policy for CRNs with Hardware Limitations, *IEEE Globecom*, CA, USA, December 2012.
 37. S. Iellamo, **L. Chen**, M. Coupechoux, Y. A. Kochetov, Strategic Planning in CRNs: A Competitive Facility Location Game Perspective, *All-Russia Conference on Optimization Problems and their Economical Applications*, Omsk, Russia, July 2012.
 38. S. Iellamo, **L. Chen**, M. Coupechoux, A. V. Vasilakos, Imitation-based Spectrum Access Policy for CRNs, Invited paper, *ISWCS*, Paris, France, August 2012.
 39. S. Iellamo, **L. Chen**, M. Coupechoux, Imitation-based Spectrum Access Policy for CSMA/CA-based CRNs, *IEEE WCNC*, Paris, France, April 2012.
 40. **L. Chen**, S. Iellamo, M. Coupechoux, Opportunistic Spectrum Access with Channel Switching Cost for CRNs, *IEEE ICC*, Kyoto, Japan, June 2011.
 41. **L. Chen**, S. Iellamo, M. Coupechoux, Ph. Godlewski, An Auction Framework for Spectrum Allocation with Interference Constraint in CRNs, *IEEE INFOCOM*, San Diego, USA, March 2010.
 42. **L. Chen**, A Distributed Access Point Selection Algorithm Based on No-regret Learning for Wireless Access Networks, *IEEE VTC*, Taipei, China, May 2010.
 43. **L. Chen**, J. Leneutre, Efficient Medium Access Control Design: A Game Theoretical Approach, *IEEE LCN*, Zurich, Suisse, October 2009.
 44. **L. Chen**, J. Leneutre, Selfishness, not always a Nightmare: A Game Theoretic Modelling of Selfish Behaviours in Mobile Ad Hoc Networks, *IEEE ICDCS*, Toronto, Canada, July 2007.
 45. **L. Chen**, J. Leneutre, On the Rate and Power Control in IEEE 802.11 WLANs: A Game Theoretic Approach, *IEEE ICCCN*, Hawaii, USA, August 2007, **Best Paper Award (1 out of 500+ submissions)**.
 46. **L. Chen**, J. Leneutre, A Game Theoretic Framework of Distributed Power and Rate Control in IEEE 802.11 WLANs, extended abstract, *IEEE ICNP*, Beijing China, October 2007.
 47. **L. Chen**, K. Almoubayed, J. Leneutre, Detection and Prevention of Greedy Behavior in Ad Hoc Networks, *CRISIS*, Marrakech, Maroc, August 2007.
 48. **L. Chen**, J. Leneutre, J-J. Puig, A Secure and Efficient Link-state Routing Protocol for Ad Hoc Networks, *ICWMC*, Bucharest, Romania, July 2006.
 49. **L. Chen**, X. Xue, J. Leneutre, A Lightweight Mechanism to Secure OLSR, *IMECS*, Hong Kong, China, June 2006.