Distributed opportunistic scheduling algorithms for wireless communications.

To, Toan

Distributed Opportunistic Scheduling Algorithms for Wireless Communications

Swansea University
Prifysgol Abertawe

Toan To

College of Engineering

Swansea University

Submitted to Swansea University in fulfillment of the requirements for the degree of

Doctor of Philosophy (Ph.D.)

March 2012

Abstract

In this thesis, we propose a number of distributed schemes for wireless communications in the cross layer design context, considering an uplink random access network in which multiple users communicate with a common base station. In addition, we perform a comprehensive study on a splitting based multiuser selection algorithm which is simple, effective, and scales with the network size.

First, we investigate a reservation-type protocol in a *channel aware* ALOHA system. Various Markovian models are used to describe the system and to capture the temporal correlation of the channel evolution. The average throughput of the system is obtained using the Markov Analysis technique and we show that the reservation protocol can achieve better performance than the original channel-aware ALOHA by reducing the collision probability.

Second, for better resource utilization in the *Opportunistic Multichannel* ALOHA scheme, we propose a simple extension to the transmission policy that exploits the idle channels. Performance analysis shows that, theoretically, the maximum system throughput can be improved by up to 63% in the asymptotic case. Through numerical results, it can be seen that a significant gain is achieved even when the system consists of a small number of users.

Third, we consider a splitting based multiuser selection algorithm in a probabilistic view. Asymptotic analysis leads to a functional equation, similar to that encountered in the analysis of the collision resolution algorithm. Subject to some conditions, the solution of the functional equation can be obtained, which provides the approximations for the expected number of slots and the expected number of transmissions required by the algorithm in a large system. These results shed light on open design problems in choosing parameters for the algorithm when considering the delay and the overhead jointly. A typical example is to optimize the parameters that minimize the weighted sum of these measures of interest.

Finally, an application of the multiuser selection algorithm is demonstrated in a study of the energy efficiency for a wireless sensor network with distributed beamforming. To facilitate the cooperative transmission from multiple sensors, we propose a transmission scheme consisting of four phases: channel state information acquisition phase, sensor selection phase, beamforming phase, and cooperative transmission phase. Considering the number of sensors to be selected as the design parameter, analysis shows that there is trade-off between the energies required for sensor selection plus beamforming phases and the energy required for cooperative transmission phase. This observation is captured by numerical results, which can provide a design guideline for energy saving and prolonging network lifetime.

Declarations and Statements

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed.. ....(candidate)

Date..................................20/09/2012

STATEMENT 1

This thesis is the result of my own investigations, except where otherwise stated. Where correction services have been used, the extent and nature of the correction is clearly marked in a footnote(s).

Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed.. ....(candidate)

Date..................................20/09/2012

STATEMENT 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organizations.

Signed.... ....(candidate)

Date..................................20/09/2012

Acknowledgment

Working in the Wireless Communication Research Lab (WCRL), College of Engineering, Swansea University, for my PhD degree has been one of the most rewarding times in my life. It has been a great pleasure to interact with and learn from many people. First and foremost, I am deeply greatful to my supervisors, Prof. Jinho Choi, who guided me from the very first day of the program and taught me how to produce a proper research work, and Dr. Xinheng Wang, who provided great support in all aspects over the last few years, especially in the most difficult times of my research. In addition, I would like to express my beholden thanks to Dr. Duc To, who is not only my brother but also my friend, my collaborator for various contributions to my work. Also, thanks to the anonymous reviewers who have given constructive comments toward my research articles, in which this thesis was partly presented, and to all the past and present colleagues in the WCRL for many enriching discussions and camaraderie we fostered together.

My family is always with me. It is no doubt that without the loves from my wife, Van, and my son, Jason, it would be very difficult to complete this thesis and reach the current stage of my life.

This work was supported financially and spiritually by my parents. They deserve the most special thanks for all the unconditional support they have provided since the very beginning.

To whom who have dedicated
their life in doing research.

Contents

List of Figures

List of Tables

Abbreviations

AP	Access Point
BS	Base Station
CAA	Channel Access Algorithm
CA-ALOHA	Channel Aware ALOHA
CDF	Cumulative Density Function
CDMA	Code Division Multiple Access
CRA	Collision Resolution Algorithm
CRI	Collision Resolution Interval
CSI	Channel State Information
DOCSIS	Data Over Cable Service Interface Specification
D-CSI	Decentralized CSI
DVB-RCS	Digital Video Broadcasting - Return Channel via Satellite
EV-DO	Evolution - Data Optimized or Evolution - Data Only
FA	Free Access
FC	Fusion Center
FCFS	First Come, First Serve
FDMA	Frequency Division Multiple Access
GA	Gated Access
ID	Identity
IID	Independent Identically Distributed
ISO	International Organization for Standardization
LAN	Local Area Network
LLC	Logical Link Control
MAC	Medium Access Control
MPR	Multi-Packet Reception
MSA	Multiuser Selection Algorithm
MST	Maximum Stable Throughput
MTA	Modified Tree Algorithm
OMC-ALOHA	Opportunistic Multichannel ALOHA
OSI	Open System Interconnection

PDF	Probability Density Function
PHY	PHYsical (layer)
PRMA	Packet Reservation Multiple Access
R-ALOHA	Reservation ALOHA
SNR	Signal to Noise Ratio
STA	Standard Tree Algorithm
SWA	Simplified Window Algorithm
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
VBDC	Volume Based Dynamic Capacity
WA	Window Access
WAN	Wide Area Network
WLAN	Wireless LAN
WSN	Wireless Sensor Network

Chapter 1

Introduction

1.1 Overview

With the advanced developments in technology over the last decade, wireless communication services are now ubiquitous and essential in our modern life. As an indication, the list of wireless enabled devices that a typical consumer requires for everyday need is long and still growing, which includes from personal hand-held gadgets such as mobile phones, tablets, cameras, GPS navigators to office equipments such as printers and computers/laptops that are connected to the Wireless Local Area Networks (WLAN). More importantly, there is a clear trend toward tighter integration and seamless connections.

Typically, wireless communications between devices/terminals are provided over standardized networks. In this era of multimedia services, coupled with the explosive growth of the Internet, traffic carried over wireless networks is expected to be high in terms of data rate to support applications such as teleconferencing, video streaming, network gaming, etc .... Furthermore, these applications often require interactions between multiple users. Therefore, the most challenging issues that network designers face today are how to improve the overall network throughput and how to allocate the available resources efficiently amongst the different network users.

Traditionally, communication networks are designed based on layered models, an approach to simplify the implementation and maintenances. In accordance, similar communication functions are grouped into logical layers. Each layer is treated as a separate entity and is designated to perform a specific task. The most famous example is the Basic Reference Model [1] which has been successfully adopted throughout various deployments and applications worldwide. However, despite being conceptually correct, the layered architecture is now well over 30 years old and seems to be outdated. The reason is that the network stack was primarily designed for wired networking which ultimately aims at improving the interoperability between networks' devices/components. In the transition to wireless, the restrictive boundaries between layers make it difficult to cater for the increasing data rate demands and start creating bottleneck. Thus, in order to overcome the fundamental challenges, cross layer design has been coined as an alternative to escape from the current network model [2].

In comparisons with wired communication, the variation of channel condition over time and space due to movements and interferences, i.e., the fading phenomenon [3], is a completely different characteristic of the wireless physical channel. In the traditional design paradigm, the fading effect needs to be mitigated by complicated techniques in the physical layer (PHY). However, recent researches in cross layer design context have proved that the consideration of the transmission characteristic from the higher layers can be rewarded with better performance systems [4]. For example, if the channel state information (CSI) from the PHY layer is shared with the MAC layer for scheduling purpose, the system throughput can be improved substantially by allowing transmissions to/from users with the favorable channel conditions and exploiting the so-called *multiuser diversity* [5]. This *opportunistic scheduling* approach has attracted a considerable amount of research works over the last decade. Specifically, it was shown that multiuser diversity can be exploited in a distributed manner using a simple variation of the ALOHA random access protocol, referred to as the *channel-aware* ALOHA (CA-ALOHA) [6]. More recently, an *opportunistic multichannel* ALOHA (OMC-ALOHA) scheme which targets at exploiting both multiuser and multichannel diversities is proposed for distributed wireless system with multiple parallel channels [7].

In general, the opportunistic scheduling approach requires the selection of users with the best channel qualities for transmissions. Thus, finding an effective multiuser selection algorithm (MSA) is an interesting research problem in wireless communications, though it clearly has many other applications. Moreover, there has been a particular interest in distributed algorithms since, typically, the channel information or any other metrics related to the selection criteria are only available locally. For example, Qin and Berry proposed a simple splitting based single user selection in random access networks [8]. The splitting based selection algorithm was further developed by Shah et. al. [9] in an attempt to generalize the algorithm. However, the lack of a comprehensive study makes it difficult to find an optimal MSA and to match it with potential applications.

1.2 Contributions

In this thesis, we investigate a number of distributed opportunistic scheduling policies for wireless communications in the cross layer design context, considering an uplink random access network in which multiple users communicate with a common base station (BS). We then comprehensively study a novel splitting based MSA which is considered to be simple, effective, and scales with the network size. The detail contributions of the thesis are as follows:

A Reservation-type Protocol for Channel Aware ALOHA

The CA-ALOHA scheme is a simple variation of the famous ALOHA protocol proposed for wireless communication in the cross layer design context [6]. The scheme employs a binary scheduling policy which, to a certain extent, is an optimal random access protocol in the sense that the only loss due to distributed channel knowledge is the loss in contention for the channel. The first contribution of this thesis is an investigation of the effect of the channel memory in a CA-ALOHA system. Under a simple correlation model for the communication channels between the users and the BS, a reservation-type protocol is proposed. Various Markovian models are used to capture the behavior of the system. The average throughput is obtained using the Markov Analysis technique and we show that the reservation protocol leads to better system performance than the original CA-ALOHA by reducing the probability of collisions in transmission.

On Exploiting Idle Channels in Opportunistic Multichannel ALOHA

In a system consisting of multiple parallel channels, OMC-ALOHA is an optimal random access scheme that can exploit both multiuser and multichannel diversities by taking the advantage of fading channel [7]. However, as inherited from the nature of the contention based random access, there are a number of channels being idle and wasted in each time slot. For better resource utilization, we propose a simple extension to the transmission policy so that the idle channels can be exploited. The basic idea is to allow the users who are in sufficiently good channel conditions to access these channels. Performance analysis shows that, theoretically, the maximum system throughput can be improved by up to 63% in asymptotic case. Through numerical results, it can be seen that a significant gain is achieved even when the system consists of a small number of users. Moreover, as the number of users increases, the system performance also increases at the same rate with a centralized scheme, which shows that multiuser diversity is preserved in the proposed scheme.

On a Novel Splitting Based Multiuser Selection Algorithm

The term *splitting algorithm* originated from the Collision Resolution Algorithm (CRA) [10, 11], a class of random access protocols that handles the collisions by splitting the set of colliding users into smaller subsets and resolve them one after another. There have been several studies on a splitting based algorithm that distributively selects the best user, or in general, the q best users out of a large population set of size n for scheduling purpose in the literature [8, 9]. However, we noted that these works only consider binary splitting with a fair coin. In particular, after a collision, the set of collided users is split into two subsets. Furthermore, the probabilities that a given user joining either subset are both equal to $\frac{1}{2}$ and thus, a natural question regarding unfair (or biased) splitting was left unanswered. For this reason, we generalize the splitting based MSA using two design parameters, namely, the *contention factor* and the *selection factor*, and thoroughly examine the problem in a probabilistic view. Specifically, we are interested in two measures: i.) the number of slots (or rounds) required for the algorithm, and ii.) the average number of transmissions needed. In general, the former can be considered to be a delay associated with the selection procedure and the latter represents the algorithm's overhead.

In asymptotic case, we show that the expectations of both measurements are given in the form of a functional equation, similar to that encountered in the analysis of the CRA [12, 13, 14]. Subject to some conditions, the solution of the functional equation can be obtained, which provides the approximations for the expected number of slots and the expected number of transmissions required for a large system. These results shed light on open design problems in choosing parameters for the algorithm when considering the delay and the overhead jointly. A typical example is to optimize the parameters that minimize the weighted sum of these measurements of interest. Illustrative examples supported by numerical results are used to show the benefits of optimally choosing the parameters for different design purposes. In addition, we derive some lower bounds on the expected number of slots and the expected number of transmissions required by the proposed algorithms. These bounds capture the behavior of the measures as the number of users q to be selected increases, and it is particularly useful if q is considered as the main design parameter.

An Energy Efficient Cooperative Transmission Scheme with Distributed Beamforming and Sensor Selection in Wireless Sensor Networks

Last but not least, we demonstrate an application of the novel splitting based MSA in a study of the energy efficiency for a wireless sensor network (WSN) with distributed beamforming. Consider the scenario that multiple sensors have identical sensing information and would like to cooperatively transmit signals to a fusion center, we propose a transmission scheme consisting of four phases:

- CSI acquisition phase;
- sensor selection phase;
- beamforming phase;
- cooperative transmission phase.

Different to other transmission schemes in the literature, our proposal includes a new *sensor selection phase* that physically selects the best sensors for cooperative transmission and maximize the energy efficiency. Analysis shows that there is a tradeoff between the energies required for sensor selection plus beamforming phases and the energies required for cooperative transmission phase in deciding the number of sensors to be selected. This observation is captured by numerical and simulation results, which can provide a design guideline for energy saving and prolonging of network lifetime.

1.3 Thesis Organization

The rest of this thesis is organized as follows. Chapter 2 provides some backgrounds of random access control schemes in wireless communications and an introduction to a standard ALOHA-type multiple access environment, which is used throughout the thesis. In Chapter 3, we investigate the effect of channel correlation in the CA-ALOHA system by studying a distributed reservation-type protocol. For an OMC-ALOHA set up, Chapter 4 proposes an extension to the transmission policy so that the idle channels can be exploited to improve the network resource utilization. Chapter 5 and Chapter 6 are devoted to a comprehensive study of a generalized splitting based MSA with two design parameters. In Chapter 7, an application of the MSA is demonstrated in a study of energy efficiency for WSN with distributed beamforming. Finally, Chapter 8 summaries the thesis and outlines some possible future research directions.

Note that, for mathematically tractable analysis, we only consider a standard simplified model of the *homogeneous system*, in which the users are assumed to be identical, throughout this thesis. Consequently, the fairness issue is not concerned in our work. In the literature, the opportunistic scheduling algorithms similar to our study for *heterogeneous system*, in which the users may be different, have been researched [15, 7] and it has been shown that the *proportional fairness* [16] is guaranteed if each user behaves exactly as if it is still in a homogeneous system. In this sense, we can conjecture the same conclusion for our research.

Parts of this work have been presented in the following journal paper and conference proceedings:

- T. To and J. Choi 'On Exploiting Idle Channels in Opportunistic Multi-channel ALOHA," *IEEE Commun. Lett.,* Vol. 14, No. 1, pp. 51–53, Jan. 2010.
- T. To, D. To, X. Wang, and J. Choi "A Reservation-type Protocol for Channel-aware ALOHA," in Proc. *21st IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, Istanbul, Turkey, Sep. 2010.
- T. To, D. To and X. Wang "On the Functional Equation Arising in a Single User Selection Algorithm," in Proc. *IEEE GLOBECOM 2011*, Houston, Texas, Dec. 2011.
- D. To, T. To and J. Choi "Energy Efficient Distributed Beamforming with Sensor Selection in Wireless Sensor Networks", in Proc. *IEEE Vehicular Technology Conf. (VTC), Spring 2012*, Yokohama, Japan, May 2012.

Chapter 2

Background of Random Access Schemes for Wireless Communications: ALOHA and Collision Resolution Algorithm

2.1 Introduction

This chapter introduces fundamental backgrounds of the medium access control (MAC) in wireless communications. In particular, we focus on two well-known random access schemes: the ALOHA and the Collision Resolution Algorithm (CRA). These distributed scheduling algorithms importantly serve as the baselines of the works being presented in this thesis.

The chapter starts with an introduction to a traditional network design model and a MAC problem in Section 2.2. Section 2.3 is devoted to a study of the ALOHA concept in conjunction with the standard multiple access system models. In addition, associated stability issues are also discussed. Developments of the CRA and its improvements are described in details in Section 2.4. Finally, Section 2.5 summaries the chapter.

2.2 Traditional Network Design Paradigm and the Medium Access Control Algorithms

Traditionally, communication networks are designed based on layered models such as the Basic Reference Model [1]. This product of the Open Systems Interconnection (OSI) effort at the International Organization for Standardization (ISO), often named the OSI model, defines a networking framework for implementing protocols in abstraction layers. In accordance, similar communication functions are grouped into logical layers. Components within a layer, often called instances of that layer, provide services to its upper layer instances while receiving services from instances of the layer below. Adoption of the reference model is believed to simplify the networking designs and maintenances for two reasons. Firstly, it ensures different network devices would all be compatible even if built by different manufacturers. Secondly, the OSI model makes network developments more robust and extensible since new protocols and network services are generally easier to be added to a layered architecture than to a monolithic one [17].

The OSI model briefly comprises of seven layers as shown in Fig. 2.1. Each layer is designated to perform a specific task. For example, the physical layer (PHY) defines coding methods, hardware connections, and media types for the actual transmission of bits over a physical medium. Up one level, the data link layer provides functions and procedures for transferring data between network entities, detecting and possibly correcting errors that may have occurred in the PHY layer. Originally, this layer was intended for point-to-point and point-to-multipoint communication as mainly seen in Wide Area Network (WAN). In Local Area Network (LAN) standard which was developed independently of the ISO work in the IEEE Project 802, the Data Link Layer is divided into MAC and Logical Link Control (LLC) sub-layers following the multiple access requirement. As the results in the IEEE 802 series of committees, the sub-layers are specified

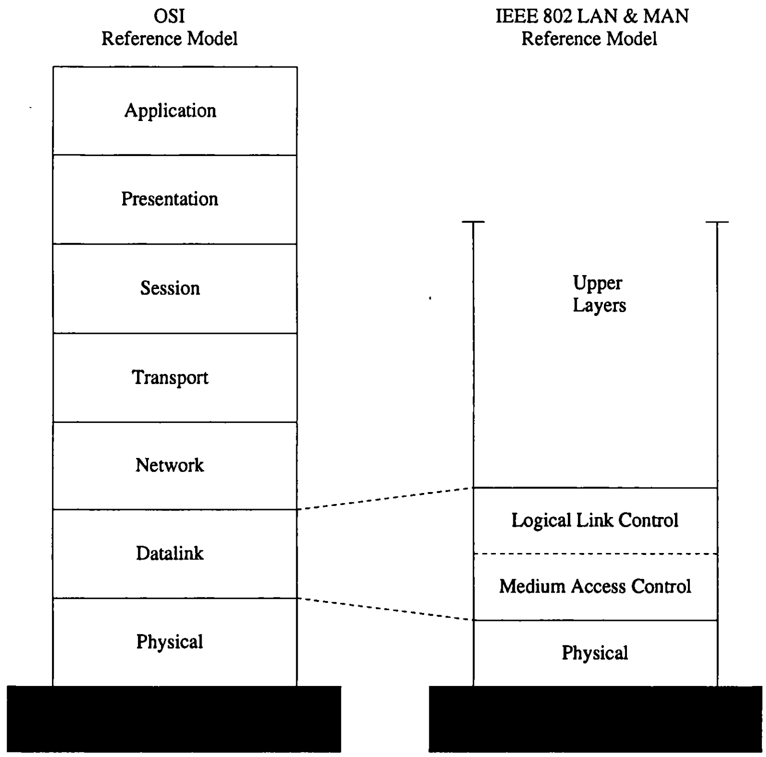

Figure 2.1: Architectural view of the OSI and the IEEE 802 reference models

such that the MAC controls how a node on the network interacts with the shared channel while the LLC shields the higher layers from concerns with the specific LAN implementation [18].

The standardization of the network model is only the starting point. There have been a wide variety of networks designed today. Most of these networks are not simple point-to-point communication, and often require interactions between multiple users or terminals. Without a prior agreement on when a user can transmit on the shared medium, we possibly end up with the situation of simultaneous transmissions, which can lead to an erroneous reception of data. Thus, a MAC protocol defining rules for orderly access the physical channel has always been fundamentally a key problem in communications [19, 20].

There are two conventional approaches to solve the multiple access problem. For protocols that belong to fixed access category such as Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA), and Code Division Multiple Access (CDMA), each user (transmitter, mobile terminal, mobile device, network node etc ...) is allocated with a certain amount of network resources, e.g., time, frequency, or the mixture of both. The user is only allowed to access the shared medium using its allocated resources but not those belonging to others. To a certain extend, this centralized scheduling based approach can be a very efficient solution in a small network with steady and relatively heavy traffic. However, when traffic is low and bursty, predefined allocation schemes mentioned above are not preferred due to low channel utilization and possibly high delay experienced by users in a large network. In this situation, contention based random access schemes such as ALOHA [21] and CRA [10, 11] are more efficient due to the excellent delay-throughput characteristics they offer. Moreover, these distributed algorithms are usually easier to implement than those based on the centralized approach.

The main disadvantage of random access techniques is low throughput. The reason is because even with a careful choice of transmission rules, there is no guarantee that simultaneous transmissions, i.e., conflicts or collisions, are completely avoided. In addition, the ALOHA type protocols may suffer from the stability issue [22]. Nevertheless, contention based access techniques have been widely used as the MAC protocols and/or components of reservation protocols in many communication environments such as mobile cellular communications, satellite data networks, Data Over Cable Service Interface Specification (DOCSIS) systems, etc, mainly because of their distributed property, especially when a central scheduler is not or has not been established. Digital Video Broadcasting - Return Channel via Satellite (DVB-RCS) [23] is a good example of how random access is used for channel reservation. Initially, contention slots are used for data packets and bandwidth requests. Volume Based Dynamic Capacity (VBDC) and constant rate assignment starts after sufficient information has been exchanged, resulting in a good trade off between low delay and bandwidth efficiency.

The research works being presented in this thesis have been developed based on two popular classes of random access protocols, namely ALOHA and CRA. In the rest of this chapter, the principle of the ALOHA and the development of the CRA are discussed in more details to provide the reader with sufficient background information.

2.3 ALOHA

2.3.1 The Development of the ALOHA

ALOHA was the first random access scheme proposed in communication. Developed by Abramson and his colleagues within the framework of the ALOHAnet program, the original protocol allows many remote users to share a single radio channel - the ALOHA channel - using a simple algorithm without the need of a central controller [21]. In principle, its concept is nothing but transmit at will. If packeted transmissions from two or more users are overlapped, collision occurs. The users whose packet collided attempt for retransmission, but at a random later time.

If the arrival times of the packets in an ALOHA channel can be modeled as a Poisson point process with parameter λ packets per second, the normalized channel traffic can be defined as

$$G = \lambda\tau, \tag{2.1}$$

where τ denotes the duration of a packet (in second). The normalized throughput of an ALOHA random access channel is then

$$S = Ge^{-2G}, \tag{2.2}$$

which implies that the maximum throughput that can be offered by the protocol is $\frac{1}{2e} = 0.184$, which is attained when the traffic G is equal to 0.5.

The performance of the ALOHA is greatly improved if a synchronized time base is established and the users are allowed to transmit packets only at the beginning of a time slot. In particular, it is assumed that packets to be transmitted are of the same length, and the time for transmitting a packet is approximately equals to the slot duration. Thus, any partial packet overlaps are eliminated. The resulting protocol is referred to as the slotted ALOHA [24], which potentially doubles the maximum throughput of the original version, i.e., $\frac{1}{e} = 0.368$. It is not difficult to predict that the peak performance occurs when the channel traffic is equal to 1.0.

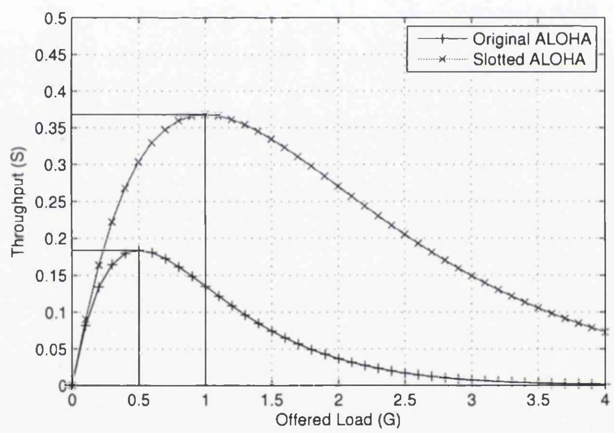

Figure 2.2: ALOHA throughput

Fig. 2.2 compares the throughput of ALOHA protocols as the input load varies. Clearly, even with the slotted version, the offered throughput is not significant compared with the centralized schemes which potentially achieve the normalized throughput of 1 (under heavy traffic condition). However, due to its simplicity and flexibility, the ALOHA concept became very popular and has been implemented commercially from early days [25, 26]. Over the last four decades, ALOHA and its variants have been adopted for use in all major mobile telephone standards (1G, 2G and 3G) and as components of many protocols in wired networks.

2.3.2 The ALOHA-type Channel

Ever since the landmark work by Abramson [21], most of the research works on ALOHA focused on the synchronized version in multiple access systems, i.e., the slotted ALOHA protocol. Moreover, they have been developed based on the so-called *collision* channel model. That is, if two or more users transmit at the same time, a collision occurs and no data is successfully delivered. Nevertheless, if a given slot contains a single packet transmission, the packet will be received correctly by the receiver. Thus, the state of the channel in any given slot can be one of:

- *idle:* no packet transmission is taking place;
- *success:* a single user successfully transmits its packet to the receiver;
- *collision:* two or more packets are being transmitted simultaneously, causing destructive interference among them.

Assuming that the propagation delay is negligible, at the end of each time slot, an instantaneous feedback is given to all transmitters so that they are aware of the state of the channel during the slot that just ended. In most cases, the feedback is *ternary*, denoted by i (or 0), s (or 1) and c (or e) for idle, success and collision, respectively. However, the uses of *binary* feedback of the type "collision–no collision" or "success–failure" are possible in some cases and have been considered in the literature [27].

Overall, the synchronous (slotted) operation, collision channel model and instantaneous error-free feedback form a set of common assumptions that describe the so-called standard environment, often termed the *ALOHA-type* channel [28]. Such standard environment has been adopted extensively for the analyses of not only the ALOHA but also many other multiple access systems, in conjunction with either finite or infinite user models. The underlying user models and the stability issues associated with the ALOHA protocols are explained in the following sub-section.

2.3.3 User Models and the Stability Issues

Infinite User Model

The infinite user model describes a system with the unlimited number of users where each user has a buffer with capability of storing at most one packet. With this assumption, no queues of packets are formed by users. The users who have packet to transmit are called backlogged users and the only queue that is considered in this model is the queue of backlogged users. Once a backlogged user successfully transmitted its packet, it is dropped out of the system. Furthermore, it is assumed that the number of packets generated in the network during each time slot is a random variable, commonly modeled with Poisson distribution [19, 20], and they always arrive at the newly generated users. This model, although unrealistic, captures the multiple access systems with a large number of users and a relatively small arrival rate. Under these conditions, the fraction of backlogged users is typically small and new arrivals at backlogged users are almost negligible. In addition, the model greatly simplify the system description and the analysis since it does not distinguish between the terms *backlogged users* and *waiting packets*, making them interchangeable.

Given a slotted ALOHA random access system, backlogged users attempt their transmissions with probability p. In usual analysis, the number of backlogged users X_t at the beginning of a time slot t is modeled as a discrete time homogeneous Markov chain, assumed to be aperiodic and irreducible. The system is said to be unstable if the associated Markov chain is transient which is the case that the number of backlogged users becomes infinite with probability one as time approaches to infinity [1]. Unfortunately, this is always the case for the slotted ALOHA with a static transmission probability [22]. Thus, stabilizing the system by adaptively controling the global variable p is an important issue related to the design of slotted ALOHA. Several methods to solve the transmission control (back-off) problem under the infinite user model have been reported. For example, Hajek [29] suggested a simple method to directly control p based on the feedback, while Clare [30] proposed an algorithm to estimate the number of backlogged users, then compute p using the backlog estimate. Both examples are efficient in the sense that they can achieve the theoretical maximum (or almost) throughput of slotted ALOHA, i.e., $\frac{1}{e}$ packets/slot.

Finite User Model

As mentioned, the infinite user model is unrealistic. In practice, we often encounter multiple access systems with a finite number of users, each has a (possibly) large buffer to store packets. However, a real system can be alternatively viewed as containing an infinite set of users if each user regards itself as a set of virtual users, one for each packet queued in its buffer. Thus, given a multiple access algorithm and that we allow for packets from virtual users to compete with each others, theoretical analyses related to the infinite user model provide a lower bound on the maximum stable throughput (MST) as well as an upper bound on the expected packet delay of the system.

[1]Definition for the stability of the system is sometimes (informally) made related to the expected packet delay [19].

With regard to the finite user model, the evolution of the queues at the buffers and the stability property of the slotted ALOHA system have been well studied [31, 32, 33, 34, 35]. From a queuing theoretic point of view, stability can be interpreted as the convergence of the queue length in distribution to a proper random variable, i.e., there exists a proper limiting distribution. Formally, consider a multiple access system with N users and let the N-tuple $\mathbf{Q}^{(t)} = (Q_1^{(t)}, \ldots, Q_N^{(t)})$ represents the queue lengths of the buffers at the beginning of time slot t, the stability can be defined as follow.

Definition 1 [31, 34, 35, 36, 37, 38]: For any random vector of natural number $\mathbf{x} \in \mathbb{N}^N$*, the system is said to be stable if there exists a* $F(\mathbf{x})$ *such that*

$$\lim_{t\to\infty} \Pr\{\mathbf{Q}^t < \mathbf{x}\} = F(\mathbf{x})$$
$$\textit{and} \lim_{\mathbf{x}\to\infty} F(\mathbf{x}) = 1, \tag{2.3}$$

where $F(\mathbf{x})$ *is the limiting distribution function, and by* $\mathbf{x} \to \infty$*, we understand that* $x_i \to \infty$ *for all* $i \in \mathcal{N} = \{1, \ldots, N\}$.

Furthermore, if the packet arrival at the queues is also a multi-dimensional stochastic process with the mean arrival rate denoted by $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_N)$, it would be natural to expect a queuing system to be stable if the average arrival rate is less than the average departure rate [39]. This gives rise to the following characterization of stability.

Definition 2 [34, 37]: For the N-user slotted ALOHA system with a given arrival process distribution, the stability region is defined as a closure of the set of arrival rates $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_N)$ *such that there exists a vector of transmission probabilities* $\mathbf{p} = (p_1, \ldots, p_N)$ *for which the queues in the system are stable.*

The first complete characterization of the stability region of slotted ALOHA for the two-user case was given by Tsybakov and Mikhailov [31]. Unfortunately, the characterization of the stability region for the general N-user case turned out to be extremely difficult for queuing theoretic analysis due to complex interactions among the queues. For the symmetric case when all users are identical, Tsybakov and Mikhailov [31] showed that maximum stable throughput is $\left(1-\frac{1}{N}\right)^{N-1}$. Letting $N \to \infty$, we again obtain the asymptotic stable throughput of $\frac{1}{e}$.

2.4 Collision Resolution Algorithm

2.4.1 Splitting Algorithm and Tree Structure Representation

The ALOHA and its variations form probably the most popular family of MAC protocols due to the maturity of being the first random access technique introduced. They suffer, however, from stability problem as mentioned in the previous section. Effectively, the ALOHA protocols have zero capacity if no control procedure is employed to stabilize the queue(s) [22]. The next major breakthrough in the area of distributed scheduling was the invention of the CRA, a contention based algorithm which, as its name suggests, handles the collisions by resolving them as soon as they occur. The simplest protocol of this type was first introduced in late 1970s by Capetanakis [10] and, independently, by Tsybakov and Mikhailov [11], who described the collision resolution procedure using a tree structure. Later, Berger [40] and Wolf [41] noted that the underlying concept had been known for long time in the context of a "Group Testing" [42, 43, 44].

Similar to the ALOHA, the CRA and its variants have been intensively studied under the standard ALOHA-type channel environment. The algorithm, however, is not only provably stable but also achieves a higher maximum throughput than the slotted ALOHA protocol. Broadly speaking, it follows a "divide and conquer" procedure that exploits the feedback information available to users to control retransmission process. Whenever transmissions in a time slot result in a collision, the set of users involved is split into smaller subsets, for instance, randomly by tossing a multi-faces coin. The subsets are then scheduled for transmissions in the subsequent time slots, one after another. If the next transmission is, again, a collision, the procedure is repeated. Otherwise, the subset is said to be satisfied, i.e., on the reception of a *success* or an *idle* feedback [1]. We say that the original collision is completely resolved when all the subsets have been satisfied (all the backlogged users involved with the collision have been transmitted successfully) and the time it takes is called a collision resolution interval (CRI).

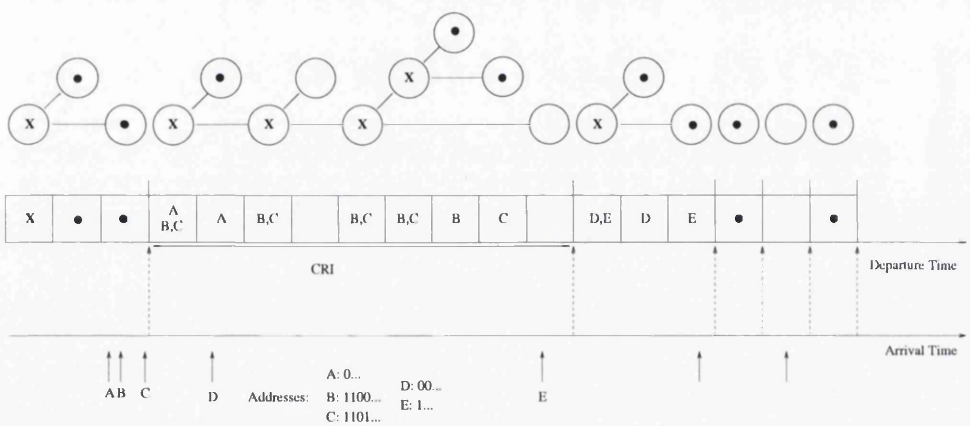

Figure 2.3: Example of a binary standard tree

The nature of the algorithm described leads to an alternative term for the algorithm: the Splitting Algorithm. It can be graphically represented by a tree structure as depicted in Fig. 2.3, where the root of the tree being the first slot of the CRI. This binary tree example is very similar to the original Capetanakis-Tsybakov-Mikhailov algorithm [10, 11], and is commonly referred to as the standard tree algorithm (STA) [28]. If the root of the tree is an idle state, or results

[1]If the system operates with binary type feedback, one can use "no-collision" to represent a satisfaction of a transmission, i.e., to indicate both *success* or an *idle* collectively

in a successful transmission, it is a terminal node. Otherwise, two subtrees emanating from the root, where each one represent a subset of users involved with the original collision after splitting.

2.4.2 Channel Access Algorithms

To a certain extent, CRA as explained previously is simply a distributed algorithm that handles collisions. Since it is originated from a well-known concept "Group Testing", it obviously can have applications in various contexts. In term of communication, CRA must be combined with a Channel Access Algorithm (CAA) which specifies when new packets may join a CRI in order to form a complete random access protocol.

The first CAA proposed in the late 1970s is the *gated channel access* [10, 11]. According to this algorithm, packets arriving at the system during a CRI are buffered and get transmitted at the beginning of the next CRI. That said, in order to join the CRA, packets have to go through a *gate*. The gate opens only at the beginning of a new CRI to allow the waiting packets in. It is then closed before the next transmission takes place and remains so until the end of the CRI.

With gated access (GA) algorithm, the durations of two consecutive CRIs are correlated since all packets generated during a CRI get transmitted in the following one. A long CRI is likely to be followed by another long CRI and vice versa [28]. This is critically an issue that limits the system performance because allowing a large number of packets in a CRI is undesirable and should be avoided. Consequentially, protocols with GA, though stable, offer lowest capacity amongst those of the same family.

An approach to break such correlation and ensure that a more reasonable number of users to join the CRA at the beginning of a CRI is the *window channel access* method suggested by Massey [45]. The algorithm resembles GA, but when the gate opens, only packets that arrived in a specified time period - a *window* on the arrival time axis - are allowed to join the new CRI. Let δ be the maximum window size specified by the window access (WA) algorithm. Also, let τ be the *lag* of the algorithm, defined as the time from the end of the current window (right boundary), until the end of the CRI it generates. The size of the next window can simply be $\min\{\tau, \delta\}$. Alternatively, instead of having a variable window size, a constant window size can be maintained by forcing the users to delay the start of the next CRI, if necessary, until the maximum window size is reached. In this case, the algorithm is called simplified window algorithm (SWA). Although the SWA exhibits higher delay than the WA, it is not difficult to see that they naturally have the same capacity [28].

GA and WA algorithm are collectively referred to as *blocked access*. In comparison with them, *free access* (FA) [46, 12, 13, 47] is far more simple algorithm: new packets are transmitted immediately at the beginning of the next slot following their arrival. With this method, the users can join the CRI at any time and they do not need to keep track of the transmission history (in term of feedback) to decide the CRI boundaries. Therefore, the network need not be brought up simultaneously; instead, users can drop in and out at random. This important property is usually referred to as *limited sensing* [28] and is the main advantage of FA over blocked access in term of implementation.

Fig. 2.4 and Fig. 2.5 present graphical examples of binary splitting algorithms with WA and FA, respectively. The STA depicted in Fig. 2.3 uses GA. The same input scenario is assumed in all three examples for a fair comparison.

2.4.3 Addressing Schemes and Stack Implementation

There are many ways for the set of users to be split after a collision. For example, in the case of STA, one can assume that each user has a binary address. After

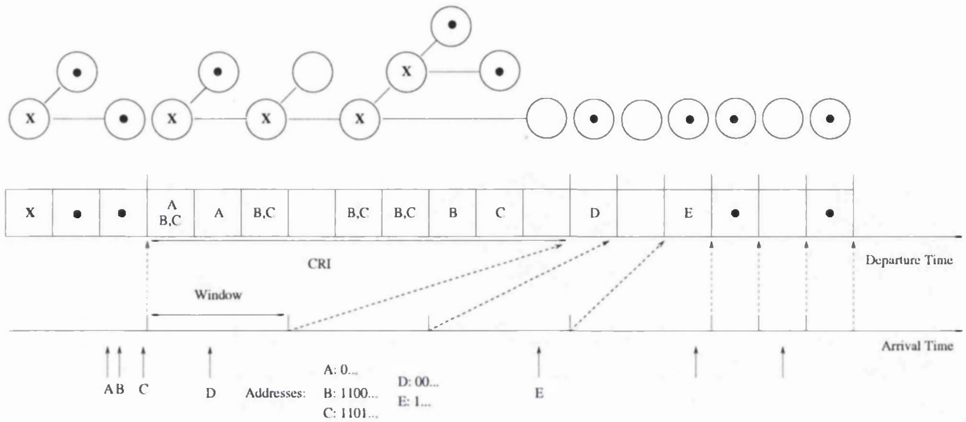

Figure 2.4: Example of a binary tree with window access

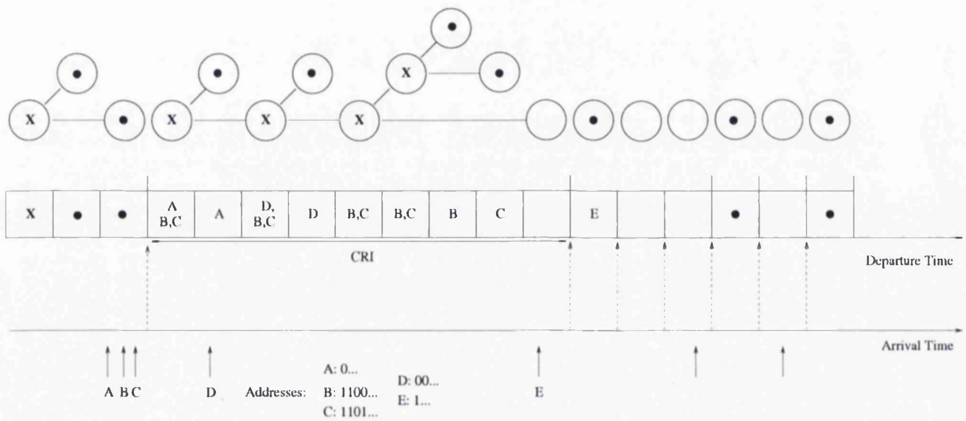

Figure 2.5: Example of a binary tree with free access

a collision, users can use the corresponding bits of the addresses to decide which subsets to join. This scheme is called *deterministic addressing*. Alternatively, users may join one of the subsets at random, i.e., *random addressing*, by generating a random bit and choose the first subset (resp. second subset) if it is 0 (resp. 1) [28]. Another common way is to use the arrival time of the packet as the index for making decision [48, 49, 50], hence the term *arrival-time addressing*. For this approach, packets that arrived in the first half of the last tried interval will join the first subset after a collision, while those arrived in the second half collectively form the second subset. Note that, window channel access algorithm does not imply the use of arrival time addressing. Nevertheless, a combination of the blocked access and the arrival time addressing would normally result in a protocol that have a *first come first serve* (FCFS) property [1].

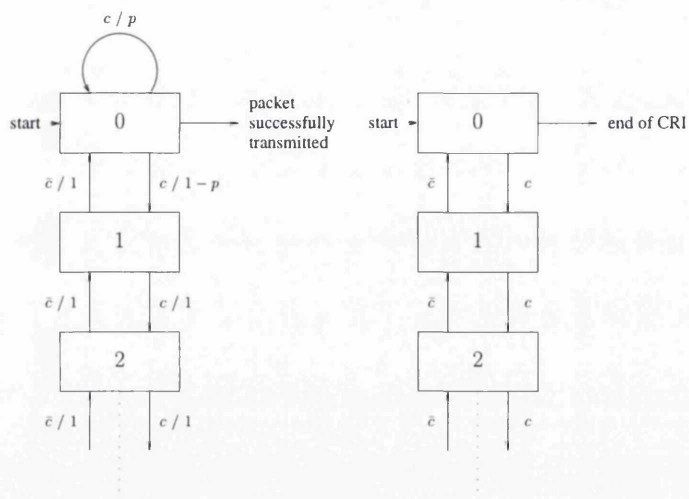

Figure 2.6: Stack Interpretation of the STA [28]
(Labels show feedback / splitting probability)

Up to this point, we have all the required elements to describe a complete collision resolution random access protocol. Fig. 2.6 presents a stack interpretation of the binary STA. Each user maintains two variables, which are the local stack pointer and the system stack pointer. While the latter is used by the CAA to determine the CRI boundaries, the former is used by the CRA to schedule transmission attempts from the user. When the system stack is at the top level, the user assume it is the beginning of a new CRI. Also, a user can transmit if its packet is at the top of the local stack. The system stack gets pushed one

[1] However, the term FCFS is often used for the Gallager-Tsybakov-Mikhailov algorithm, also known as the 0.487 algorithm [48, 51, 52]

level after a collision, while the local stack does so only if the user decides to join the second subset after a collision (generate a random bit "1", for example, with random addressing) or did not participate in the last collision. For STA, both stacks get popped after a no-collision (success or idle) slot. When the system is brought up, all users start with their packets at the top of each stack.

As mentioned, when the FA algorithm is used, users do not need to keep track of the CRI boundaries. Consequently, the system stack pointer is no longer required. A user with a newly generated packet sets its local pointer to the top of the stack and transmits immediately in the following slot [46, 12, 13, 47]. If no other users transmit, the packet is successful. However, if there is another transmission (or more), collision occurs and no packet gets though. In either case, other users cannot distinguish whether the successful transmission/collision is due to a new packet transmission or those previously involved. Thus, it makes no differences in their perspective and they simply push or pop their local stacks according to the feedback they receive.

2.4.4 Improvements to the Standard Tree Algorithm

A Modified Tree Algorithm - Level Skipping

The original STA sees no difference between success and idle slots. Thus, only binary type feedback "collision-no collision" is required. If ternary feedback is available and that the collision is followed by an idle slot, it is obvious that all the packets involved with the collision will be scheduled for transmission in the next slot (assigned to the second subset), thus, will be generating another guaranteed collision. Therefore, it serves no purpose to let the algorithm to proceed with this transmission and waste another slot. Instead, the algorithm can be modified to omit this step and jump directly to the next level of the tree. The algorithm in this case is referred to as the modified tree algorithm (MTA) or the *level* *skipping* algorithm [19, 20, 28]. This simple modification increases the capacity of the splitting algorithm from 0.346 [10, 11] - which is below the capacity of (stabilized) ALOHA to 0.375 [11, 45]. Unfortunately, it does have an unexpected side effect: the deadlock situation in the presence of feedback error. In specific, if a single idle slot is incorrectly interpreted as a collision, the active set, though empty, will be split into 2 sub-(empty)-sets. The next slot will likewise be idle, and the MTA will automatically split the second empty subset again and again endlessly, causing deadlock. Therefore, in practice, the number of successive level skipping slots should be limited, say, to the maximum value h, which can be moderately large if the feedback is quite reliable, otherwise h should be small. Note that the STA is not exposed to the same problem and is more robust than the MTA since the second empty subset always gets transmitted.

Biased Splitting and d-ary Splitting

The basic STA performs optimally when fair splitting is used. Intuitively, this is predictable from the symmetry of the algorithm description as well as the respective equations arising in studying the system performance [1]. The MTA, on the other hand, treats the second subset differently if the first subset was empty. Thus, for better performance, biased splitting becomes nature. It is shown that the performance of MTA with GA attained the normalized capacity of 0.381 when the splitting probability is optimally chosen [47]. For the case of WA, if both the (maximum) window size and the splitting probability are optimized, the MST of the protocol is 0.468, somewhat impressive in comparison with the (stabilized) ALOHA.

Although Tsybakov and Mikhailov described their first CRA through d-ary trees [11], the formal generalization and analysis was only carried out in mid-1980s [47]. Interestingly, it was shown that *ternary* splitting is optimal for certain

[1]The chapter is devoted for background information and we omit all related detail analyses of the CRA protocols. Interested readers are referred to [19, 20, 28] for rather complete surveys and discussions.

protocols, for example the STA with GA/FA or the MTA with FA. In addition, it is not difficult to modify the addressing schemes and the stack implementation to adapt to the change. More specially, a level is skipped if all $d-1$ previous slots at the same level were empty.

Tree Prunning and the FCFS Algorithm

The next important observation about the Tree Algorithm is *tree prunning*, which was independently discovered by Gallager [48], Tsybakov and Mikhailov [51], and others (Ruget and/or Pellaumai [53]). The idea of tree prunning is as follows: if the first subset after a collision also results in another collision, we have no information about the number of packets in the second subset. In theory, this second subset is likely to be empty, and a slot might be wasted if the algorithm proceeds with it. Thus, instead of following the normal operation, it is more efficient to drop this subset from further consideration in the current CRI, i.e, to *prune* the tree branch, and to combine it with some other yet unexamined traffic. A complete random access protocol with a combination of level skipping, tree prunning, arrival time addressing and WA is often known as the Part-and-Try algorithm, the FCFS algorithm, the Gallager-Tsybakov-Mikhailov algorithm, or simply the 0.487 algorithm. Further study on modifications to the FCFS algorithm using optimal dynamic biased splitting leads to the most efficient contention based multiple access protocol to date, which can achieves the MST of 0.4878 packets/slot [49], again, assuming an infinite population and a Poisson arrival model.

2.5 Summary

In summary, the medium access control problem has always been a fundamental issue in communications. Solutions to this problem are often classified into two categories depending on the network resources allocation. While fixed access schemes such as TDMA and FDMA are effective in small networks with steady and heavy traffic, random access control protocols perfectly suit large networks with bursty traffic due to the excellent delay-throughput characteristics they offer.

In this chapter, we provided some background on the two well-known contention based techniques: the ALOHA and the CRA. The former was the first random access scheme proposed in communication. In principle, its concept is nothing but transmit at will. In the event of a collision, the users involved attempt the retransmission at random time later. Due to flexibility and simplicity of the algorithm, the ALOHA and its variations have been adopted widely as the MAC protocol and components of the reservation protocols in many communication environments.

The ALOHA protocol suffers from low throughput and stability problem. The next major breakthrough in random access was the introduction of the CRA, a distributed algorithm which handles the collisions by resolving them as soon as they occurs. In particular, it exploits the feedback information available to users algorithmically to control retransmission process. In terms of communications, a CRA needs to be combined with a CAA in order to realize a random access protocol. Interestingly, protocols of this type are not only provably stable but also can achieve a higher maximum throughput than the slotted ALOHA.

There has been number of researches on the CRA over the last three decades. Of all the developments and variations of the CRA, the FCFS algorithm using optimal dynamic biased splitting is the most efficient contention based random access protocol, which achieves the MST of 0.4787 packets/slot for the infinite population model and the Poisson arrival distribution.

The principles of the above mentioned distributed scheduling techniques are the baseline of the research in this thesis. In particular, we present the ALOHA-type protocols in cross layer design context in Chapter 3 and 4. We then propose a novel splitting based multiuser selection algorithm (MSA) using two design parameter. This proposal shares the "divide and conquer" principle with the CRA; therefore, the two problems are naturally related. However, it shall be noted that the MSA may have many applications, which includes scheduling in wireless communication as we are studying in this thesis. Probabilistic studies on the MSA in Chapter 5 and Chapter 6 lead to the lower bounds and the exact expressions for the calculation of two measures of interest in the asymptotic case. Finally, Chapter 7 demonstrates an application of the proposed splitting algorithm in a cooperative transmission scheme with the distributed beamforming for wireless sensor networks.

Chapter 3

A Reservation-Type Protocol for Channel-Aware ALOHA

3.1 Introduction

Traditionally, communication systems have been designed based on the layered structure where the OSI model [1] is typically adopted. In this way, communication protocols for different layers are developed independently, making designs and maintenances flexible. In particular, systems can be deployed with devices/components/equipment from different manufactures. Furthermore, replacement of the protocol implementation in one layer can be done seamlessly without any effect on other layers [17]. While this abstraction model is provably useful in designing wired networks, the limited interaction between logical layers can be an issue in wireless networking due to completely different characteristics of the physical channel that needs to be coped with. For example, variations of the channel states over time and space due to movements and interferences, i.e., the fading phenomenon [3], limit the performance of the wireless systems. Thus, in transition from wired to wireless communications, cross layer design has been used as an alternative to the traditional paradigm [2]. The principle idea behind exchanging information between layers is to use various parameters from different layers for joint optimization of the protocols across the communication stack.

In particular, given a multiuser communication system, if the channel state information (CSI) from the physical layer (PHY) is shared with the MAC layer for scheduling purposes, the system throughput can be improved substantially by allowing transmissions to/from the users with the favorable channel conditions exploit the *multiuser diversity* [5] inherent in the wireless environment. While this *opportunistic scheduling* approach attracted a number of researches [54, 55, 56, 57, 16] and has been incorporated into the practical design of Qualcomm's High Data Rate system (Evolution Data Optimized or Evolution Data Only first generation - 1xEV-DO) [58] for downlink, the associated overhead and the long delay incurred by obtaining the CSI are not easy to justify in the reverse direction (uplink). For this reason, there is a recent line of works that study the effect of CSI on the distributed multiple access networks (see [36, 6, 15] and the references therein). In these papers, a common assumption is that each user only has access to its own CSI - this is often termed the *decentralized CSI* (D-CSI) assumption. Specifically, Qin and Berry [6] showed that the multiuser diversity can still be exploited in a distributed manner using a simple variation of the ALOHA protocol, referred to as the *channel-aware* ALOHA (CA-ALOHA). In this scheme, a user transmits its data (with probability 1) to the common receiver, only if its channel gain is higher than a predetermined threshold H_t. The threshold value is chosen to maximize the probability of success for each transmission, hence, maximize the system throughput. Asymptotically, this binary scheduling policy is an optimal random access scheme in the sense that only loss due to distributed channel knowledge is the loss due to contention for the channel. The ratio of the throughput of the CA-ALOHA to the throughput of the centralized system [1] is shown to be $\frac{1}{e}$, a well-known ratio achieved by the ordinary slotted ALOHA system. This intuitive claim was later formally proved by Yu and Giannakis [15]. Furthermore,

[1] The centralized system, in this sense, is the system in which the common receiver has full knowledge of the CSI of all communication channels from the users and is responsible for coordinating (scheduling) transmissions for the users.

the authors of [15] also stated that the channel with memory does not affect the maximum throughput of the system since the users do not cooperate. Each user simply makes its own transmission decision based on the expected contention in the current time slot, but not on any previous channel history. Nevertheless, the above statement does not imply that knowledge of the channel memory cannot be exploited.

In this chapter, we study the impact of the channel temporal correlation in a CA-ALOHA system. Under a simple correlation model for communication channels between users and the base station (BS) [1], where the channel can be one of the two states 'GOOD' or 'BAD', a reservation-type protocol is proposed. Its principle is to allow the transmitting user who is experiencing a 'GOOD' state to reserve the channel for the subsequent time slots until its channel turns 'BAD'. We will show that the system can enjoy higher throughput than the traditional CA-ALOHA as the probability of collisions is reduced.

Ever since the landmark work by Abramson [21], there has been a vast number of researches on the ALOHA and its variation. In particular, the reservation-type protocols such as the reservation ALOHA (R-ALOHA) and packet reservation multiple access (PRMA) have been proposed and analyzed assuming the static channel model [59, 60, 61, 62, 63]. However, to the best of our knowledge, there has been no work that considered reservation-type protocols in the cross-layer design perspective. Therefore, our ultimate aim is to provide an insight into exploiting channel time correlation in the random access.

The rest of this chapter is organized as follows. In Section 3.2, the CA-ALOHA system and the correlated channel model are described. Section 3.3 explains the proposed reservation-type protocol. Markovian models that are used to obtain the system throughput are developed in Section 3.4. Numerical study is presented in Section 3.5 to justify our proposal. Finally, Section 3.6 concludes this chapter.

[1]Throughout the thesis, the term BS will be consistently used as an alternative to the common receiver, central node, access point, etc terms, unless stated otherwise.

3.2 System Model

3.2.1 Random Access Network Model

We consider a time-slotted random access network where N users would like to send data to a central BS over a shared medium. The channels between the users and the BS are assumed to be block-fading [64]. That is, channel states remain unchanged within each time slot and varies from one slot to another. At the beginning of each time slot, it is assumed that each user only knows its own CSI, i.e., the D-CSI assumption. We only consider packet transmissions and assume that the users are homogeneous, meaning that the statistical properties of the user channels are identical. For communications between the users and the BS, the collision channel model is assumed [19, 20]. In particular, the BS can receive at most one packet in each time slot and a collision occurs if there is more than one user transmitting at the same time. At the end of a time slot, the BS broadcasts a ternary feedback to all the users to indicate the outcome of just ended time slot. The feedback message can be 0, 1 or e, representing the idle, success, or collision outcomes, respectively.

Note that, in this chapter, we assume a full-load scenario. That is, the users always have data to transmit. With this assumption, stability issue will not be considered. Instead, we direct our attention on studying the system throughput (in the other words, the probability of successful transmission) using a reservation-type protocol.

In the original CA-ALOHA framework [6, 15], the system employs a binary transmission scheduling policy. A user decides to transmit its data based on the outcome of the comparison between its channel gain and a threshold H_t. The threshold value is chosen such that the average transmission probability s, i.e., the probability that the channel gain is above H_t, is equal to $\frac{1}{N}$. More importantly, it is often assumed that the channel states are independent in time domain and each user makes the transmission decision purely based on its channel state at the beginning of each time slot. In this chapter, we consider a scenario where the users experience correlated fading over time. As a temporal correlation is considered, it is likely that a user with good channel state in a given time slot can have its channel condition unchanged in subsequent slots. Therefore, collisions could be reduced if that user can reserve the communication channel in a number of time slots provided that its channel is still 'GOOD'. In this section, the correlated fading channel model is explained.

3.2.2 Channel Model

As we consider a homogeneous system, all the wireless links between the users and the BS have the identical channel statistics. The channel condition over time can be modeled as a two-state Markov chain as shown in Fig. 3.1. In each time slot, the 'GOOD' or 'BAD' (G or B) state indicates that the user has channel gain greater or smaller than a threshold value H_t. As mentioned, the probability of the channel gain greater than H_t is $s = \frac{1}{N}$. Consequently, the steady-state probabilities of state G and B are given by $\pi_G = s$ and $\pi_B = 1 - s$, respectively. This channel model is in fact a simplified finite-state Markov channel model that can be found in [65, 66]. Denote by p_{GB} and p_{BG} the transition probabilities from G to B and from B to G, respectively. It can be shown that the transition probabilities are given by

$$\begin{aligned} p_{GB} &= r(1-s); \\ p_{BG} &= rs, \end{aligned} \tag{3.1}$$

for some $r \in (0, 1]$. Here, r is the correlation factor of the channel model. Note that when $r = 1$, the channel is memoryless. Smaller values of r representing the higher correlations between the channel gains in two consecutive time slots. In reality, the exact value of r depends on the specification of an underlying practical scenario. For example, under the same time coherence, the longer slot duration will lead to a decrease in correlation between slots and hence larger r.

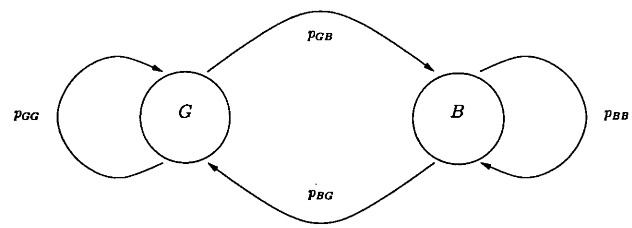

Figure 3.1: Model C: Markov chain modeling temporal correlation for the wireless communication channel between a user and the BS

3.3 Reservation-Type Protocol

Our simple reservation-type protocol is described as follows. If the outcome of the previous time-slot is not successful, i.e., on the reception of 0 or e feedback, all the users in the system contend for the channel as in the conventional CA-ALOHA [6]. However, when the outcome is successful (a feedback message "1" is broadcast to all users), the channel can be *reserved* by the transmitting user. The channel reservation remains valid for as long as the reserving user still experiences a good channel state. During this time, other users are not allowed to access the channel, thus no collisions happen.

In the description above, we assume that the inactive user who has 'GOOD' channel condition in a time slot know if the medium is reserved or has been released so that they can start contending for the channel later on. Effectively, this assumption makes the scheme analogous to the R-ALOHA with the *end-of-use* flag included [60, 61]. The mechanism facilitating this principle is as follows. The reserving user transmits immediately at the beginning of the time slot it would like to reserve without hesitation. Meanwhile, the waiting users sense the medium for a short duration, called a mini-slot. A *busy* sensing outcome at the end of the mini-slot indicates that the channel is reserved for the communication. In this case, the waiting users back-off (defer the transmissions in order not to interfere with the current transmission) and try again in the next time slot. Note that, the idea of using a mini-slot is not new and has appeared, for example, in [67] and more recently in [6, 8]. It is often assumed that the duration of mini-slot is very small compared to the duration of time slot, therefore, the time wasted for mini-slot sensing is negligible.

Intuitively, the proposed reservation protocol can reduce the collision probability between the users, hence, the system performance is improved. However, it does have one undesirable side effect. Once the user reserves the channel, other users have to wait for the channel being released before they can start contending for the channel, thereby causing the additional delay. In other words, the proposed protocol is not suitable for delay constrained applications. Nevertheless, statistically, all users experience independent channel conditions. Therefore, in longterm, the fairness of channel access is guaranteed.

3.4 System Throughput Based on Markov Chain Analysis

The technique we employ to study the system performance in this chapter is a well-known *Markov Analysis* [60, 61]. In this section, we formulate the Markovian models that are necessary for calculation of the system throughput. Firstly, to distinguish between the Markov chain models, we refer to the Markov chain representing the channel state of each user in Fig. 3.1 as Model C with state space $\{G, B\}$.

The entire system can be viewed as a $(N+1)$-states Markov chain, referred to as Model $D^{(N)}$ with state space $\{0, 1, \ldots, N\}$ as depicted in Fig. 3.2 [1]. In a

[1] The super-script (N) indicates that we are considering N users in total.

given time slot, the system being in state i implies that there are i users in G state. Throughout this chapter, since we assume the system in a steady-state, the time slot indexes can be dropped.

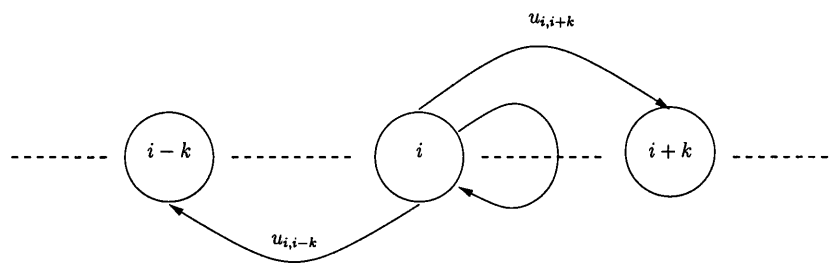

Figure 3.2: Model D: Markov chain modeling the number of users in 'GOOD' state in given a time slot

Denote by $\phi^{(N)} = [\phi_0^{(N)}, \phi_1^{(N)}, \ldots, \phi_N^{(N)}]$ the steady-state distribution of Model $D^{(N)}$. Since $\phi_i^{(N)}$ is the probability that i users are in state G, $\phi_i^{(N)}$ has a binomial distribution with the parameters N and $s = \frac{1}{N}$, i.e.,

$$\phi_i^{(N)} = \binom{N}{i} \frac{1}{N^i}(1 - \frac{1}{N})^{N-i}, \ 0 \leq i \leq N. \tag{3.2}$$

It can be seen from (3.2) that the steady-state distribution of $D^{(N)}$ is independent of the correlation factor r. In addition, we also note that the throughput of the original CA-ALOHA is equal to the steady-state probability of state 1, i.e., $\phi_1^{(N)}$. The average probability of having exactly one user in G state in a given time slot is given by

$$T_{CA-ALOHA} = \phi_1^{(N)} = (1 - \frac{1}{N})^{(N-1)}. \tag{3.3}$$

Let $\mathbf{U}^{(N)}$ be the transition matrix of Model $D^{(N)}$. The (i, j)th element, denoted by $u_{i,j}^{(N)}$, is the one-step transition probability from state i to state j. The transition probabilities are obtained as follows:

$$u_{i,i+k}^{(N)} = \sum_{j=0}^{\min(i,N-i-k)} \binom{i}{j} p_{GB}^{j}\, p_{GG}^{i-j} \binom{N-i}{j+k} p_{BG}^{j+k}\, p_{BB}^{N-i-j-k}$$
$$0 \leq i \leq N,\ 0 \leq k \leq N-i; \tag{3.4}$$
$$u_{i,i-k}^{(N)} = \sum_{j=0}^{\min(i-k,N-i)} \binom{i}{j+k} p_{GB}^{j+k}\, p_{GG}^{i-j-k} \binom{N-i}{j} p_{BG}^{j}\, p_{BB}^{N-i-j}$$
$$0 \leq i \leq N,\ 0 \leq k \leq i. \tag{3.5}$$

In order to calculate the system throughput of the reservation-type protocol, we consider another Markovian model which necessitates a state description consisting of the outcome of the transmission in each time slot. The third Markov chain is referred to as Model E. In this model, each state has two parameters, which is denoted by (i, f), where i, $i \in \{0, 1, \ldots, N\}$, indicates the number of users in G state during the given time slot and $f \in \{F, S\}$ indicates the outcome of the slot. Here, S and F denote the successful (feedback message from BS is 1) and the unsuccessful outcomes (feedback message from BS is 0 or e), respectively. A graphical presentation of Model E is given in Fig. 3.3. Note that this model has $2N$ states in total since $(0, S)$ and $(1, F)$ are invalid states [1].

Denote by $\psi = [\psi_{(0,F)}, \psi_{(1,S)}, \psi_{(2,F)}, \psi_{(2,S)}, \ldots, \psi_{(N,F)}, \psi_{(N,S)}]$ the steady-state distribution of Model E. The throughput of the reservation protocol is then obtained by

$$T_{Res-CA-ALOHA} = \sum_{i=1}^{N} \psi_{(i,S)}, \tag{3.6}$$

which can be graphically seen as a sum of the probabilities of states bounded by the dashed curve in Fig. 3.3. In addition, it can be seen with the aid of the

[1] In Section 3.2, we assumed a standard ALOHA-type environment, i.e., collision channel and ternary feedback. However, up to this point, we see that the reservation-type protocol can operate with binary feedback of type "S - success" and "F - failure".

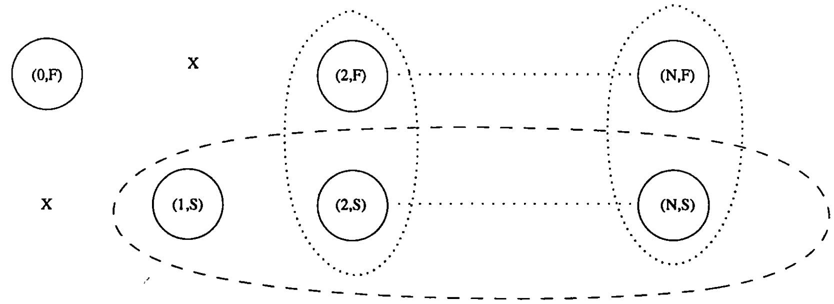

Figure 3.3: Model E: Markov chain indicating the number of users in 'GOOD' state in a given time slot and the outcome of that slot

dotted curves in Fig. 3.3 that

$$\begin{aligned} \phi_0 &= \psi_{(0,F)}; \\ \phi_1 &= \psi_{(1,S)} = T_{CA-ALOHA}; \\ \phi_i &= \psi_{(i,F)} + \psi_{(i,S)}, \text{ for } 2 \le i \le N. \end{aligned} \tag{3.7}$$

From (3.6) and (3.7), we have the following observation.

Observation 1 *The throughput of the reservation scheme is greater than or equal to that of the CA-ALOHA. It is predicted that the equality holds if* $r = 1$, *which is the case when the channel states are independent from one time slot to another (no correlation).*

Note that the performance gain of the reservation scheme is due to the fact that the probability of collision is reduced.

Our aim is to calculate the steady-state distribution of Model E. Denote by $\mathbf{V}$ the one-step transition matrix of Model E. Each element $v_{(i,m),(j,n)}$, $i, j \in \{0, 1, \ldots, N\}$, $m, n \in \{F, S\}$, of $\mathbf{V}$, which represents the transition probability from state (i, m) to (j, n), is obtained as follows.

Case 1: The current state is (i, F), $i \neq 1$. The transition from (i, F) to (j, S), $2 \leq j \leq N$, is invalid, i.e., $v_{(i,S),(j,F)} = 0 \; \forall \; i \neq 1, \; 2 \leq j \leq N$, since having the successful transmission after a channel contention period can only occur if there is a single user with good channel condition. Thus, the possible transitions from (i, F), $i \neq 1$, can only be $(i, F) \rightarrow (j, F)$, $j \neq 1$, and $(i, F) \rightarrow (1, S)$ with the probabilities:

$$\begin{aligned} v_{(i,F),(1,S)} &= u_{i,1}^{(N)}, \; i \neq 1; \\ v_{(i,F),(j,F)} &= u_{i,j}^{(N)}, \; i, j \neq 1, \end{aligned} \tag{3.8}$$

where $u_{i,j}^{(N)}$ is computed as in (3.4) and (3.5).

Case 2: The current state is (i, S), $i > 0$. The transition from (i, S) to (N, F) is not allowed, i.e., $v_{(i,S),(N,F)} = 0 \; \forall \; i \in \{0, 1, \ldots, N\}$, since having unsuccessful transmission in a slot after a reservation period can only occur if the reserving user releases the channel, that is, its channel condition becomes bad and consequently, there are at most $(N-1)$ users in 'GOOD' state. The other transition probabilities are given by

$$\begin{aligned} v_{(i,S),(1,S)} &= p_{GG} \, u_{i-1,0}^{(N-1)} + p_{GB} \, u_{i-1,1}^{(N-1)}; \\ v_{(i,S),(j,F)} &= p_{GB} \, u_{i-1,j}^{(N-1)}, \quad j \neq 1, \; j \neq N; \\ v_{(i,S),(j,S)} &= p_{GG} \, u_{i-1,j-1}^{(N-1)}, \quad 1 < j \leq N, \end{aligned} \tag{3.9}$$

where $u_{i,j}^{(N-1)}$ is the transition probability from state i to state j of Model $D^{(N-1)}$, which is formed by considering a system with $N-1$ users (excluding the user who is reserving the channel) and the states are the number of users in G state. The transition probability $u_{i,j}^{(N-1)}$ can be calculated as for Model $D^{(N)}$ using (3.4) and (3.5).

It is well-known that steady-state distribution $\boldsymbol{\psi}$ of Model E is the normalized eigenvector associated with the eigenvalue 1 of $\mathbf{V}$ [68]. The normalization is necessary to guarantee that $\sum_{i,j} \psi_{(i,j)} = 1$. Finally, the system throughput can be calculated from $\boldsymbol{\psi}$ using (3.6).

3.5 Numerical Results

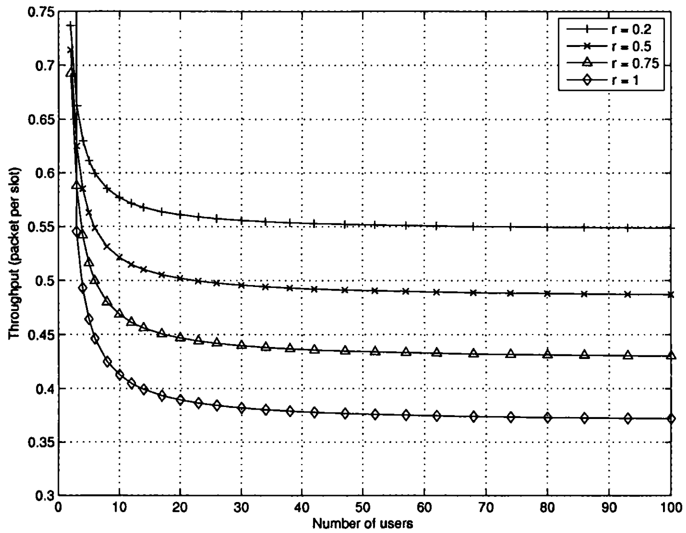

Figure 3.4: Throughput vs. number of users of CA-ALOHA system with a reservation-type protocol

Based on calculations in the previous section, we perform numerical study to obtain the performance of the proposed reservation scheme. Fig. 3.4 shows the system throughput $T_{Res-CA-ALOHA}$ versus the number of users N for different values of the correlation factor r. For given r, we observe that the system throughput approaches a stable value when N is sufficiently large, which is similar to the traditional ALOHA. Specially, when $r = 1$, corresponding to the case of memoryless channel, the stable throughput approximates 0.368, the well-known $\frac{1}{e}$ fraction for the ALOHA random access protocol. However, as r is close to 0, i.e., the channel is highly correlated, the system performance is significantly improved. This observation is further supported by Fig. 3.5, which shows the average throughput of the reservation protocol for different values of the correlation factor r given that there are $N = 100$ users. As expected, the smaller r the better performance can be achieved.

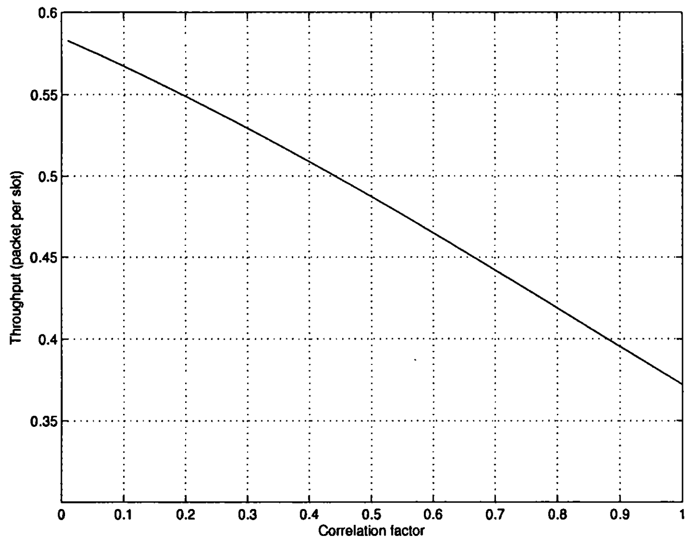

Figure 3.5: Throughput vs. correlation factor of CA-ALOHA system with a reservation-type protocol and a large number of users

3.6 Conclusions

In this chapter, we proposed a reservation-type protocol for random access networks assuming a cross-layer design context. In our proposal, by taking the advantage of knowledge of the channel correlation statistics, the system can achieve higher throughput by reducing the probability of collisions between users when one of them transmitted successfully and its channel is still in 'GOOD' state. Numerical study showed that when there are no correlations, the system throughput agrees with the conventional CA-ALOHA. However, the improvement of the system performance can be significant when the channels are more temporally correlated.

Chapter 4

Exploiting Idle Channels in Opportunistic Multichannel ALOHA

4.1 Introduction

As mentioned in the previous chapter, it is well-known that multiuser diversity can be exploited in wireless communications [5]. The underlying idea is that in wireless communication systems, the more users are present, the more likely that some users are in very good state at any time due to the random fading. Instead of mitigating channel fluctuations, the transmissions can be scheduled to/from the users with favorable channel conditions so that the total the system throughput increases with the number of users. This opportunistic scheduling approach has been presented in a number of recent works on the downlink communications [56], ad-hoc networks [57], and multi-antenna systems [16].

The opportunistic scheduling often needs to be realized in a centralized manner in which case the scheduling agent at the base station (BS) in a cellular network or the access point (AP) in a wireless LAN must know fading level of every users. This could be obtained, for example, by requiring the users to estimate their channel gains and forward the information to the central scheduler.

However, in a large network with many users, this method will not scale well and the delay in conveying channel state information (CSI) to the scheduler linearly grows with the number of users, a critical limiting factor in terms of the performance. In [6], Qin and Berry consider how to exploit the multiuser diversity in a distributed system. The scenario is the uplink and it is assumed that each user only has knowledge of its own fading level but no knowledge of the channel states of other users in the network. The users transmit with the transmission probability that is chosen based on their channel gains by a scheduling algorithm that jointly addresses both the PHY and MAC issues in a wireless network.

In particular, given the distributed knowledge of the channel conditions, a user will transmit when its channel gain is above a threshold H_t where H_t is chosen so that the overall transmission probability is equal to the desired value p, i.e., $H_t = \bar{F}_H^{-1}(p)$ where $\bar{F}_H(H_t) = \int_{H_t}^{\infty} f_H(h)dh$ denotes the channel gain's complementary distribution function. It can be shown that for an optimal value of H_t, the throughput of the proposed *channel aware* ALOHA (CA-ALOHA) scheme increases with the number of users at the same rate as in the centralized scheme. Asymptotically, the ratio of the throughput of the CA-ALOHA to the throughput of the centralized scheme is equal to $\frac{1}{e}$, the same well-known ratio achieved by the standard slotted ALOHA system. This can be interpreted as saying that the only loss due to distributed channel knowledge in fading environment with the opportunistic transmissions is the loss due to contention for the channel where as the multiuser diversity can still be exploited.

The idea of the CA-ALOHA is further studied for the uplink clustered OFDM-based multichannel slotted ALOHA system by Bai and Zhang [7]. In this paper, a system with M channels is considered and it is assumed that all channels are homogeneous and independent. The *opportunistic multichannel* ALOHA (OMC-ALOHA) scheme which targets both the multiuser and the multichannel diversities is proposed. At the beginning of a time slot, if the user has more than

m channels with the channel gains greater than the pre-determined threshold level H_t, it transmits its data immediately using the best m channels. Otherwise, the user keeps silent in this time slot. Similar to the CA-ALOHA, the overall transmission probability given the channel gain distribution can be obtained as

$$p = \sum_{i=m}^{M} \binom{M}{i} F(H_t)^{M-i} \big(1 - F(H_t)\big)^i, \tag{4.1}$$

where $F(.)$ denotes the common channel gain cummulative distribution function (CDF).

Given M, m, p and N, the number of users in the system, the long term average number of successful transmissions per slot is

$$E[X] = Nmp\left(1 - \frac{m}{M}p\right)^{N-1}, \tag{4.2}$$

where X is a random variable representing the number of channels bearing a successful transmission in each slot.

The throughput of the OMC-ALOHA scheme is the product of the average number of successful transmission channels and the average number of transmissions per channel. For a given number of channels M, with a large number of users N, it is plausible to sub-optimally set $m = 1$, i.e., each user should only transmit in one channel [7]. In this case, a sub-optimal value of the transmission probability is $p = \frac{M}{N}$. It can be shown that the OMC-ALOHA is an effective random access scheme in the sense that the only loss in the throughput compared to the optimal centralized scheme is due to contentions for the channel. This also means that there is always wastage of the capacity due to a number of channels being idle in each slot as inherited from the nature of the ALOHA based random access since the randomness not only occurs in the time domain but also in the frequency domain (across the channels).

In this chapter, we propose a simple extension to the transmission policy so that the idle channels can be exploited. The basic idea is to allow users who have sufficiently good channel conditions to access the idle channels. Assuming that the duration of a time slot is long enough, the time slot can be further divided into a number of mini-slots. At the beginning of each time slot, all users opportunistically contend for channels using the threshold level H_1, exactly as in the case of the OMC-ALOHA. The users who are not transmitting (*idle users*) identify the idle channels by performing carrier sensing. Idle channels can be then accessed by the idle users in subsequent mini-slots using a sequence of the threshold levels H_k, $k \geq 2$. Later, it will be shown that the proposed scheme preserves the multiuser and multichannel diversities of the OMC-ALOHA while exploiting the network resources more efficiently.

The rest of this chapter is organized as follows. Section 4.2 describes the uplink multichannel system model. In Section 4.3, the transmission scheme exploiting the idle channels in the OMC-ALOHA is proposed. In addition, we provide the theoretical limit of our proposed scheme. Mathematical derivation of the threshold levels is performed in Section 4.4. Based on the resulting formula, numerical study is presented in Section 4.5 to show the performance of our proposal. Finally, Section 4.6 provides some concluding remarks on this chapter.

4.2 System Model

We adopt the random access framework in [7], that is the uplink clustered OFDM-based multichannel slotted ALOHA system where N homogeneous users communicate with a single BS through M multiple parallel Rayleigh block fading channels. The users experience independent fadings across the channels and the channel conditions are assumed to be invariant within one slot duration but vary randomly from one slot to another. The channel gains are therefore modeled as a set of independent and identical distributed (IID) exponential random variables with the CDF

$$F(h) = 1 - \exp\left(-\frac{h}{h_0}\right), \tag{4.3}$$

where h_0 is the average channel gain.

On the user side, the D-CSI knowledge is assumed. At the beginning of a time slot, each user always has access to the gains of all M channels between itself and the BS. This can be facilitated by allowing pilot symbols to be sent from the BS for the channel estimation prior to the uplink transmissions [6, 8]. In practice, the pilots need to be broadcast in the same coherence bandwidth as the uplink channel, for example, using a time division duplex (TDD) system. Theoretically, this distributed approach only requires one-half of a round trip times for the channel measurement, which is significantly less than for the centralized channel measurement scheme in large systems, and it scales as the number of users increases.

As in the standard ALOHA-type environment [19, 20, 28], we assume the collision reception model with instantaneous (0,1,e) feedback broadcast on each channel at the end of each time slot. In addition, we assume a full load scenario. That is, the users always have data to transmit. In this case the stability issue will not be considered.

Note that, in our analysis, we assume that all users are operating under identical conditions. In the literature, the opportunistic transmission schemes for *heterogeneous system*, in which users may have different channel distribution, have been studied [15, 7]. It is shown that the *proportional fairness* [16] is guaranteed [1] if each user behaves exactly as if it is still in a homogeneous system, that is, the user determines a threshold for itself based on the D-CSI knowledge and based on knowledge of the number of users in the system.

As discussed by the authors in [7], the problem of choosing the optimal values for m and the overall transmission probability p to maximize the throughput is

[1] The conditions for the proportional fairness are: i.) the sum of logs (or equivalently, the product) of throughput is maximized, and ii.) allowing all user at least a minimal level of service.

non-convex and can only be solved by numerical methods. However, when the number of users is significantly greater than the number of channels, it is plausible that each user should transmit in one channel ($m = 1$) and the transmission probability can be sub-optimally approximated as $p = \frac{M}{N}$. Therefore, to illustrate the idea of exploiting the idle channels, we assume $N >> M$ and we the sub-optimal values $m = 1$ and $p = \frac{M}{N}$ for our analysis so that the problem becomes mathematically tractable. The proposed opportunistic transmission scheme using a sequence of mini-slots to exploit the idle channels is described in the next section. It will be seen that our scheme is simple and can be applied also to other cases, given the optimal values of m and p.

4.3 Exploiting Idle Channels in OMC-ALOHA with Mini-Slots

In the OMC-ALOHA scheme [7], given N, M, m and the overall transmission probability p, the long term average number of successful transmissions per slot is obtained using Eq. (4.2). Let Y be the random variable denoting the number of idle channels in each slot. The expectation of Y can be derived in a similar manner as for the average number of idle users in each slot, i.e.,

$$E[Y] = M\left(1 - \frac{1}{N}\right)^N, \tag{4.4}$$

where we fixed $m = 1$ and used the transmission probability $p = \frac{M}{N}$. In order to exploit the idle channels, we consider the following simple variation to the transmission scheme.

Assume that the duration of a time slot T is long enough. Then the time slot can be divided into a number of mini-slots of duration β as depicted in Fig. 4.1.

- At the beginning of the time slot (also at the beginning of the first mini-slot), all users *opportunistically* contend for the channels with respect to

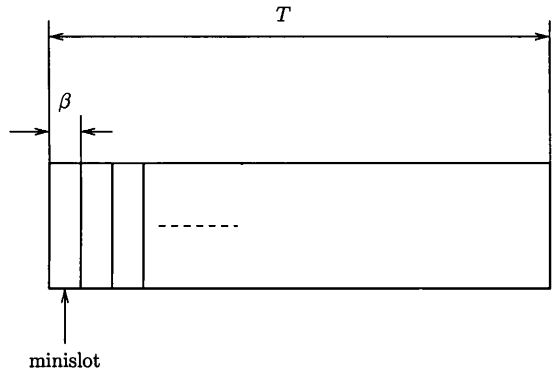

Figure 4.1: A time slot made up of several mini-slots

the threshold level H_1. The users that are not transmitting (the idle users) listen to all channels for the duration of a mini-slot. If the channel is used by one or more users, it is seen as being busy. Otherwise, the channel is idle and shall be accessible.

- At the beginning of the subsequent mini-slots, the idle channels and the idle users form a new *virtual system*. The users of such the *virtual system* transmit their data using the OMC-ALOHA transmission policy described above with the new threshold levels H_k, $k \geq 2$.

As mentioned in the previous chapter, the idea of using mini-slots is not new and was used by Qin and Berry in the splitting algorithm [6, 8]. The difference between their work and our proposed scheduling policy lies in the fact that the splitting algorithm resolves the collisions in the single channel system, i.e. $M = 1$, while we aim to exploit the idle channels in a multichannel setup.

In the asymptotic case, we have

$$\lim_{N\to\infty} E[X] = M\left(1 - \frac{1}{N}\right)^{N-1} \approx \frac{M}{e}, \tag{4.5}$$

$$\lim_{N\to\infty} E[Y] \approx \frac{M}{e}. \tag{4.6}$$

These equations imply that, in a large system, there is a fraction of $\frac{1}{e}$ number of channels (36.79%) that are successfully used as well as the same number of channels is not utilized after the initial transmissions in the first mini-slot [1]. This is equivalent to saying that a fraction $\frac{1}{e}$ of the system capacity is exploited and a fraction $\frac{1}{e}$ is wasted due to the idle channels (the rest is wasted due collisions). Here, the system capacity S_c is understood to be the throughput of the optimal centralized scheme. By applying the OMC-ALOHA for the second mini-slot as proposed, one would expect that an improvement of $\frac{S_c}{e^2}$ can be made from exploiting the idle channels. In general, the kth mini-slot can yield an extra fraction $\frac{S_c}{e^k}$, which decays exponentially with k. However, as we assumed that the duration of a mini-slot is β, one would expect a fraction $\frac{\beta}{T}$ in transmission time of the users after each mini-slot. Let S_K denote the throughput of the system after using K mini-slots. We have that

$$\begin{aligned} S_K &= \sum_{k=1}^{K} \frac{S_c}{e^k}\left(1-(k-1)\frac{\beta}{T}\right) \\ &= \sum_{k=1}^{K} \frac{S_c}{e^k} - \frac{\beta}{T}\sum_{k=1}^{N} K\frac{k-1}{e^K}S_c, \end{aligned} \tag{4.7}$$

in which the first term is the performance improvement, which decays exponentially with the number of mini-slots, and the loss due to sensing delay is accounted for in the second term.

We now consider the limiting case $K \to \infty$ when the slot duration is sufficiently long, i.e., assuming that $\frac{\beta}{T} \to 0$. This assumption is valid since in the carrier sensing systems, the sensing delay is approximated as the worst case propagation delay from one user to another (tens μs) which is much smaller than the duration of a slot (several ms). Also, assuming that the system has the extremely large number of users and channels. In this case, the number of mini-slots used

[1] These equations are expectable due to the contention based nature of the OMC-ALOHA scheme and provide good approximation even when N is as small as 20.

K can be large. We have that

$$\lim_{K \to \infty} S_K = \frac{S_c}{e-1} - \frac{\beta}{T}\frac{S_c}{(e-1)^2}. \tag{4.8}$$

Since $\frac{\beta}{T} \to 0$, the second term of (4.8) is negligible compare with the first term. Consequently, the theoretical limit of the proposed schene is $\frac{S_c}{e-1} \approx 0.58S_c$, a 63% improvement to the original OMC-ALOHA.

In practical implementations, the number of mini-slots in use shall be small since the performance gain decays exponentially with the number of mini-slots.

4.4 Threshold Levels Calculation

In order to let the idle users access the idle channels at the beginning of the kth mini-slot, the threshold levels H_k $(k \geq 1)$ need to be determined. This subsection contains mathematical derivations of the threshold levels based on the sub-optimal choice of $m = 1$. Thus, at the beginning of the kth mini-slot, the user transmits if it has at least one channel in the set of idle channels with the channel gain greater than H_k.

The relationship between the threshold H_k and the transmission probability p_k in mini-slot k is given as [7]

$$p_k = \sum_{i=1}^{M_k} \binom{M_k}{i} F_k(H_k)^{M_k - i}\big(1 - F_k(H_k)\big)^i, \tag{4.9}$$

where M_k is a random variable denoting the number of available channels at the beginning of the kth mini-slot and $F_k(h)$ is the CDF of the channel gains at the beginning of the kth mini-slot. Note that in this equation, $p_k = \frac{M_k}{N_k}$ (also sub-optimal [7]) where N_k is a random variable denoting the number of available users at the beginning of kth mini-slot. In order to calculate H_k from (4.9), it is necessary to know M_k, N_k and to specify the function $F_k(.)$.

For $k = 1$, $p_1 = \frac{M}{N}$ and $F_1(.)$ is the CDF given in (4.3) and H_1 can be obtained by inverting (4.9). At the beginning of the kth mini-slot, $k > 1$, the set of channel gain samples only contains the samples that are smaller than H_{k-1}. This statistical truncation makes the channel gain samples a singly right truncated exponential distribution with the CDF

$$F_k(h) = F(h|h \leq H_{k-1}) = \frac{F(h)}{F(H_{k-1})} \\ = \frac{1 - \exp\left(-\frac{h}{h_0}\right)}{1 - \exp\left(-\frac{H_{k-1}}{h_0}\right)}. \quad (4.10)$$

Let X_k and Y_k denote the number of successful channels and the number of idle channels after k mini-slots, respectively. As generalization of (4.5) and (4.6), the average number of successful and idle channels in the kth mini-slot is equal to $E[X_k] = E[Y_k] = \frac{E[M_k]}{e}$ and therefore,

$$E[M_k] = E[Y_{k-1}] = \frac{M}{e^{k-1}}. \quad (4.11)$$

Eq. (4.11) shows that the number of available channels in the *virtual system* exponentially decays with k. Thus, even with a large number of channels, the number of mini-slots required is still relatively small. For example, assuming that there are 1024 channels in the original system, there is less than one idle channel (on average) left after 7 mini-slots.

Let N_k and U_k be random variables denoting the number of idle users at the beginning of the kth mini-slot and the number of users starting their transmission in that mini-slot, respectively. It can be shown that U_k is binomially distributed with probability $p_k = M_k/N_k$ in a population set N_k. Therefore, $E[U_k] = p_k E[N_k] = \frac{E[M_k]}{E[N_k]} E[N_k] = E[M_k]$ and the average number of idle users at the beginning of the $(k+1)$th mini-slot is

$$E[N_{k+1}] = E[N_k] - E[U_k] = E[N_k] - E[M_k] \\ = N - \sum_{l=1}^{k} \frac{1}{e^{l-1}} M. \quad (4.12)$$

From (4.9) and (4.10), the thresholds are obtained by

$$H_k = -h_0 \times \ln \left\{ 1 - \left[1 - \exp\left(-\frac{H_{k-1}}{h_0} \right) \right] \left(1 - \frac{E[M_k]}{E[N_k]} \right)^{\frac{1}{E[M_k]}} \right\}, \tag{4.13}$$

for $k > 1$, where $E[M_k]$ and $E[N_k]$ are given in (4.11) and (4.12), respectively. Note that, the values of the transmission probabilities p_k and the threshold levels H_k only change if the system population changes. The proposed scheme maintains the simplicity of the ALOHA and the thresholds can be calculated prior to the network set up for use throughout the network lifetime.

4.5 Numerical Results

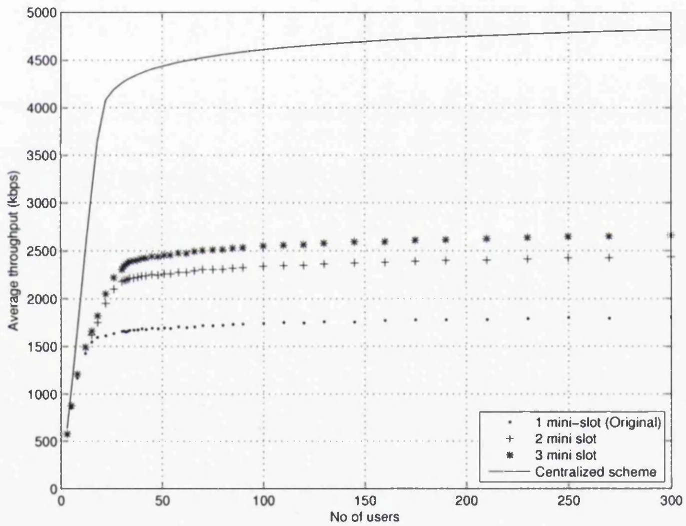

Figure 4.2: Throughput of the proposed scheme exploiting idle channels vs. the number of users

In this section, we present numerical results to verify the predicted performance of the proposed scheme. Fig. 4.2 plots the throughput of the system assuming 1 mini-slot (original OMC-ALOHA), 2 mini-slots, 3 mini-slots and assuming the optimal centralized scheme. Here, we choose the number of channels

$M = 20$ and the number of users N varies from 1 to 300. The total system bandwidth is set to 1MHz and the average signal to noise ratio (SNR) is set to 0dB. In addition, we use the Shannon capacity formula to represent the user transmission rates. Note that for a small number of users, the curves do not show the advantage of our proposal since the values of $m = 1$ and $p = M/N$ are far from the optimal. However, as soon as the number of users becomes greater than the number of channels, the curves show that a significant performance improvement can be gained by exploiting the idle channels (over 30% improvement with 2 mini-slots), while preserving both the multichannel and the multiuser diversities of the OMC-ALOHA scheme.

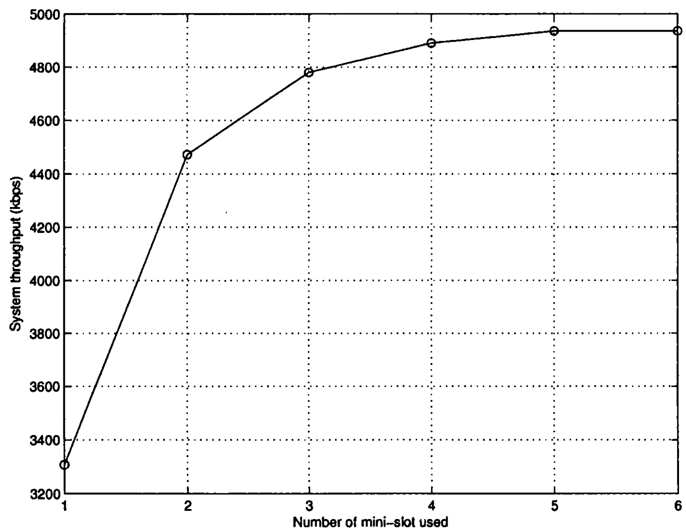

Figure 4.3: Throughput of the proposed scheme exploiting idle channels for the different number of mini-slots used

To see the trend of the throughput enhancement as the number of mini-slots increases, we numerically study a system with a large number of users ($N = 2000$) and the number of channels is set to $M = 1000$. As can be seen in Fig. 4.3, by using the first 3 mini-slots, a significant improvement can be observed. However, the throughput gain slows down gradually in the subsequent mini-slots and the system performance converges to a limit as discussed in Sec. 4.3. Thus, we can say that the number of mini-slots required can be small to achieve a sufficient performance gain while not incurring too much overhead, e.g. 3 mini-slots in this case.

4.6 Conclusions

In this chapter, we studied a simple extension to the OMC-ALOHA protocol. Our proposal allows to access the idle channels in each time slot, and thus to reduce the wastage during random transmissions. The scheme preserves both the multiuser and multichannel diversities of the OMC-ALOHA and exploits the network resources more efficiently. Performance analysis showed that the scheme can asymptotically improve the system performance by up to 63%.

An analytical formula to calculate the threshold levels was derived, assuming that the system consists of a large number of users. Our numerical study of the algorithm confirmed that a significant performance improvement can be gained even for a small number of users. When the number of users increases, the system throughput also increases at the same rate as with the centralized scheme, regardless of the number of mini-slots used.

Finally, we note that, the number of mini-slots required can be small since the throughput improvement slows down quickly and converges to a limit due to the fact that the number of idle channels decays exponentially with the number of mini-slots.

Chapter 5

A Novel Splitting Based Multiuser Selection Algorithm

5.1 Introduction

Finding an effective multiuser selection algorithm (MSA) for the resource allocation is a challenging problem in wireless communications with many other applications. For example, consider a simple wireless network with a large number of users. It is well known that the overall network performance can be improved by selecting users with the best channel qualities for data transmission and exploiting the so-called *multiuser diversity* [5]. This *opportunistic scheduling* approach has recently attracted a considerable amount of research works, especially with distributed algorithms since, typically, the channel information is only available locally. In practice, the centralized polling algorithms are not preferred because the resources they require to gather necessary information increases linearly with the number of users in the system.

In particular, for distributed scheduling, Qin and Berry proposed a splitting algorithm which aim at selecting the best users for transmission in random access networks [8]. In each step of the algorithm, only those users which channel gains lie between the upper and lower thresholds are allowed to transmit. The threshold levels are updated locally and independently after each step based on the outcome of the previous one which is available in the form of the broadcast feedback from the central base station (BS). It was shown that the average number of steps required by the algorithm to converge is less than 2.507, regardless of the size of the network and, this bound is independent of the actual fading distribution.

The splitting based selection algorithm was further studied by Shah et. al. [9]. In order to improve the efficiency of this algorithm, the authors introduced a new parameter called the *contention load factor*, and obtained an asymptotic analysis based on the associated Markov chain. A generalized version of this algorithm which aims at selecting more than one user was also studied. The results showed that there is a close connection between the splitting algorithm and the collision resolution algorithm [10, 11]. Putting these two problems under a unified view, we deem that there are some questions for discussion: i.) since only binary splitting with fair coin was considered in [9], a natural concern regarding the splitting using biased coin was left unanswered; ii.) variations suggested for the CRA such as V*-ary* splitting [47] and the *tree pruning* [48, 51] can be considered in the multiuser selection.

The rest of this thesis is used to tackle the first question above. In particular, Section 5.2 discusses the related work. The system model is described in Section 5.3 in conjunction with description of the novel distributed MSA that uses two design parameters called the *contention factor* and the *selection factor*. This proposed scheme is considered to be general while retaining the simplicity of the splitting algorithm. Section 5.4 is used to formulate of (recursive) equations for calculating the number of slots and the number of transmission required by the algorithm. We then derive the lower bounds on the expectation of these measures of interest in Section 5.5. Concluding remarks for this chapter are provided in Section 5.6.

We defer numerical studies of the results in this chapter to Chapter 6 after we perform deeper analysis for an asymptotic case. Furthermore, application of the proposed MSA will be demonstrated for a cooperative transmission scheme in a wireless sensor network in Chapter 7.

5.2 Related Works

To certain extent, the multiuser selection problem is strongly related to the *partitioning a sample with binary type questions* [69]. If members of the group (a sample set) are assigned with real observations (the metric) that are independent and identically distributed (IID) or at least exchangeable, for example, the channel conditions associated with the users in a wireless network, then it is desirable to find the user with the largest observation (or the q best users) by asking binary-type questions. The questions recursively split the set of users in consideration (the active set) into two subsets, such that the first subset (resp. the second subset) contains the users with affirmative answers (resp. negative answers). Designing the questions is equivalent to setting the probabilities of users joining the subsets which corresponds to the probabilistic view of the algorithm. If *full feedback* is available, that is, if after each step, it is known exactly how many users answered the question affirmatively (and thus join the first subset), Arrow et.al. [69] showed that for any starting group size and for any distribution, the minimal expected number of questions required to find the holder of the largest observation is less than 2.428. The reason that different distributions do not alter this result is because the cumulative distribution of any random variable is uniformly distributed in $(0, 1)$ to which the problem can be reduced to.

Later, Cohen et. al. [50] revisited this problem in the context of multiuser communication networks. Assuming that the number of affirmative answers is not known exactly but either it is 0, 1 or e (i.e., the *ternary feedback*, they found that the expected number of questions needed to find the best user is approximately 2.445 for a large number of users. This result is numerically close to that of [69] and can be interpreted that the loss caused by ternary feedback is marginal. Nevertheless, ternary feedback is of practical important in communications and has been applied widely [19, 20].

The results presented in [69] and [50] are obtained by adaptively adjusting the splitting probability after each step according to the outcome of the previous step that is fed back by the central entity (i.e., the base station (BS)). Recently, we studied simpler but still efficient version of the single user selection algorithm [70]. By optimally choosing the design parameters, our algorithm is numerically comparable with the classical results mentioned.

Broadly speaking, the splitting based MSA follows a *divide and conquer* procedure, which is also the principal idea of the CRA. Thus, the two algorithms are naturally connected. The main differences between them are the purposes they serve and the way they are being evaluated. While the splitting based MSA is evaluated mainly by its expected duration, the CRA belongs to a class of random access protocols and is often evaluated by the maximum stable throughput (MST) as well as the expected packet delay of the system.

In a strict sense, the term CRA refers to a class distributed algorithms that handle collisions by resolving them algorithmically as soon as they occur. In order to form a complete random access protocol, the CRA needs to be combined with a channel access algorithm (CAA). Looking into details, the protocol of this type is, in fact, an extreme case of the splitting based MSA. The CRA with time addressing and a blocked access is equivalent to a selection algorithm in which there is an infinite number of users and all the users need to be selected in the first come first serve (FCFS) order.

Another line of works related to the distributed selection problem is the *distributed council election.* Originated in the research on computer programming and networking (see [71] and the references therein), the problem is to elect a small number of representatives (the council) out of a (possibly large) group of anonymous candidates. During the election process, candidates do not communicate directly with each other. Instead, they can only send unicast messages containing their identity (ID) to the central entity. The central entity, in turn, is able to broadcast global feedback to the entire group. Formation of the council is decided based on the received unicast messages from the candidates. In general, studies on the distributed algorithm for council election pay particular interest to two measures: i.) the expected number of rounds for election; and ii.) the expected number of unicast messages needed. Note that, description of the system model shows some similarities between the distributed council election and the MSA. The differences lie in the problem statement and the research objectives, which leads to different analyses and optimization problems. Only when the desired council consists of a single member, the distributed council election problem collapses to a *distributed leader election* algorithm which is, in fact, equivalent to the single user selection mentioned above [50, 70].

5.3 System Model and the Splitting Based MSA

Throughout the thesis, we have been considering a familiar time-slotted multiple access model, that is, a network of n users communicate with a common BS over a collision channel. In particular, the BS can receive information from at most one user in each time slot and the collision occurs if there is more than one user transmitting at the same time. At the end of each time slot, a standard $(0, 1, e)$ ternary feedback, representing *idle, success* and *collision* outcomes, respectively, is broadcast to all users. Note that, for the multiuser selection problem and, specifically, for the splitting algorithm herein, transmitted packets are often small and only contain basic identity information of users who are contending for the channel.

Algorithm 1 Distributed Multiuser Selection Algorithm

Input: n, q, p_c, p_s

Output: *selected*;

```
selected ← 0; transmitted ← 0; lcount ← 1; gcount ← 1;
feedback ← 0; prevfeedback ← 0; N_sel ← 0; N_act ← N;
while N_sel < q do
  if feedback = 1 then
    N_sel ← N_sel + 1; N_act ← N_act − 1;
    if transmitted = 1 then
      selected ← 1;
    end if
  end if
  if !selected then
    if feedback = e then
      gcount = gcount + 1
      if lcount = 0 then
        lcount ← split(p_s);
      else
        lcount ← lcount + 1;
      end if
    else
      if feedback = 0 and prevfeedback = e then
        if lcount = 1 then
          lcount ← split(p_s);
        end if
      else
        gcount = gcount − 1;
        if gcount = 0 then
          lcount ← split(p_c / N_act);
        else
          lcount = lcount − 1;
        end if
      end if
    end if
    prevfeedback ← feedback;
    transmitted ← (lcount = 0);
    get(feedback);
  end if
end while
```

The rest of this thesis studies a generalized splitting based algorithm which aims at selecting q (desired) users from the original set of n users in a distributed manner. The basic principle of the splitting algorithm is to recursively split the active set into two subsets by allowing users to make decisions to transmit locally and independently at the beginning of each time slot using the chosen design parameters. The set of transmitting users form the first subset, which becomes the active set for the next time slot, conditioned on the event that an e feedback is broadcast. If the outcome is idle, the active set for the next time slot is the second subset, which is also the same as the active set of the previous slot. Every time we have a successful transmission, a user is selected. The algorithm terminates when exactly q users have been extracted. During the user selection process, the splitting probabilities are set based on two parameters: i.) the *contention load factor* p_c, similar to that of [9], and ii.) the *selection factor* p_s. The former is used in the *contention slots*, which is defined as the time slot when the number of users in the active set is deterministically known, while the latter is used when the number of users in the active set is characterized by a random variable. Formally, the selection procedure can be realized using Algorithm 1. This algorithm resembles the stack implementation of the CRA explained in Section 2.4 and is performed by each user in the system. Here, *lcount* and *gcount* are the local and global counters respectively that are maintained by the users. At the beginning of a time slot, the user transmits its packet if its local counter is at the top of the stack, i.e., $lcount = 0$. The counters are increased or decreased after each slot depending on the feedback message. Furthermore, the function *split(p)* is a splitting function, which returns value "0" (resp. "1") with the probability p (resp. $1-p$). Note that, similar to the MTA, this algorithm also uses the level skipping, that is, if a collision is followed by the idle slot, the second subset is automatically split into two. All users terminate the algorithm when the counter N_{sel} reaches q, telling them that no more selections are required [1].

[1]The last value of N_{sel} that a selected user recorded represents order that it is selected

Note that, unlike related works [50, 8, 9], we do not specify applications of the splitting algorithm. We limit the current work to the probabilistic view of the multiuser selection problem. How to define the desired users and how to choose the design parameters depends on the application scenario, i.e., the metrics associated with the users; this is currently outside the scope of this chapter. An application of the MSA will be demonstrated later in the thesis.

5.4 Recursive Equations

First, let $L_n^{(q)}(p_c, p_s)$ and $M_n^{(q)}(p_c, p_s)$, $p_c > 0$, $p_s \in (0, 1)$, be the random variables denoting the duration (in slots) of the MSA to select q users and the total number of transmission attempts (the total number of data packets sent out by the users) required throughout the selection process, respectively, given that there are n users in the original set, $n \geq q$. Here, the two design parameters are the contention load factor p_c and the selection factor p_s. Let $l_n^{(q)}(p_c, p_s)$ denote the expected length of the algorithm, i.e., the average number of slots required, and $\mu_n^{(q)}(p_c, p_s)$ denote the expected number of transmissions. We also define the following functions:

$$L_n^{(q)}(p_c, p_s) = l_n^{(q)}(p_c, p_s) = 0 \text{ if } q \leq 0, \tag{5.1}$$

$$L_1^{(1)}(p_c, p_s) = l_1^{(1)}(p_c, p_s) = 1, \tag{5.2}$$

$$M_n^{(q)}(p_c, p_s) = \mu_n^{(q)}(p_c, p_s) = 0 \text{ if } q \leq 0, \tag{5.3}$$

$$M_1^{(1)}(p_c, p_s) = \mu_1^{(1)}(p_c, p_s) = 1. \tag{5.4}$$

Let a *selection epoch* be defined as a part of the splitting algorithm which starts with a contention slot and ends when either all users transmitting in that slot have been selected or when exactly q users have been selected throughout the process, whichever comes first. Denote by $S_m^{(q)}(p_s)$ the duration of the selection epoch conditioned on m users transmitting in the contention slot, and the algorithm (still) needs to select q more users. Also, denote by $T_m^{(q)}(p_s)$ the total number of transmission attempts by m users in the selection epoch. These random variables are solely dependent on the selection factor p_s. Let $s_m^{(q)}(p_s)$ and $\tau_m^{(q)}(p_s)$ be expectations of these random variable. Similar to (5.1) and (5.3), we define,

$$S_m^{(q)}(p_s) = s_m^{(q)}(p_s) = 0 \text{ if } q \leq 0, \tag{5.5}$$

$$T_m^{(q)}(p_s) = \tau_m^{(q)}(p_s) = 0 \text{ if } q \leq 0. \tag{5.6}$$

Also, it shall be noted that

$$S_m^{(q)}(p_s) = S_m^{(m)}(p_s) \text{ if } q \geq m, \tag{5.7}$$

$$s_m^{(q)}(p_s) = s_m^{(m)}(p_s) \text{ if } q \geq m, \tag{5.8}$$

$$T_m^{(q)}(p_s) = T_m^{(m)}(p_s) \text{ if } q \geq m, \tag{5.9}$$

$$\tau_m^{(q)}(p_s) = \tau_m^{(m)}(p_s) \text{ if } q \geq m. \tag{5.10}$$

By definition, when there is no collision in a contention slot, the duration of the contention epoch is deterministically known, i.e.,

$$S_m^{(q)}(p_s) = s_m^{(q)}(p_s) = 1 \text{ if } m \in \{0, 1\}. \tag{5.11}$$

Likewise,

$$T_0^{(q)}(p_s) = \tau_0^{(q)}(p_s) = 0, \tag{5.12}$$

$$T_1^{(q)}(p_s) = \tau_1^{(q)}(p_s) = 1. \tag{5.13}$$

However, when the selection epoch starts with a collision, i.e., for $m \geq 2$, the specification of the algorithm yields the following recursions

$$S_m^{(q)}(p_s) = \begin{cases} 1 + S_Y^{(q)}(p_s) + S_{m-Y}^{(q-Y)}(p_s) & \text{if } Y > 0, \\ 1 + S_m^{(q)}(p_s) & \text{otherwise,} \end{cases} \tag{5.14}$$

and

$$T_m^{(q)}(p_s) = \begin{cases} m + T_Y^{(q)}(p_s) + T_{m-Y}^{(q-Y)}(p_s) & \text{if } Y > 0, \\ T_m^{(q)}(p_s) & \text{otherwise.} \end{cases} \tag{5.15}$$

where Y is a binomial random variable with the parameters m and p_s, denoting the number of users transmitting in the slot just after the collision.

The second lines of (5.14) and (5.15) take into account the fact that the algorithm ignores the predetermined collision after an idle slot and proceed to the next step of the algorithm immediately (i.e., level skipping). They induce the following recursions for the expectations

$$\begin{aligned} s_m^{(q)}(p_s) = {} & 1 + (1-p_s)^m s_m^{(q)}(p_s) \\ & + \sum_{k=1}^{m} \binom{m}{k} p_s^k (1-p_s)^{m-k} \left(s_k^{(q)}(p_s) + s_{m-k}^{(q-k)}(p_s) \right), \end{aligned} \tag{5.16}$$

$$\begin{aligned} \tau_m^{(q)}(p_s) = {} & m(1-(1-p_s)^m) \\ & + \sum_{k=0}^{m} \binom{m}{k} p_s^k (1-p_s)^{m-k} \left(\tau_k^{(q)}(p_s) + \tau_{m-k}^{(q-k)}(p_s) \right). \end{aligned} \tag{5.17}$$

Intuitively, in the event of collision of size m in a contention slot, $m \geq 2$, the algorithm splits the set of transmitting users into two subsets and evaluates them one after another. If the first subset has k users, then the second one must contain the remaining users, i.e., $m-k$. Conditions (5.5) and (5.6) ensures that recursions (5.14), (5.15), (5.16) and (5.17) are meaningful.

In addition to (5.14) and (5.15), the basic principle of the splitting algorithm also implies that

$$L_n^{(q)}(p_c, p_s) = S_X^{(q)}(p_s) + L_{n-X}^{(q-X)}(p_c, p_s), \tag{5.18}$$

$$M_n^{(q)}(p_c, p_s) = T_X^{(q)}(p_s) + M_{n-X}^{(q-X)}(p_c, p_s), \tag{5.19}$$

where X is a random variable denoting the number of users transmitting in the contention slot, which is binomially distributed with the parameters n and $\frac{p_c}{n}$.

Thus, the average number of slots required by the algorithm and the expected total number of transmissions respectively are given as follows:

$$l_n^{(q)}(p_c, p_s) = \sum_{m=0}^{n} \binom{n}{m} \left(\frac{p_c}{n}\right)^m \left(1 - \frac{p_c}{n}\right)^{n-m} \left(s_m^{(q)}(p_s) + l_{n-m}^{(q-m)}(p_c, p_s)\right), \tag{5.20}$$

$$\mu_n^{(q)}(p_c, p_s) = \sum_{m=0}^{n} \binom{n}{m} \left(\frac{p_c}{n}\right)^m \left(1 - \frac{p_c}{n}\right)^{n-m} \left(\tau_m^{(q)}(p_s) + \mu_{n-m}^{(q-m)}(p_c, p_s)\right). \tag{5.21}$$

Again, one can see that conditions (5.1) and (5.4) ensure that recursions (5.18), (5.19), (5.20) and (5.21) are meaningful.

Note that, (5.16), (5.17), (5.20) and (5.21), are recursive formulas to calculate the average durations of the algorithm and the selection epoch, as well as the expected number of transmissions required. That is, given the initial conditions (5.1) to (5.13), we can solve for $l_n^{(q)}(p_c, p_s)$, $\mu_m^{(q)}(p_s)$, $s_m^{(q)}(p_s)$ and $\tau_m^{(q)}(p_s)$ algebraically for any n, m, q. For example, the first few values of $s_m^{(q)}(p_s)$ are listed in Table 5.1. However, the calculations are algebraically lengthy and do not show the scalable nature of the algorithm. In the rest of this chapter, we derive some lower bounds on the measures of interest.

Table 5.1: First few values of $s_m^{(q)}(p_s)$

m \ q	1	2	3
2	$1 + \frac{1}{2p_s(1-p_s)}$	$2 + \frac{1}{2p_s(1-p_s)}$	
3	$1 + \frac{2+3p_s}{6p_s(1-p_s)}$	$2 + \frac{5}{6p_s(1-p_s)}$	$3 + \frac{5}{6p_s(1-p_s)}$

5.5 Lower Bounds of the Average Duration and the Expected Number of Transmissions

First, consider a selection epoch which starts with a collision of size m in the contention slot while the algorithm still needs to select only one user. By convention, the duration of the selection epoch and the number of transmissions required are denoted by $S_m^{(1)}(p_s)$ and $T_m^{(1)}(p_s)$, respectively. Note that, for any $q > 0$ and for any p_s, the initial condition (5.11) gives $S_0^{(q)}(p_s) = s_0^{(q)}(p_s) = 1$ and $S_1^{(q)}(p_s) = s_1^{(q)}(p_s) = 1$. Similarly, conditions (5.12) and (5.13) give $T_0^{(q)}(p_s) = \tau_0^{(q)}(p_s) = 0$ and $T_1^{(q)}(p_s) = \tau_1^{(q)}(p_s) = 1$. In all other cases, the following lemma gives lower bounds of the expected duration of the selection epoch to select one user and of the expected number of transmissions required for that epoch.

Lemma 1 *The expected duration of selection epoch to select one user out of a collision set of size m, $m > 1$, and the average number of transmission required for that epoch, respectively, are lower bounded as,*

$$s_m^{(1)}(p_s) > \log_{\frac{1}{p_s}}(m) + 1, \tag{5.22}$$

$$\tau_m^{(1)}(p_s) > \frac{m - p_s}{1 - p_s}. \tag{5.23}$$

Proof: Starting with a collision of size m in the first (contention) time slot, let A_i be the number of users left in the active set after the ith slot. Thus, $A_1 = m$ and

$$S_m^{(1)}(p_s) = \inf\{i \ : A_i = 1 | A_1 = m\}, \tag{5.24}$$

$$T_m^{(1)}(p_s) = \sum_i A_i. \tag{5.25}$$

Note that the sequence $\{A_i\}$ is a Markov chain, that is, given $A_i = a$, A_{i+1} is independent of A_j for $0 < j < i$. Due to the nature of the splitting algorithm with the level skipping, we have

$$\begin{aligned}\mathbb{E}[A_{i+1}|A_i = a] &= \sum_{k=1}^{a} \binom{a}{k} p_s^k (1-p_s)^{a-k} k + (1-p_s)^a a \\ &> \sum_{k=1}^{a} \binom{a}{k} p_s^k (1-p_s)^{a-k} k = p_s a. \end{aligned} \tag{5.26}$$

Since $A_1 = m$, $\mathbb{E}[A_2|A_1 = m] > p_s m$ and by iteration, we obtain

$$\mathbb{E}[A_{S_m^{(1)}(p_s)}|A_1 = m] > \mathbb{E}[p_s^{S_m^{(1)}(p_s)-1} m]. \tag{5.27}$$

Because $f(x) = p_s^x$ is a convex function with respect to x, by the Jensen's inequality, we have

$$p_s^{s_m^{(1)}(p_s)-1} m < \mathbb{E}[p_s^{S_m^{(1)}(p_s)-1} m] < \mathbb{E}[A_{S_m^{(1)}(p_s)}|A_1 = m] = 1, \tag{5.28}$$

and therefore $s_m^{(1)}(p_s) > \log_{\frac{1}{p_s}}(m) + 1$.

Finally, from (5.25) and the above iteration, we have

$$\tau_m^{(1)} = \sum_i \mathbb{E}[A_i|A_1 = m] > \mathbb{E}\left[\sum_{i=0}^{S_m^{(1)}(p_s)-1} p_s^i m\right] = \frac{m - p_s}{1 - p_s}, \tag{5.29}$$

which completes the proof of Lemma 1. □

Corollary 1 *For a sub-optimal value $p_s = 0.5$ (fair splitting), the lower bounds of the average number of slots and the expected number of transmissions required in a selection epoch to select one user, given that there was a collision in the previous contention slot ($m > 1$), are*

$$s_m^{(1)}(0.5) > \log_2(m) + 1, \tag{5.30}$$

$$\tau_m^{(1)}(0.5) > 2m - 1. \tag{5.31}$$

Now, consider a selection epoch which starts with a collision of size m and the algorithm still needs to select q users. Note that if $q \le 0$, conditions (5.5) and (5.6) give $s_m^{(q)}(p_s) = \tau_m^{(q)}(p_s) = 0$. Also, from (5.8) and (5.10), $s_m^{(q)}(p_s) = s_m^{(m)}(p_s)$ and $\tau_m^{(q)}(p_s) = \tau_m^{(m)}(p_s)$ if $q \ge m$. When $0 < q < m$, the following lemma gives the lower bounds on the average duration of the epoch and the expected number of transmissions required.

Lemma 2 *For a selection epoch that needs to select q users out of a collision set of size m, $0 < q < m$, we have*

$$s_m^{(q)}(p_s) \ge s_m^{(q-1)}(p_s) + 1, \tag{5.32}$$

$$\tau_m^{(q)}(p_s) \ge \tau_m^{(q-1)}(p_s) + 1. \tag{5.33}$$

Proof: Both parts of Lemma 2 can be proved by induction. Formally, from Table 5.1, one can easily see that

$$s_j^{(i)}(p_s) \ge s_j^{(i-1)}(p_s) + 1 \text{ for } 0 < i < j < 3. \tag{5.34}$$

Given a pair (q, m) where $0 < q < m$, assume that $s_j^{(i)} > s_j^{(i-1)} + 1$ for all $0 < i < q$, $0 < j < m$ and $i < j$, we are proving that $s_m^{(q)} > s_m^{(q-1)} + 1$.

From recursive equation (5.16), we have

$$\begin{aligned} s_m^{(q)}(p_s) &= \frac{1 + \sum_{k=1}^{m-1} \binom{m}{k} p_s^k (1-p_s)^{m-k} \left(s_k^{(q)}(p_s) + s_{m-k}^{(q-k)}(p_s)\right)}{1 - p_s^m - (1-p_s)^m} \\ &\ge \frac{1 + \sum_{k=1}^{m-1} \binom{m}{k} p_s^k (1-p_s)^{m-k} \left(s_k^{(q-1)}(p_s) + s_{m-k}^{(q-k-1)}(p_s) + 1\right)}{1 - p_s^m - (1-p_s)^m} \\ &= s_m^{(q-1)}(p_s) + 1, \end{aligned} \tag{5.35}$$

which completes the proof for the first part of Lemma 2. The second part of the lemma can be proved in a similar manner. □

Corollary 2 *For $0 < q < m$,*

$$s_m^{(q)}(p_s) > \log_{\frac{1}{p_s}}(m) + q, \tag{5.36}$$

$$\tau_m^{(q)}(p_s) > \frac{m-1}{1-p_s} + q. \tag{5.37}$$

Furthermore, for $p_s = 0.5$,

$$s_m^{(q)}(0.5) > \log_2(m) + q, \tag{5.38}$$

$$\tau_m^{(q)}(0.5s) > 2m - 2 + q. \tag{5.39}$$

Now, consider the asymptotic case when the number of users n tends to infinity, by dropping the sub-index and let $l^{(q)}(p_c, p_s)$ and $\mu^{(q)}(p_c, p_s)$ denote the average length of the algorithm and the expected number of transmissions, respectively.

Lemma 3 *The average number of slots required by the algorithm to find the q desired users, and the expected number of transmissions in an asymptotic case, for $n \to \infty$, are given as*

$$l^{(q)}(p_c, p_s) = \sum_{m=0}^{\infty} e^{-p_c} \frac{p_c^m}{m!} s_m^{(q)}(p_s) + \sum_{m=0}^{q-1} e^{-p_c} \frac{p_c^m}{m!} l^{(q-m)}(p_c, p_s), \tag{5.40}$$

$$\mu^{(q)}(p_c, p_s) = \sum_{m=0}^{\infty} e^{-p_c} \frac{p_c^m}{m!} \tau_m^{(q)}(p_s) + \sum_{m=0}^{q-1} e^{-p_c} \frac{p_c^m}{m!} \mu^{(q-m)}(p_c, p_s). \tag{5.41}$$

Proof: For given q, if $n \to \infty$, we also have $(n - q) \to \infty$. In addition, the binomial random variable X, which denotes the number of users transmitting in a contention slot, is approximately Poisson distributed in this case. With the initial condition (5.1) and the Poisson approximation, Eq. (5.40) follows directly from Eq. (5.20) by replacing $\binom{n}{m} \left(\frac{p_c}{n}\right)^m \left(1 - \frac{p_c}{n}\right)^{n-m}$ with $e^{-p_c} \frac{p_c^m}{m!}$ and letting $l^{(q-m)}(p_c, p_s) = 0$ where appropriate. The second part of the lemma can also be proved similarly. □

Denote by $\tilde{l}^{(q)}(p_c, p_s)$ and $\tilde{\mu}^{(q)}(p_c, p_s)$ the lower bounds of $l^{(q)}(p_c, p_s)$ and $\mu^{(q)}(p_c, p_s)$, respectively. Using Corollary 2 and Lemma 3, we have the following theorem

Theorem 1 *In asymptotic case when the number of users approaches infinity, the lower bounds of the average number of slots and the expected number of transmissions required for the splitting based MSA to select q users are given by*

$$\tilde{l}^{(q)}(p_c, p_s) = \frac{e^{-p_c}}{1-e^{-p_c}} \left(1 + \sum_{m=1}^{q-1} \frac{p_c^m}{m!} \left(\tilde{l}^{(q-m)}(p_c, p_s) + \log_{\frac{1}{p_s}}(m) + m\right)\right) + \frac{e^{-p_c}}{1-e^{-p_c}} \sum_{m=q}^{\infty} \frac{p_c^m}{m!} \left(\log_{\frac{1}{p_s}}(m) + q\right), \tag{5.42}$$

$$\tilde{\mu}^{(q)}(p_c, p_s) = \frac{e^{-p_c}}{1-e^{-p_c}} \sum_{m=1}^{q-1} \frac{p_c^m}{m!} \left(\tilde{\mu}^{(q-m)}(p_c, p_s) + \frac{m-1}{1-p_s} + m\right) + \frac{e^{-p_c}}{1-e^{-p_c}} \sum_{m=q}^{\infty} \frac{p_c^m}{m!} \left(\frac{m-1}{1-p_s} + q\right). \tag{5.43}$$

Proof: For a single user selection algorithm ($q = 1$), Lemma 3 and Corollary 2 implies that

$$l^{(1)}(p_c, p_s) > 1 + \frac{e^{-p_c}}{1-e^{-p_c}} \sum_{m=1}^{\infty} a_m, \tag{5.44}$$

where $a_m = \frac{p_c^m}{m!} \log_{\frac{1}{p_s}}(m)$. The expression on the right hand side of (5.44) is the lower bound of the average duration of the single user selection algorithm, denoted by $\tilde{l}^{(1)}(p_c, p_s)$, if and only if the series a_m converges. In order to check if it this the case, we consider,

$$\lim_{m\to\infty} \frac{a_{m+1}}{a_m} = \lim_{m\to\infty} \frac{p_c}{m+1} \log_m(m+1) = 0. \tag{5.45}$$

Thus, using the D' Alembert criterion (the ratio test) [72] the series converges and $\tilde{l}^{(1)}(p_c, p_s)$ exists. By iteration, we have the lower bound of the average duration of the MSA as given in (5.42). The proof for the lower bound of the expected number of transmissions can be done similarly. □

5.6 Conclusion

In this chapter, we proposed a generalized splitting algorithm for the multiuser selection. The algorithm is characterized by two design parameters, namely, the contention factor p_c and the selection factor p_s. In particular, the two measures of interest are the duration of the algorithm (in time slots) and the number of transmissions required by all users in the system. Based on the principle of the algorithm, we obtained the exact expressions for the calculation of these measurements as well as their expectations. The expressions are in the form of recursive equations which are complex and do not reveal the scalable nature of the algorithm. For this reason, we derived lower bounds of these measures. As the algorithm was studied in a probabilistic view, these lower bounds are independent of the application scenarios and can be used as good approximations of behavior of large systems.

Chapter 6

Functional Equation Arising and Asymptotic Analysis of a Splitting Based MSA

6.1 Introduction

In the previous chapter, we described a novel splitting algorithm for the multiuser selection based on the single user version described in [70]. Our proposal is considered to be general while retaining the simplicity and the distributed nature of the selection problem. From the principle of the algorithm, recursive equations for the duration of the algorithm and for the number of transmissions required were formulated. Furthermore, we derived the lower bounds of the expectations of these measures. This chapter is devoted to a comprehensive analysis of the algorithm in the asymptotic case. The ultimate aim is to show the scalable nature of our proposal as well as to obtain a useful method for choosing the parameters.

The rest of this chapter is organized as follows. Functional equations arising in the MSA are showed in Section 6.2. Section 6.3 describes the contraction conditions that are necessary for establishing the solution of the functional equation in Section 6.4. Based on the general solution, we derive exact expressions for the expectations of the two measures of interest for the asymptotic case. Furthermore, using these expressions as closed form approximations for large network, the design parameters can be optimized. The simulation and numerical results are given in Section 6.5. Finally, Section 6.6 concludes the chapter.

6.2 Functional Equations for Asymptotic Analysis of the MSA

Recall Lemma 3 in Chapter 5. The first summation in (5.40) is, in fact, the unconditional expectation of the duration of a selection epoch when the algorithm (still) needs to select q users. Re-arrange (5.40) and let

$$u^{(q)}(p_c, p_s) \stackrel{\text{def}}{=} \sum_{m=0}^{\infty} e^{-p_c} \frac{p_c^m}{m!} s_m^{(q)}(p_s) \tag{6.1}$$

$$= (1 - e^{-p_c}) l^{(q)}(p_c, p_s) - \sum_{m=1}^{q-1} e^{-p_c} \frac{p_c^m}{m!} l^{(q-m)}(p_c, p_s). \tag{6.2}$$

Similarly, let

$$v^{(q)}(p_c, p_s) \stackrel{\text{def}}{=} \sum_{m=0}^{\infty} e^{-p_c} \frac{p_c^m}{m!} \tau_m^{(q)}(p_s) \tag{6.3}$$

be the unconditional expectation of the number of transmission needed for a selection epoch when the algorithm needs to select q users. From Eq. (5.41), we have

$$v^{(q)}(p_c, p_s) = (1 - e^{-p_c}) \mu^{(q)}(p_c, p_s) - \sum_{m=1}^{q-1} e^{-p_c} \frac{p_c^m}{m!} \mu^{(q-m)}(p_c, p_s). \tag{6.4}$$

Eq. (6.2) and Eq. (6.4) are recursive formulas (in term of q) for $l^{(q)}(p_c, p_s)$ and $\mu^{(q)}(p_c, p_s)$, respectively. Using the initial conditions and the equations derived in Chapter 5, we have the next lemma

Lemma 4 *In asymptotic case, the unconditional expectation of the duration of a selection epoch when algorithm (still) needs to select q users ($q \geq 1$), and the* *unconditional expectation of the number of transmissions required are given by the following functional equations*

$$\begin{aligned} u^{(q)}(p_c,p_s) &= u^{(q)}(p_sp_c,p_s) + e^{-p_sp_c}u^{(q)}\big((1-p_s)p_c,p_s\big) \\ &+ \sum_{m=1}^{q-1} e^{-p_sp_c}\frac{(p_sp_c)^m}{m!}u^{(q-m)}\big((1-p_s)p_c,p_s\big) \\ &+ 1 - e^{-p_c} - p_ce^{-p_c} - e^{-p_sp_c} - I_{(q>1)}p_sp_ce^{-p_c}, \end{aligned} \tag{6.5}$$

$$\begin{aligned} v^{(q)}(p_c,p_s) &= v^{(q)}(p_sp_c,p_s) + e^{-p_sp_c}v^{(q)}\big((1-p_s)p_c,p_s\big) \\ &+ \sum_{m=1}^{q-1} e^{-p_sp_c}\frac{(p_sp_c)^m}{m!}v^{(q-m)}\big((1-p_s)p_c,p_s\big) \\ &+ p_c - p_sp_ce^{-p_c} - (1-p_s)p_ce^{-p_sp_c}. \end{aligned} \tag{6.6}$$

where $I_{(q>1)}$ is the indicator function, which is defined as follow:

$$I_{(q>1)} = \begin{cases} 1 & \text{if } q > 1 \\ 0 & \text{otherwise.} \end{cases} \tag{6.7}$$

Proof: By substituting (5.5), (5.10) and (5.16) into (6.1), we have (6.8) which contains two terms U_1 and U_2. These terms represent two sub-epochs generated by splitting the active set after a collision in the contention slot. In general, the first subset has (on average) p_sp_c users. The derivation for U_1 is given in (6.9), where the last line follows the definition in (6.1). This can be interpreted as the length of the sub-epoch, which is generated by evaluating the first subset, and which is statistically indistinguishable from that of the selection epoch with the same number of users. Here, $e^{-p_sp_c}$ and $p_sp_ce^{-p_sp_c}$ account for the events that the first subset is either empty or has exactly one user, respectively. The derivation for U_2 is slightly different and is not as straightforward. How the second subset be evaluated depends on how many users the algorithm still needs to select after successfully evaluating the first sub-epoch. Moreover, we need to consider two different cases. For the single user selection ($q = 1$), U_2 is derived in (6.10). However, for the multiuser selection ($q \geq 2$), if the first subset has exactly one

$$u^{(q)}(p_c,p_s) = e^{-p_c} + p_c e^{-p_c} + \sum_{m=2}^{\infty} e^{-p_c}\frac{p_c^m}{m!}$$
$$\left(1+\sum_{k=1}^{m}\binom{m}{k}p_s^k(1-p_s)^{m-k}s_k^{(q)}(p_s)+\sum_{k=0}^{m}\binom{m}{k}p_s^k(1-p_s)^{m-k}s_{m-k}^{(q-k)}(p_s)\right)$$
$$= 1+\underbrace{\sum_{m=2}^{\infty}e^{-p_c}\frac{p_c^m}{m!}\sum_{k=1}^{m}\binom{m}{k}p_s^k(1-p_s)^{m-k}s_k^{(q)}(p_s)}_{U_1}$$
$$+\underbrace{\sum_{m=2}^{\infty}e^{-p_c}\frac{p_c^m}{m!}\sum_{k=0}^{m}\binom{m}{k}p_s^k(1-p_s)^{m-k}s_{m-k}^{(q-k)}(p_s)}_{U_2}. \qquad (6.8)$$

$$U_1 = \sum_{m=1}^{\infty}\sum_{k=1}^{m}e^{-p_c}\frac{(p_c)^m}{m!}\frac{m!}{k!(m-k)!}p_s^k(1-p_s)^{m-k}s_k^{(q)}(p_s) - p_s p_c e^{-p_c}$$
$$= \sum_{k=1}^{\infty}e^{-p_s p_c}\frac{(p_s p_c)^k}{k!}s_k^{(q)}(p_s)\underbrace{\sum_{m=k}^{\infty}e^{-(1-p_s)p_c}\frac{\big((1-p_s)p_c\big)^{(m-k)}}{(m-k)!}}_{=1} - p_s p_c e^{-p_c}$$
$$= u^{(q)}(p_s p_c, p_s) - e^{-p_s p_c} - p_s p_c e^{-p_c}. \qquad (6.9)$$

if $q=1$

$$U_2 = e^{-p_s p_c}\sum_{m=0}^{\infty}e^{-(1-p_s)p_c}\frac{\big((1-p_s)p_c\big)^m}{m!}s_m^{(q)}(p_s) - e^{-p_c} - (1-p_s)p_c e^{-p_c}$$
$$= e^{-p_s p_c}u^{(q)}\big((1-p_s)p_c, p_s\big) - e^{-p_c} - (1-p_s)p_c e^{-p_c}. \qquad (6.10)$$

else

$$U_2 = \sum_{k=0}^{\infty}e^{-p_s p_c}\frac{(p_s p_c)^k}{k!}\sum_{m=k}^{\infty}e^{-(1-p_s)p_c}\frac{\big((1-p_s)p_c\big)^{m-k}}{(m-k)!}s_{m-k}^{(q-k)}(p_s)$$
$$- e^{-p_c} - p_s p_c e^{-p_c} - (1-p_s)p_c e^{-p_c}$$
$$= \sum_{k=0}^{q-1}e^{-p_s p_c}\frac{(p_s p_c)^k}{k!}u^{(q-k)}\big((1-p_s)p_c, p_s\big)$$
$$- e^{-p_c} - p_s p_c e^{-p_c} - (1-p_s)p_c e^{-p_c}. \qquad (6.11)$$

user (with probability $p_s p_c e^{-p_s p_c}$), the second subset can not be empty (with probability $e^{-(1-p_s)p_c}$), hence the extra term $p_s p_c e^{-p_c}$ $(= p_s p_c e^{-p_s p_c} e^{-(1-p_s)p_c})$ is included. The detailed derivation for U_2 in this case is given in (6.11).

The first part of Lemma 4 follows by putting U_1 (from (6.9)) and U_2 (from (6.10) or (6.11)) back into (6.8). The second part of the lemma can be proved in a similar manner. □

Lemma 4 shows two functional equations that need to be tackled. Unfortunately, it is not possible to obtain the solutions analytically for the current form of these equations. In the next section, we will obtain similar functional equations with contraction conditions which are necessary for establishing the closed form formulas for the measures.

6.3 Contraction Conditions

Lemma 5 *Let*

$$\alpha^{(q)}(p_c, p_s) \stackrel{\text{def}}{=} \sum_{m=0}^{\infty} \frac{e^{-p_c} p_c^m}{1 - e^{-p_c}} \frac{(s_m^{(q)}(p_s) - 1)}{m!}, \tag{6.12}$$

$$\beta^{(q)}(p_c, p_s) \stackrel{\text{def}}{=} \sum_{m=0}^{\infty} \frac{e^{-p_c} p_c^m}{1 - e^{-p_c}} \frac{(\tau_m^{(q)}(p_s) - m)}{m!}, \tag{6.13}$$

then

$$\lim_{p_c \to 0} \alpha^{(q)}(p_c, p_s) = 0, \tag{6.14}$$

$$\lim_{p_c \to 0} \beta^{(q)}(p_c, p_s) = 0. \tag{6.15}$$

Proof: From Eq. (6.12), we have

$$\lim_{p_c \to 0} \alpha^{(q)}(p_c, p_s) = \sum_{m=2}^{\infty} \frac{(s_m^{(q)}(p_s) - 1)}{m!} \lim_{p_c \to 0} \frac{b_m(p_c)}{f(p_c)}, \tag{6.16}$$

where $b_m(p_c) = e^{-p_c}p_c^m$ and $f(p_c) = 1 - e^{-p_c}$. Note that, for $m \geq 2$

$$\lim_{p_c \to 0} b_m(p_c) = \lim_{p_c \to 0} f(p_c) = 0. \tag{6.17}$$

Using the L'Hospital's rule [72], it can be shown that

$$\begin{aligned}\lim_{p_c \to 0} \frac{b_m(p_c)}{f(p_c)} &= \lim_{p_c \to 0} \frac{b'_m(p_c)}{f'(p_c)} \\ &= \lim_{p_c \to 0} \frac{me^{-p_c}p_c^{m-1} - e^{-p_c}p_c^m}{e^{-p_c}} \\ &= 0 \quad \forall m \geq 2, \end{aligned} \tag{6.18}$$

and the first part of Lemma 5 follows. The second part of the lemma can be proved similarly. □

Lemma 5 above shows the limiting condition of $\alpha^{(q)}(p_c, p_s)$ and $\beta^{(q)}(p_c, p_s)$, respectively. Naturally, $\alpha^{(q)}(p_c, p_s)$ is the unconditional expectation of the length of the selection epoch excluding the first contention slot scaled by $1 - e^{-p_c}$, the probability that the selection set is non empty. From (6.1) and (6.12), one can obtain the relationship between $u^{(q)}(p_c, p_s)$ and $\alpha^{(q)}(p_c, p_s)$ as follows:

$$u^{(q)}(p_c, p_s) = 1 + (1 - e^{-p_c})\alpha^{(q)}(p_c, p_s). \tag{6.19}$$

Consequently, (6.2) can be rewritten as

$$\begin{aligned} l^{(q)}(p_c, p_s) = \alpha^{(q)}(p_c, p_s) &+ \frac{1}{1 - e^{-p_c}} \\ &+ \sum_{m=1}^{q-1} \frac{e^{-p_c}}{1 - e^{-p_c}} \frac{p_c^m}{m!} l^{(q-m)}(p_c, p_s), \end{aligned} \tag{6.20}$$

Similarly, $\beta^{(q)}(p_c, p_s)$ denotes the unconditional expectation of the number of transmissions required in the selection epoch, excluding the first m transmissions in the contention slot, and scaled by $\Pr\{m > 0\}$. Analogous to (6.19) and (6.20), we have

$$v^{(q)}(p_c,p_s) = p_c + (1-e^{-p_c})\beta^{(q)}(p_c,p_s), \tag{6.21}$$

$$\mu^{(q)}(p_c,p_s) = \beta^{(q)}(p_c,p_s) + \frac{p_c}{1-e^{-p_c}} + \sum_{m=1}^{q-1} \frac{e^{-p_c}}{1-e^{-p_c}} \frac{p_c^m}{m!} \mu^{(q-m)}(p_c,p_s). \tag{6.22}$$

Eq. (6.20) and (6.22) imply that in order to calculate $l^{(q)}(p_c,p_s)$ and $\mu^{(q)}(p_c,p_s)$, we can try to find analytic forms of $\alpha^{(q)}(p_c,p_s)$ and $\beta^{(q)}(p_c,p_s)$, instead of $u^{(q)}(p_c,p_s)$ and $v^{(q)}(p_c,p_s)$. Furthermore, putting $u^{(q)}(p_c,p_s)$ and $v^{(q)}(p_c,p_s)$ from (6.19) and (6.21) into (6.5) and (6.6), after some manipulations, we have

$$\alpha^{(q)}(p_c,p_s) = t_1(p_c,p_s)\alpha^{(q)}(p_sp_c,p_s) + t_2(p_c,p_s)\alpha^{(q)}\big((1-p_s)p_c,p_s\big) + t_\alpha^{(q)}(p_c,p_s), \tag{6.23}$$

$$\beta^{(q)}(p_c,p_s) = t_1(p_c,p_s)\beta^{(q)}(p_sp_c,p_s) + t_2(p_c,p_s)\beta^{(q)}\big((1-p_s)p_c,p_s\big) + t_\beta^{(q)}(p_c,p_s), \tag{6.24}$$

where

$$t_1(p_c,p_s) = \frac{1-e^{-p_sp_c}}{1-e^{-p_c}}; \tag{6.25}$$

$$t_2(p_c,p_s) = \frac{e^{-p_sp_c}(1-e^{-(1-p_s)p_c})}{1-e^{-p_c}}; \tag{6.26}$$

$$t_\alpha^{(q)}(p_c,p_s) = 1 - \frac{p_ce^{-p_c} + I_{(q>1)}p_sp_ce^{-p_c}}{1-e^{-p_c}} + \sum_{m=1}^{q-1} \frac{e^{-p_sp_c}(p_sp_c)^m}{(1-e^{-p_c})m!} + t_2(p_c,p_s)\sum_{m=1}^{q-1}\frac{(p_sp_c)^m}{m!}\alpha^{(q-m)}\big((1-p_s)p_c,p_s\big); \tag{6.27}$$

$$t_\beta^{(q)}(p_c,p_s) = p_sp_c + \sum_{m=1}^{q-1} \frac{(1-p_s)p_ce^{-p_sp_c}(p_sp_c)^m}{(1-e^{-p_c})m!} + t_2(p_c,p_s)\sum_{m=1}^{q-1}\frac{(p_sp_c)^m}{m!}\beta^{(q-m)}\big((1-p_s)p_c,p_s\big). \tag{6.28}$$

Importantly, we can show that both $t_1(.)$ and $t_2(.)$ are positive and

$$t_1(p_c, p_s) + t_2(p_c, p_s) = 1, \tag{6.29}$$

for $p_c > 0$ and $0 < p_s < 1$. Thus, they are bounded in $(0, 1)$.

The following lemma provides another important property of $t_\alpha^{(q)}(p_c, p_s)$ and $t_\beta^{(q)}(p_c, p_s)$.

Lemma 6 *The quantities $t_\alpha^{(q)}(p_c, p_s)$ and $t_\beta^{(q)}(p_c, p_s)$ given in (6.27) and (6.28), respectively, are both positive and*

$$\lim_{p_c \to 0} t_\alpha^{(q)}(p_c, p_s) = \lim_{p_c \to 0} t_\beta^{(q)}(p_c, p_s) = 0. \tag{6.30}$$

Consequently, both $t_\alpha^{(q)}(p_c, p_s)$ and $t_\beta^{(q)}(p_c, p_s)$ locally increase with p_c in the vicinity of 0.

Proof: Consider

$$t_\alpha^{(1)}(p_c, p_s) = f_1(p_c) = 1 - \frac{p_c e^{-p_c}}{1 - e^{-p_c}} \tag{6.31}$$

and, for $q > 1$, also

$$t_\alpha^{(q)}(p_c, p_s) = f_1(p_c) + f_2(p_c, p_s) + f_3(p_c, p_s) + f_3(p_c, p_s), \tag{6.32}$$

where

$$f_2(p_c, p_s) = \frac{p_s p_c (e^{-p_s p_c} - e^{-p_c})}{1 - e^{-p_c}}; \tag{6.33}$$

$$f_3(p_c, p_s) = \sum_{m=2}^{q-1} \frac{e^{-p_s p_c}(p_s p_c)^m}{(1 - e^{-p_c}) m!}; \tag{6.34}$$

$$f_4(p_c, p_s) = t_2(p_c, p_s) \sum_{m=1}^{q-1} \frac{(p_s p_c)^m}{m!} \alpha^{(q-m)}\big((1 - p_s) p_c, p_s\big). \tag{6.35}$$

It can be seen that each $f_i(p_c, p_s)$, $i \in \{1, 2, 3, 4\}$ is positive. We now prove that $\lim_{p_c \to 0} f_i(p_c, p_s) = 0$. First

$$\begin{aligned} \lim_{p_c \to 0} f_1(p_c) &= 1 - \lim_{p_c \to 0} \frac{p_c e^{-p_c}}{1 - e^{-p_c}} \\ &= 1 - \lim_{p_c \to 0} \frac{b_1(p_c)}{f(p_c)}, \end{aligned} \tag{6.36}$$

where $f(p_c) = 1 - e^{-p_c}$. One can see that

$$\lim_{p_c \to 0} b_1(p_c) = \lim_{p_c \to 0} f(p_c) = 0, \tag{6.37}$$

Using the L'Hospital's rule [72], we have

$$\begin{aligned} \lim_{p_c \to 0} f_1(p_c) &= 1 - \lim_{p_c \to 0} \frac{b_1'(p_c)}{f'(p_c)} \\ &= 1 - \lim_{p_c \to 0} \frac{e^{-p_c} - p_c e^{-p_c}}{e^{-p_c}} = 0 \end{aligned} \tag{6.38}$$

Using the same method, we can prove that

$$\lim_{p_c \to 0} f_2(p_c, p_s) = \lim_{p_c \to 0} f_3(p_c, p_s) = 0. \tag{6.39}$$

and $\lim_{p_c \to 0} f_4(p_c, p_s) = 0$ follows from Lemma 5.

The fact that $t_\beta^{(q)}(p_c, p_s)$ is positive and locally increases in the vicinity of 0, can be proved in a similar manner. □

Remark 1: In (6.23) and (6.24), $\alpha^{(q)}(p_c, p_s)$ and $\beta^{(q)}(p_c, p_s)$ are expressed as functional equations similar to those given in Lemma 4. The main difference are the conditions given in (6.29) and Lemma 6, which we refer to as the *contraction conditions.*

6.4 Solution of Functional Equation

Subject to the contraction conditions, the functional equation can be solved. Let

$$\sigma_1(p_c) \stackrel{\text{def}}{=} p_s p_c, \tag{6.40}$$

$$\sigma_2(p_c) \stackrel{\text{def}}{=} (1 - p_s) p_c, \tag{6.41}$$

and let H be the semi-group of linear substitutions generated by functions σ_1 and σ_2, where the semi-group operation is the composition of functions. The identity of H, denoted by ϵ, is the function $\epsilon(p_c) = p_c$. Any $\sigma \in H$ can be written in the form

$$\sigma = \sigma_{i_1}\sigma_{i_2}\dots\sigma_{i_k}, \tag{6.42}$$

for $k \geq 0$ and $i_j \in \{1,2\}$.

In addition, define

$$|\sigma|_1 \overset{\text{def}}{=} \text{card}\{j|i_j = 1\}; \tag{6.43}$$

$$|\sigma|_2 \overset{\text{def}}{=} \text{card}\{j|i_j = 2\}; \tag{6.44}$$

$$\text{and } |\sigma| \overset{\text{def}}{=} |\sigma|_1 + |\sigma|_2; \tag{6.45}$$

where $|\sigma|_i$ is the number of occurrences of σ_i in σ and $|\sigma|$ is called the *length* of substitution. Let $H^k \subset H$ be the subset containing all substitutions of length k. Any element $\sigma^k(p_c) \in H^k$, which can be written in the form of (6.42), has a value equal to $p_s^{|\sigma|_1}(1-p_s)^{|\sigma|_2}p_c$. Thus, for any $p_c > 0$ and $p_s \in (0,1)$, we have

$$\sigma^k\sigma_j(p_c) < \sigma^k(p_c), \;\forall k \geq 0, j \in \{1,2\}, \tag{6.46}$$

$$\lim_{k\to\infty} \sigma^k(p_c) = \sigma^\infty(p_c) = 0. \tag{6.47}$$

Note that the semigroup H satisfies the obvious decompositions:

$$H = \begin{cases} \{\epsilon\} \cup \sigma_1 H \cup \sigma_2 H \\ \{\epsilon\} \cup H\sigma_1 \cup H\sigma_2, \end{cases} \tag{6.48}$$

and correspondingly for a given k, $k \geq 1$

$$H_k = \begin{cases} \sigma_1 H_{k-1} \cup \sigma_2 H_{k-1} \\ H_{k-1}\sigma_1 \cup H_{k-1}\sigma_2. \end{cases} \tag{6.49}$$

The following theorem states the solution to functional equations of the form (6.23) and (6.24).

Theorem 2 *If $t(p_c)$, $t_1(p_c)$, $t_2(p_c)$ are positive entire functions satisfying the following contraction conditions*

C1) *$t(p_c)$ is locally increasing function in the vicinity of 0 and*

C2) $t_1 + t_2 = 1$,

then functional equations of the form

$$\lambda(p_c) = t_1(p_c)\lambda\big(\sigma_1(p_c)\big) + t_2(p_c)\lambda\big(\sigma_2(p_c)\big) + t(p_c), \tag{6.50}$$

where

$$\lim_{p_c \to 0} \lambda(p_c) = 0, \tag{6.51}$$

have a unique entire solution given by

$$\lambda(p_c) = \sum_{\sigma \in H} r\big(\sigma(p_c)\big)t\big(\sigma(p_c)\big), \tag{6.52}$$

where for given $\sigma^k = \sigma_{i_1} \dots \sigma_{i_k} \in H^k, k \geq 0$,

$$r\big(\sigma^k(p_c)\big) \triangleq \prod_{j=0}^{k-1} t_{i_{j+1}}\big(\sigma^{k,j}(p_c)\big). \tag{6.53}$$

Here, $\sigma^{k,j}(p_c)$ is an element of H^j formed by taking the first j substitutions of σ^k.

Proof: By iteration, $\lambda(p_c)$ in (6.50) can be written as

$$\lambda(p_c) = \lambda^*(p_c) + \sum_{k=0}^{\infty} c_k, \tag{6.54}$$

where

$$c_k = \sum_{\sigma^k \in H^k} r\big(\sigma^k(p_c)\big)t\big(\sigma^k(p_c)\big), \tag{6.55}$$

$$\lambda^*(p_c) = \lim_{k \to \infty} \sum_{\sigma^k \in H^k} r\big(\sigma^k(p_c)\big)\lambda\big(\sigma^k(p_c)\big). \tag{6.56}$$

Eq. (6.54) holds if

1. The series c_k converges and,

2. $\lambda^{*(q)}(p_c, p_s)$ exists.

First, using decomposition in (6.49), we have

$$c_{k+1} = \sum_{\sigma^{k+1} \in H^k \sigma_1 \cup H^k \sigma_2} r\big(\sigma^{k+1}(p_c)\big) t\big(\sigma^{k+1}(p_c)\big)$$
$$= \sum_{\sigma^k \in H^k} r\big(\sigma^k(p_c)\big) \sum_{j=1}^{2} t_j\big(\sigma^k(p_c)\big) t\big(\sigma^k \sigma_j(p_c)\big). \tag{6.57}$$

Therefore,

$$\lim_{k\to\infty} c_{k+1} < \sum_{\sigma^k \in H^k} r\big(\sigma^k(p_c)\big) t\big(\sigma^k(p_c)\big) \sum_{j=1}^{2} t_j\big(\sigma^k(p_c)\big) = c_k, \tag{6.58}$$

since $\sigma^k \sigma_j(p_c) < \sigma^k(p_c)$ (Eq. (6.46)) and as k tends to infinity, $\sigma^k(p_c)$ approaches 0 (Eq. (6.47)), while $t(.)$ is a positive and locally increasing function in the vicinity of 0 (condition **C1**). Thus

$$\lim_{k\to\infty} \frac{c_{k+1}}{c_k} < 1, \tag{6.59}$$

by the D'Alembert's criterion (the ratio test) [72], the series converges.

On the other hand, because

$$\lim_{k\to\infty} \lambda\big(\sigma^k(p_c)\big) = 0 \text{ and}$$
$$\lim_{k\to\infty} r\big(\sigma^k(p_c)\big) = 0, \tag{6.60}$$

we can see that $\lambda^*(p_c)$ is an entire function that is bounded by 0. By Liouville's theorem for complex analysis [73], the entire function is constant; more precisely, $\lambda^*(p_c) = 0$. Therefore, we can conclude that $\lambda(p_c)$ given in (6.52) is the convergent sum, and it is a unique solution of the functional equation given in (6.50). □

Theorem 2 gives rise to the following corollary.

Corollary 3 *In the asymptotic case, the average duration of the splitting based MSA to select q users and the expected number of transmissions are given by*

$$l^{(q)}(p_c,p_s) = \frac{1}{1-e^{-p_c}} + \frac{e^{-p_c}}{1-e^{-p_c}}\sum_{m=1}^{q-1}\frac{p_c^m}{m!}l^{(q-m)}(p_c,p_s) + \sum_{k=0}^{\infty}\sum_{\sigma_k\in H_k} r\big(\sigma_k(p_c)\big)t_\alpha^{(q)}\big(\sigma_k(p_c),p_s\big) \tag{6.61}$$

$$\mu^{(q)}(p_c,p_s) = \frac{p_c}{1-e^{-p_c}} + \frac{e^{-p_c}}{1-e^{-p_c}}\sum_{m=1}^{q-1}\frac{p_c^m}{m!}\mu^{(q-m)}(p_c,p_s) + \sum_{k=0}^{\infty}\sum_{\sigma_k\in H_k} r\big(\sigma_k(p_c)\big)t_\beta^{(q)}\big(\sigma_k(p_c),p_s\big). \tag{6.62}$$

Remark 2: As $l^{(q)}(p_c,p_s)$ and $\mu^{(q)}(p_c,p_s)$ are given in the form of positive convergent series. Considering the first few terms of the summations in (6.61) and (6.62) (says up to $k=5$) gives good approximations of the average duration and of the expected number of transmissions required in the asymptotic case. Furthermore, given a large system, the truncated expressions can be used to calculate the measures of interest, which allow the system designers to quickly compute the necessary parameters for the system optimization, when required.

6.5 Numerical Results and Discussions

There is a close relationship between the selection algorithm proposed and the CRA random access. In particular, the selection factor is analogous to the splitting probability of the CRA while the contention factor represents the traffic intensity at the beginning of each collision resolution interval (CRI), i.e., the product of the input rate and the window size of the simplified window access (SWA) algorithm [45, 28]. In other words, it is always possible to construct a random access protocol using the generalized MSA and the throughput of such the protocol is equal to the ratio of the number of users to be selected q and the average duration $l^{(q)}(p_c, p_s)$ of the algorithm, i.e., the inverse of the time expected to find a user. Note that, most researches in the literature [8, 9, 70, 50, 69] only evaluate the average duration of the algorithm $l^{(q)}(p_c, p_s)$.

If we assume that a ternary feedback is negligibly small and there is no other cost involved with the selection process, the expected number of transmissions $\mu^{(q)}(p_c, p_s)$ can be considered to be the overhead of the algorithm. From (6.22), the selection factor p_s only affects $\mu^{(q)}(p_c, p_s)$ through $\beta^{(q)}(p_c, p_s)$, and also,

$$\lim_{p_s \to 0} \beta^{(q)}(p_c, p_s) = 0. \tag{6.63}$$

Thus, given the contention factor p_c, a small value of p_s close to 0 would result in a small overhead. Furthermore, it can be shown that

$$\lim_{p_c \to 0} \mu^{(1)}(p_c, p_s) = 1. \tag{6.64}$$

Though not formally proved in this chapter, we expect that $\mu^{(q)}(p_c, p_s)$ is a non-decreasing function of both parameters p_c and p_s. Therefore, if there is no delay constraint, the trivial solution to minimize the overhead is $p_c = p_s = \epsilon$, where ϵ is set to be arbitrarily small. In this case, an infinite delay is expected, which makes the situation non-realistic and undesirable. In practice, we encounter designs that take into account both measures jointly, so different optimization problems can be formulated. One simple possibility is to minimize a weighted sum of the overhead and the duration as follows:

$$\min_{p_c, p_s} \left[d_l l^{(q)}(p_c, p_s) + d_\mu \mu^{(q)}(p_c, p_s) \right]. \tag{6.65}$$

Here d_l and d_μ are the weight factors, showing how important a measure is comparing to the another. Often, these factors are application dependent. In the remaining of this section, we consider two illustrative cases: i.) $d_l = 1$ and $d_\mu = 0$, i.e., only the delay is concerned and ii.) $d_l = 2/3$ and $d_\mu = 1/3$, i.e., the delay is considered twice as important as the overhead associated with the selection process. Note that, all the infinite series are truncated up to $k = 5$ in the computations.

When $d_l = 1$ and $d_\mu = 0$, the problem in (6.65) collapses to minimizing the expected duration of the algorithm, i.e., to find

$$\{p_c^{(q)*}, p_s^{(q)*}\} = \arg\min_{p_c, p_s} l^{(q)}(p_c, p_s). \tag{6.66}$$

The solution to this problem can be found numerically using some optimization algorithms. For different number of users q to be selected, Table 6.1 shows the optimal parameters obtained using the steepest descent method. Note that, in order to guarantee the global optimality of the parameters obtained using the numerical optimization, $l^{(q)}(p_c, p_s)$ needs to be convex. However, in this work, we omit the mathematical proof and assume the convexity.

Figs. 6.1 and 6.2 plot $l^{(2)}(p_c, p_s)$ as a function of p_c and p_s for the cases of $p_s = p_s^{(2)*}$ and $p_c = p_c^{(2)*}$, respectively. They are supported by simulations for two cases $n = 20$ and $n = 100$. These results not only confirm the convexity assumption but also show that truncation at $k = 5$ can be used to approximate the performance.

For comparison purposes, Fig. 6.3 and Fig. 6.4 show the simulation results for the average duration of the algorithm to select one user and three users as the total number of available users varies for two cases: i.) the parameters are chosen optimally $p_c = p_c^{(q)*}$, $p_s = p_s^{(q)*}$; and ii.) the parameters are set to maximize the probability of success in the contentions slots ($p_c = 1$) and also the fair splitting ($p_s = 0.5$). It can be observed from these figures that the number of slots required increases with the number of users in the system, but gradually converges to the numerical asymptotic as n gets large. In addition, when the optimal parameters are used, the algorithm performs better even when the number of users n is as small as 10. The performance gain increases as the number of users to be selected q increases. Furthermore, we note that for large n, a simple single user selection algorithm with two parameters requires approximately 2.45 slots on average, which is comparable to that of the complex algorithm proposed in [50].

Table 6.1: Optimal parameters $\{p_c^{(q)*}, p_s^{(q)*}\}$

q	$d_l = 1,\ d_\mu = 0$	$d_l = 2/3,\ d_\mu = 1/3$
1	$\{1.1052, 0.4534\}$	$\{0.7978, 0.3676\}$
2	$\{1.2358, 0.4763\}$	$\{0.8634, 0.3793\}$
3	$\{1.2545, 0.4528\}$	$\{0.8860, 0.3466\}$
4	$\{1.2867, 0.4280\}$	$\{0.8929, 0.3398\}$
5	$\{1.2939, 0.4219\}$	$\{0.8985, 0.3360\}$

We now consider the overhead and the delay jointly. For demonstration, the weight factors are set to $d_l = 2/3$ and $d_\mu = 1/3$. The optimal parameters for different q are also recorded in Table 6.1. Fig. 6.5 and Fig 6.6 plot the weighted sum of the average duration of the algorithm and the expected number of transmissions as the parameters varies. Due to the effect of $\mu^{(q)}(p_c, p_s)$, the optimal values of the parameters are now shifted to the left which clearly show an implication that a lower overhead would be achieved for smaller values of the parameters p_c and p_s.

Fig. 6.7 and Fig 6.8 plot the cost function as the number of users to be selected q increases for two cases: i.) $d_l = 1$, $d_\mu = 0$; and ii.) $d_l = 2/3$, $d_\mu = 1/3$, respectively. In these figures, the parameters are chosen optimally (as listed in Table 6.1). We also plot the cost function obtained using the lower bounds of $l^{(q)}(p_c, p_s)$ and $\mu^{(q)}(p_c, p_s)$ given in Theorem 1. In both cases, the gap between the lower bounds and exact expressions are visible. However, as the cost function increases somewhat linearly with q, the curves obtained using (5.42) and (5.43) capture the trend and follow similarly.

Remark 3: The truncation of the infinite sums in Corollary 3 gives the tight lower bounds of $l^{(q)}(p_c, p_s)$ and $\mu^{(q)}(p_c, p_s)$ since both series converge and only contain positive terms. The lower bounds we derived in Chapter 5 are not as tight as those based on the truncation [1]. However, in terms of computation, the complexity of (6.61) and (6.62) could be rather high as q gets larger. Therefore, if one wants to consider the number of users to be selected as a design parameter, the bounds in Theorem 1 are practically useful since the behavior of the measures can be captured as q increases.

6.6 Conclusions

In this chapter, we studied the performance of the splitting based MSA in the asymptotic case. Mathematical analysis of the measures of interest leaded to functional equations of the same form. Subject to some contraction conditions, the exact expressions for the average duration of the algorithm and the expected number of transmissions required were derived. Furthermore, these results shed light to the problems in choosing the parameters for the algorithm. For minimizing the expected number of transmissions (the overhead), the two parameters shall be set arbitrarily small. On the other hand, the resulting delay, represented by the average duration, would be infinite in this case. For the design purposes, it is possible to assume both the delay and the overhead jointly. For example, one can consider optimizing the parameters in order to minimize the weighted sum of the measures of interest.

[1] To avoid confusion, we refer to truncations (6.61) and (6.62) as the exact expressions for $l^{(q)}(p_c, p_s)$ and $\mu^{(q)}(p_c, p_s)$.

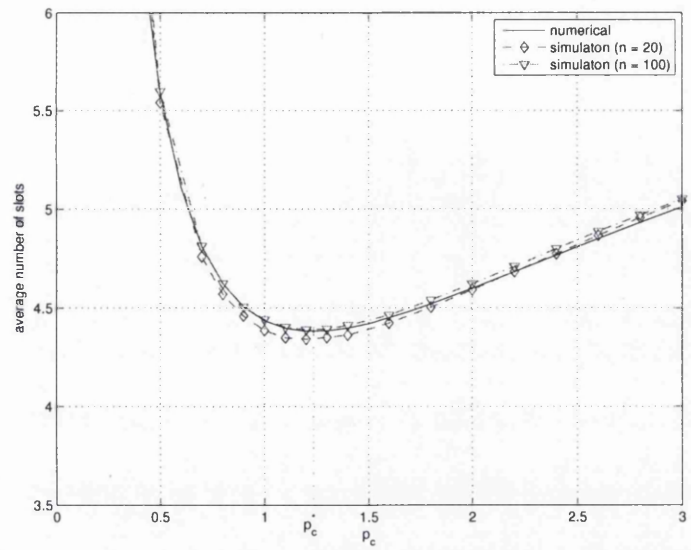

Figure 6.1: The average number of slots required to select two users as a function of p_c ($p_s = p_s^{(2)*}$)

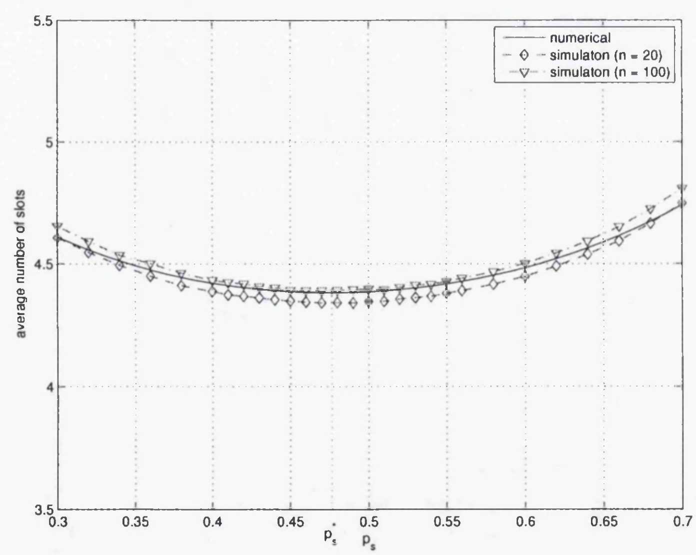

Figure 6.2: The average number of slots required to select two user as a function of p_s ($p_c = p_c^{(2)*}$)

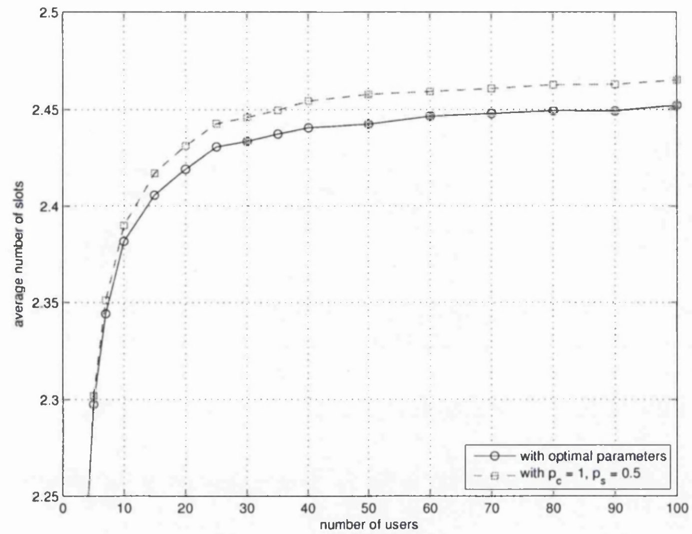

Figure 6.3: The average number of slots required to select one user for different network sizes

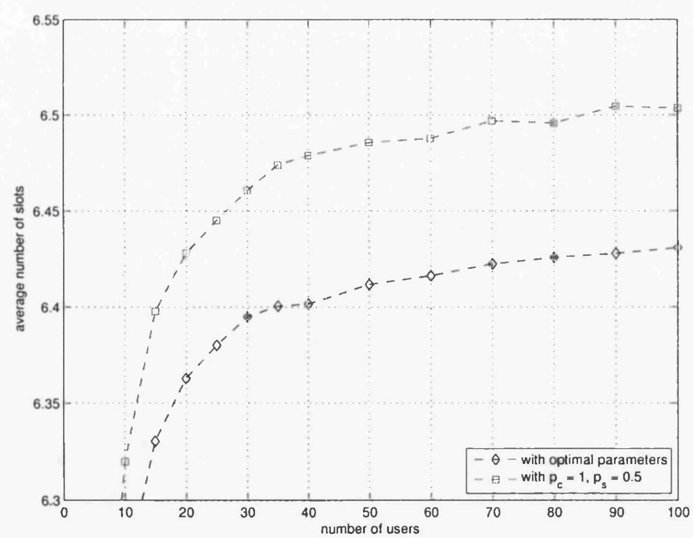

Figure 6.4: The average number of slots required to select three users for different network sizes

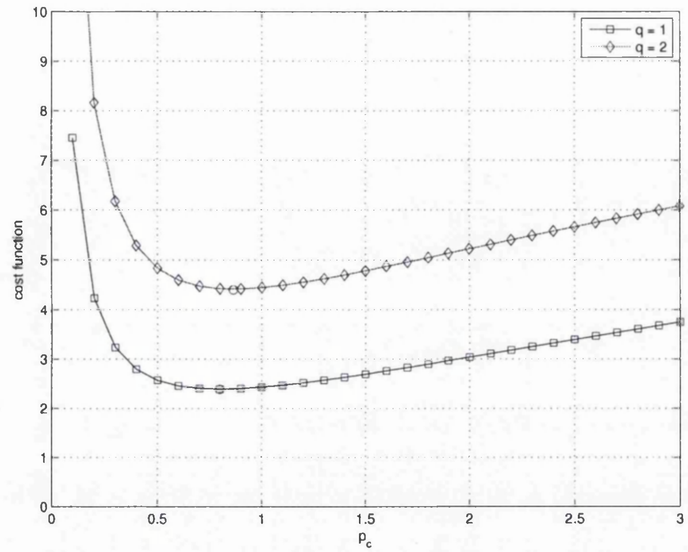

Figure 6.5: A cost function .vs the contention factor p_c ($d_l = 2/3$, $d_\mu = 1/3$, $p_s = p_s^{(q)*}$)

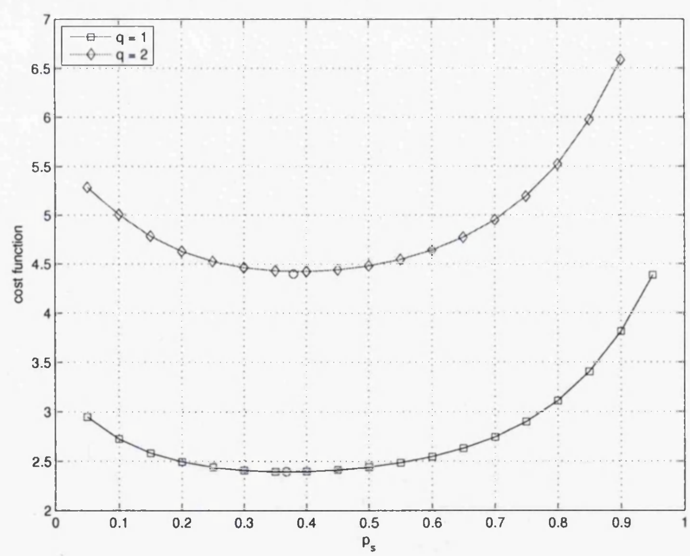

Figure 6.6: A cost function .vs the selection factor p_s ($d_l = 2/3$, $d_\mu = 1/3$, $p_c = p_c^{(q)*}$)

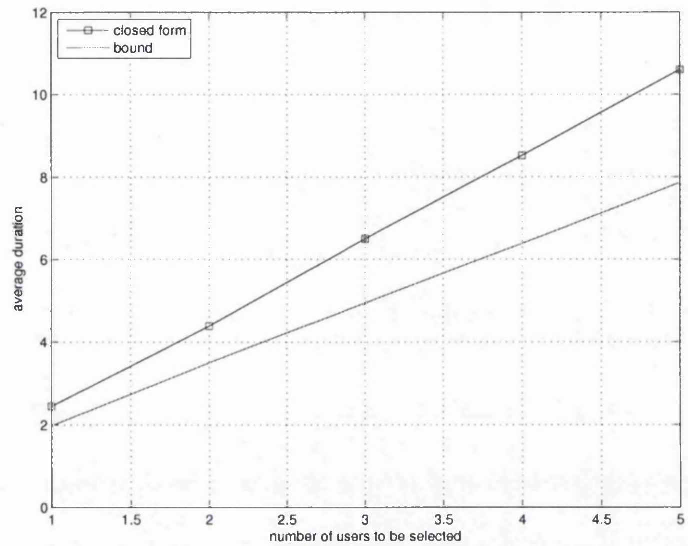

Figure 6.7: A cost function .vs the number of users to be selected ($d_l = 1$, $d_\mu = 0$)

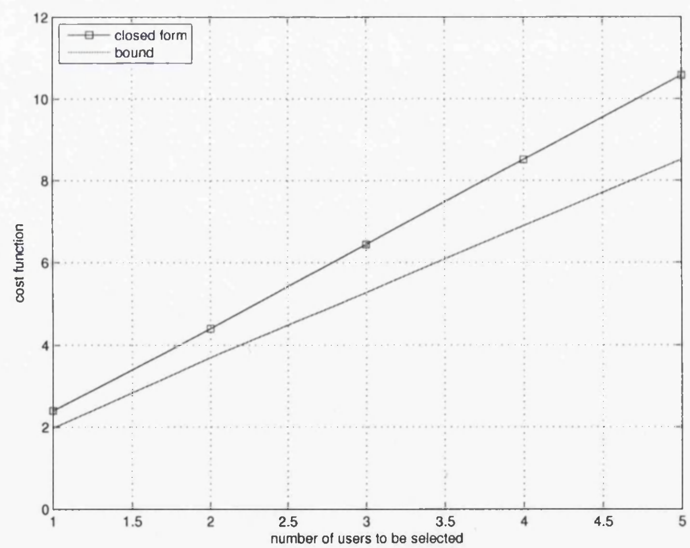

Figure 6.8: A cost function .vs the number of users to be selected ($d_l = 2/3$, $d_\mu = 1/3$)

As a final note, the lower bounds derived in the previous chapter are not tight. However, they require lower computational complexity compared with the expressions derived in this chapter. In addition, they capture the behavior of the measures of interest as q increases. Thus, these bounds are useful when the number of users to be selected is considered as a design parameter.

Chapter 7

An Energy Efficient Cooperative Transmission Scheme with Distributed Beamforming and Sensor Selection in Wireless Sensor Networks

7.1 Introduction

Distributed beamforming has been recently studied in WSN as a promising cooperative transmission technique for efficient data fusion of sensors observations [74, 75, 76, 77, 78]. By collaborating their transmission using *distributed phased arrays* the sensor nodes are able to emulate a traditional fixed array of antenna elements and achieve the same gains in terms of main lobe enhancement, side lobe reduction, and null pointing to improve the intended receiver's SNR. Furthermore, the interference caused by unwanted transmitters (if any) can be removed. The use of the term "distributed" has two distinct meaning in the sense of distributed beamforming [79]. First, in contrast to traditional beamforming literature which relies on a strict, uniform placement of the antenna elements, the antennas of the distributed beamformers, i.e., the sensor nodes themselves, are distributed in some randomly structured fashion. In this case, the location of each element must be considered on its own, rather than simply considering the location of the array as a whole. Note that, the sensors are still controlled by some central sources; hence some quantities such as the locations, phase offsets, and transmit capabilities of each sensor are taken into account when calculating the beamforming weights. Second, each sensors participating in cooperative transmission are independent processing units. Thus, methods for determining ideal complex weights must be distributed in the sense that they can be carried out by each node individually without sharing significant amount of information.

With the distributed beamforming approach, the sensor transmission power can be reduced as the number of active sensors increases. However, for coherent combining at the fusion center (FC), the carrier phases of the signals transmitted by sensors need to be aligned through a control signal, i.e., feedback. When a certain adaptive algorithm is used for the carrier phase alignment, there are two important issues for the distributed beamforming: i.) the amount of control signal; and ii.) the convergence rate. Thus, it is desirable to have a transmission scheme with the fast convergence rate and the low control overhead. In [80], it is shown that binary signaling can be used for carrier phase alignment.

On the other hand, if the time division duplex (TDD) mode is used for communication, the phase alignment can be achieved by having each sensor to estimate the channel coefficients of the wireless link to the FC based on the channel reciprocity. This approach requires the FC to broadcast a pilot signal so that each sensor can carry out the channel estimation. The sensors can then use the (estimated) channel knowledge to adjust the phase of transmitted signals for coherent combining at the FC. In this case, adaptive algorithms are not required. However, there is still another issue. Provided that the FC requires a target signal-to-noise ratio (SNR) to successfully decode the sensors messages, it is not necessary to use all the available sensors for cooperative transmission as the resulting SNR at the

FC might be higher than needed. To obtain energy savings, it would be desirable that the number of active sensors is minimized, while the target SNR is met [81].

In this chapter, we demonstrate application of the splitting based multiuser selection algorithm (MSA) for the sensor selection in a cooperative transmission scheme with the distributed beamforming which potentially achieves the energy efficiency. In brief, triggered by a request from the FC, the communication between the sensors and the FC is performed in four (4) phases:

- channel state information (CSI) acquisition phase;
- sensor selection phase;
- beamforming phase;
- cooperative transmission phase.

Throughout the chapter, it is assumed that the TDD is employed. The total energy consumption of all four phases is considered. Different to other transmission schemes in the literature, our proposal includes a new *sensor selection phase* that selects the best sensors for the cooperative transmission and to maximize the energy efficiency.

The remainder of this chapter is organized as follows. The system model is presented in Section 7.2. A cooperative transmission scheme consisting of four phases is proposed in Section 7.3. Section 7.4 provides analyses of the energy consumption. Simulation and numerical results are shown in Section 7.5. Finally, Section 7.6 concludes the chapter with some remarks.

7.2 System Model

We consider a WSN consisting of n sensors denoted by $\mathtt{S}_1, \mathtt{S}_2, \ldots, \mathtt{S}_n$, and a FC, denoted by $\mathtt{D}$, representing the destination. A graphical illustration of the network is depicted in Fig. 7.1. All the network nodes (sensors and the FC) are equipped with a single antenna. Periodically, the FC collects messages (representing sensor measurements over the time) which are commonly available at all sensors. It is assumed that all sensors use the same coding scheme of L-element codewords. Upon request, a common data packet of L symbols, denoted by $\mathbf{s}$, will be cooperatively sent to $\mathtt{D}$.

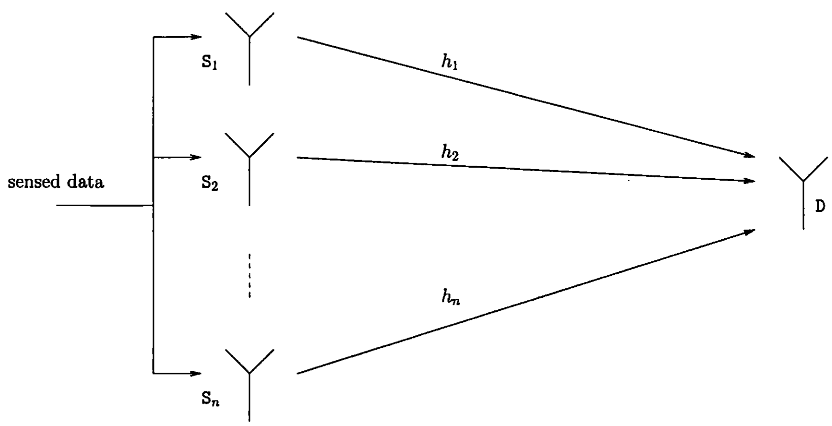

Figure 7.1: Sensor network model with fusion center

Let h_i, $i = 1, 2, \ldots, n$, be the channel gains of the links from $\mathtt{S}_i$ to $\mathtt{D}$. As the TDD mode is considered, this channel gain is identical to that of the reverse link from $\mathtt{D}$ to $\mathtt{S}_i$. Note that, in this chapter, we assume that all channels are flat fading and that the channel gains h_i are IID for different i. This assumption is reasonable if the sensors are uniformly deployed in approximately equal distance from the FC and provided that transmission to the FC experiences multipath fading, i.e., there are rich scatterers around the FC. Let $\alpha_i = |h_i|^2$ and denote by $f(\alpha)$ the common probability density function (PDF) of α_i's.

For transmission of messages to the FC, multiple sensors can cooperatively form a distributed beamformer provided that each of them knows the CSI of all the active sensors. Let $\mathcal{Q}$, $\mathcal{Q} \subset \{1, 2, \ldots, n\}$, whose cardinality is denoted by q, be the set of sensor indices that are activated for distributed beamforming. When a single sensor $\mathtt{S}_i$ is used, $\Omega = \{i\}$ and $q = 1$, while $\Omega = \{1, 2, \ldots, n\}$ and $q = n$ if all the available sensors are selected. Given the set of active sensors Ω, the received signal at the FC is given by

$$\mathbf{y} = \sum_{i \in \Omega} w_i h_i \mathbf{s} + \mathbf{v}, \tag{7.1}$$

where w_i is the beamforming weight at $\mathtt{S}_i$ and $\mathbf{v} \sim \mathcal{CN}(0, N_0\mathbf{I})$ is the background noise at $\mathtt{D}$. The transmission power at $\mathtt{S}_i$ is therefore given by $P_i = |w_i|^2$.

With cooperative transmissions, it is well known that the energy efficiency can be improved because the required transmission power is reduced. Let SNR_D be the SNR level required by $\mathtt{D}$ so that it can correctly decode the transmitted packets. For example, when the message of RL bits is encoded to become a packet of L symbols and a random Gaussian codebook is employed, $\mathsf{SNR}_\mathsf{D} = \left(2^R - 1\right)$. If only a single random sensor $\mathtt{S}_i$, that knows its own channel gain h_i, is used to transmit, it is required that $w_i = \frac{N_0 \mathsf{SNR}_\mathsf{D} h_i^*}{\alpha_i}$, corresponding to the transmission power of $P_i = \frac{N_0 \mathsf{SNR}_\mathsf{D}}{\alpha_i}$, to achieve the target SNR. On the other hand, if multiple sensors $\mathtt{S}_i$'s, $i \in \Omega$, are randomly chosen, the beamforming weights and the total transmission power are $w_i = \frac{N_0 \mathsf{SNR}_\mathsf{D} h_i^*}{\sum_{j \in \Omega} \alpha_j}$ and $P_\Omega = \sum_{i \in \Omega} |w_i|^2 = \frac{N_0 \mathsf{SNR}_\mathsf{D}}{\sum_{i \in \Omega} \alpha_i}$, respectively. Assuming that $f(\alpha)$ is an exponential function (Rayleigh fading), $\mathbb{E}[P_i] = \infty$ due to a non-zero probability of $|h_i|^2$ being zero, i.e., an infinite power would be required when a single random sensor is used. However, with the distributed beamforming, $\mathbb{E}\left[P_\Omega\right] = \frac{\mathbb{E}[P_i]}{q-1}$, which is finite when $q > 1$. This shows a great benefit of cooperative transmissions in terms of the energy efficiency.

In the discussion above, the CSI knowledge is assumed to be available at both sides of the link, the sensors and the FC, without any extra cost. Under this assumption, a trivial solution to the energy saving is to utilize all sensors. However, this assumption is non-realistic and the cost associated with the CSI acquisition often increases linearly with the number of sensors. If this cost is taken into account, utilizing all the sensors may not necessarily be the best option. In addition, how to obtain the CSI at sensors and, if only some sensors are used, and how to select good sensors needs to be studied. In the next section, we propose a scheme to select q sensors with the best channel gains powers among the n available sensors. We then analyze the total energy required for the transmission of a message to the FC in Section 7.4. This study provides a guideline in deciding a proper number of sensors for the minimum energy consumption per message given a set of system parameters.

7.3 Proposed Cooperative Transmission Scheme

As mentioned, the FC collects information from the sensors periodically. In response to the FC request, the sensors cooperatively transmit until the transmission of the message is completed. In addition, we assume that the channel gains remain constant over a duration longer than the time required for cooperative transmission, i.e., slow fading assumption. Thus, the distributed beamformers can be used by activated sensors until the last symbol of the current data packet is sent to the FC. Overall, communication is performed in four phases as illustrated in Fig. 7.2.

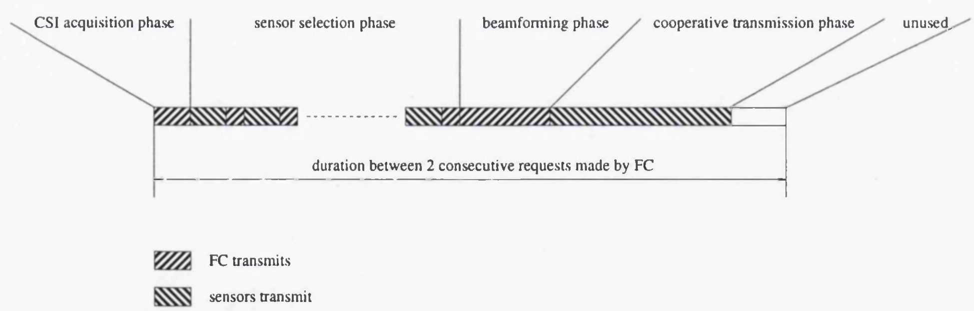

Figure 7.2: The proposed cooperative transmission scheme with four phases

- **CSI acquisition phase:** The FC broadcasts a pilot packet of R_t symbols to the sensors with the power P_t. This pilot packet is also used to inform the sensors that the current information is requested by the FC. Based on the received pilot symbols, each sensor estimates its channel gain. The energy required for this phase is given by

$$E_{\text{bc}} = R_t P_t. \tag{7.2}$$

In general, there is a trade-off between the transmission power and the number of pilot symbols in order to achieve a certain estimation quality. If the transmission power is lowered, more pilot symbols are required. Nevertheless, the required amount of energy is independent of the message to be transmitted and the number of sensors. In this chapter, we consider this amount of energy to be a fixed parameter.

- **Sensor selection phase:** The q best sensors are distributively selected based on a binary splitting algorithm, similar to the one described in Chapter 5. In principle, this phase consists of a sequence of time slots (or rounds), in which the sensors independently polls for selection. At the beginning of a time slot, each sensor compares its channel gain with a threshold α_{th} to decide whether or not to vote to be selected. A voting decision is a small message containing the information about the identity of the voting sensor added to a pilot packet of R_t symbols which, in turn, can be used by the FC to estimate the channel coefficient of the wireless link from the selected sensor after the voting decision can be detected successfully. Specifically, we assume that the identity part of the voting decision is negligible and that the pilot symbols are transmitted with the power P_t. At the end of each time slot in the sensor selection phase, the FC feeds a ternary indicator back to the sensors to inform them about the voting result it detected. Normally, if no sensor voted, the broadcast feedback indicator is a idle (0) symbol. When only one sensor voted, the feedback is a success (1). Otherwise, the feedback indicator is a collision (e). Let the set of sensors that have not yet selected and the set of sensors with the right to vote be the backlogged set and the active set, respectively. At the beginning of each time slot, α_{th} is recomputed based on the FC feedback so that the probability of each sensor deciding to vote in the active set is $\frac{p_c}{N_b}$ if the active set and the backlogged set are identical, or p_s otherwise. Here, p_c and p_s are the parameters of the algorithm, known as the contention factor and the selection factor, and N_b is the number of sensors in the backlogged set.

Algorithm 2 shows behavior of $\mathtt{S}_i$ in the selection phase. This algorithm is, in fact, an application specific version of Algorithm 1 in Chapter 5 where the metric associated with each sensor is the power gain of the channel from the sensor to the FC. Here, *get(feedback)* is the procedure of detecting the feedback indicator sent by the FC and

$$split(a, b, p) \triangleq F^{-1}\big(pF(a) + (1-p)F(b)\big), \tag{7.3}$$

is the splitting function. Here $F(\alpha) = \int_0^\alpha f(x)dx$ is the cumulative distribution function (CDF) of α_i. Note that when the channel gains are independent and identically distributed (IID), the threshold values are identical for all the sensors and the sensors with better channel conditions are always selected. However, this algorithm only provides a (proportional) fair selection if the sensors have different channel gain distributions.

Triggered by a single request from the FC, let $L_n^{(q)}$ and $M_n^{(q)}$ be the number of the feedback indicators used by the FC (the number of selection rounds) and the number of transmissions used by all sensors in selecting the q best sensors, respectively. The energy consumed during the selection phase is given by

$$E_{\text{sel}} = L_n^{(q)} R_t P_t + M_n^{(q)} R_f P_f, \tag{7.4}$$

where R_f and P_f are the number of symbols and the transmission power of the feedback indicators, respectively.

Algorithm 2 Algorithm performed in the selection phase.

Input: $n, q, p_c, p_s, F(\cdot)$
Output: *selected*
$N_{\text{sel}} \leftarrow 0$; $N_{\text{bl}} \leftarrow N$;
$selected \leftarrow 0$; $voted \leftarrow 0$;
$feedback \leftarrow 0$; $prevfeedback \leftarrow 0$;
$level \leftarrow 0$; $\alpha_{\text{low}}(level) \leftarrow \infty$;
while $N_{\text{sel}} < q$ **do**
if $feedback = 1$ **then**
$N_{\text{sel}} \leftarrow N_{\text{sel}} + 1$; $N_{\text{bl}} \leftarrow N_{\text{bl}} - 1$;
if *voted* **then**
$selected \leftarrow 1$;
end if
end if
if $!selected$ **then**
if $feedback = e$ **then**
$level \leftarrow level + 1$;
$\alpha_{\text{up}}(level) \leftarrow \alpha_{\text{up}}(level - 1)$;
$\alpha_{\text{low}}(level) \leftarrow split(\alpha_{\text{low}}(level - 1), \alpha_{\text{up}}(level - 1), p_s)$;
else
if $feedback = 0$ **and** $prevfeedback = e$ **then**
$\alpha_{\text{up}}(level) \leftarrow \alpha_{\text{low}}(level)$;
$\alpha_{\text{up}}(level - 1) \leftarrow \alpha_{\text{low}}(level)$;
$\alpha_{\text{low}}(level) \leftarrow split(\alpha_{\text{low}}(level - 1), \alpha_{\text{up}}(level - 1), p_s)$;
else
if $level > 0$ **then**
$level \leftarrow level - 1$;
else
$\alpha_{\text{up}}(level) \leftarrow \alpha_{\text{low}}(level)$;
$\alpha_{\text{low}}(level) \leftarrow split(0, \alpha_{\text{up}}(level), \frac{p_c}{N_{\text{bl}}})$;
end if
end if
end if
$\alpha_{\text{th}} \leftarrow \alpha_{\text{low}}(level)$; $voted \leftarrow (|h_i|^2 > \alpha_{\text{th}})$;
$prevfeedback \leftarrow feedback$;
$get(feedback)$;
end if
end while

As discussed in Chapter 6, the expectation of $L_n^{(q)}$ and $M_n^{(q)}$ can be approximated asymptotically. Furthermore, it was shown that the average duration of an algorithm and the expected number of transmissions increase with the number of sensors q to be selected. Consequently, the more sensors are selected, the higher energy would be required in this phase.

- **Beamforming phase:** The cooperative beamformers are efficiently constructed if each of the selected sensors knows the sum of the channel gains of all the active sensors. Let $\mathsf{S}_{[1]}, \mathsf{S}_{[2]}, \ldots, \mathsf{S}_{[q]}$ denote the set of the q selected sensors sorted in a descending order of the channel power gains, i.e., let $\alpha_{[1]} > \alpha_{[2]} > \ldots > \alpha_{[q]}$. Since $h_{[i]}, i = 1, 2, \ldots, q$, are known to $\mathtt{D}$, it transmits a message of R_b symbols with the power $\frac{N_0 \mathsf{SNR}_\mathsf{S}}{\alpha_{[q]}}$, where SNR_S is the required SNR identical at all sensors, to inform all the selected sensors about the sum $\sum_{i=1}^{q} \alpha_{[i]}$. The energy required in this phase is given as

$$E_{\text{bf}} = \frac{R_b N_0 \mathsf{SNR}_\mathsf{S}}{\alpha_{[q]}}. \tag{7.5}$$

As can be seen from (7.5), the larger the number of sensors q is used, the more likely that $\alpha_{[q]}$ gets smaller which results in a higher amount of energy consumption in this phase.

- **Cooperative transmission phase:** Upon knowing the sum $\sum_{i=1}^{q} \alpha_{[i]}$, a distributed beamformer can be constructed by the active sensors. All q selected sensors cooperatively send the common packet $\mathbf{s}$ of L symbols to the FC. Specifically, the ith best sensor, $\mathsf{S}_{[i]}$, transmits with the power $\frac{N_0 \mathsf{SNR}_\mathsf{D} \alpha_{[i]}}{\left(\sum_{j=1}^{q} \alpha_{[j]}\right)^2}$. The energy required for this phase is given as

$$E_{\text{coop}} = \frac{L N_0 \mathsf{SNR}_\mathsf{D}}{\sum_{i=1}^{q} \alpha_{[i]}}. \tag{7.6}$$

Similar to the discussion for the beamforming phase, one can see that the sum $\sum_{i=1}^{q} \alpha_{[i]}$ gets larger as more sensors are selection. Thus, Eq. (7.6) implies that, the more sensors are selected, the less energy would be required in this phase.

Overall, the total energy required to transmit a packet to the FC is given by

$$E_{\text{tot}} = E_{\text{bc}} + E_{\text{se}} + E_{\text{bf}} + E_{\text{coop}}. \tag{7.7}$$

7.4 Average Energy Consumption per Message

In this section, we analyze the expected energy required to transmit a data packet of L symbols using the proposed cooperative transmission scheme given the parameters n, q, $\mathsf{SNR_D}$, $\mathsf{SNR_S}$, P_t, R_t, P_f, R_f, and R_b. Throughout this section, we assume that $f(\alpha) = \exp(-\alpha)$, $\alpha \geq 0$, i.e., all communication links operate over Rayleigh fading channel with the unit mean [1].

In the selection phase, assuming that the system consist of a relatively large number of sensors, the asymptotic expressions can be used to approximate expectations of $L_n^{(q)}$ and $M_n^{(q)}$. Eq. (7.4) implies that the average energy required for the sensor selection $\mathbb{E}[E_{\text{sel}}]$ is a weighted sum of the expectation of these two measurements. In this case, the parameters p_c and p_s can be optimized with relatively complex computations as discussed in Chapter 6. However, it was shown that the expectations of these measures would increase somewhat linearly with the number of sensors q to be selected, so it is more important to optimize this parameter rather than optimize p_c and p_s. Thus, we use the suboptimal choices of $p_c = 1$ to maximize the probability of having a sensor selected in the first round and $p_s = 0.5$ for the fair splitting.

Note that, when $n = \infty$, the lower bounds of $l^{(q)}$ (the average duration of the selection phase) and $\mu^{(q)}$, (the expected number of transmissions required to select

[1]This assumption, used in our analyses, is not required for the operation of scheme proposed in the previous section.

q users) were derived in Chapter 5. These bounds were shown to be useful in capturing the behaviors of these measures as the number of sensors to be selected increases as in Chapter 6. We rely on these lower bounds to estimate the energy required for the selection phase. Denote by $\tilde{l}^{(q)}$ and $\tilde{\mu}^{(q)}$ the corresponding lower bounds. With $p_c = 1$ and $p_s = 0.5$, from Theorem 1, we have

$$\tilde{l}^{(q)} = \frac{1}{e-1}\left(1 + \sum_{m=1}^{q-1} \frac{l^{(q-m)} + \log_2(m) + m}{m!} + \sum_{m=q}^{\infty} \frac{\log_2(m) + q}{m!}\right), \tag{7.8}$$

$$\tilde{\mu}^{(q)} = \frac{1}{e-1}\left(\sum_{m=1}^{q-1} \frac{\mu^{(q-m)} + 3m - 2}{m!} + \sum_{m=q}^{\infty} \frac{2m + q - 2}{m!}\right), \tag{7.9}$$

After substituting these bounds in (7.5), we get the approximate average energy required for the sensor selection as

$$\mathbb{E}[E_{\text{sel}}] \approx \tilde{l}^{(q)} Q_t P_t + \tilde{\mu}^{(q)} Q_f P_f. \tag{7.10}$$

In the beamforming phase, since α_i are independent and exponentially distributed, $\alpha_{[q]}$, the qth element in the descending order list of the channel power gain, can be written as [82]

$$\alpha_{[q]} = \sum_{i=1}^{n-q+1} \frac{1}{n-i+1} \beta_i, \tag{7.11}$$

where $\beta_i = i(\alpha_{[n-i+1]} - \alpha_{[n-i]})$. Note that β_i are independent and distributed as α_i. Let $f_{[q]}(\alpha)$ denote the PDF of $\alpha_{[q]}$. From (7.11), the closed-form expression of $f_{[q]}(\alpha)$ can be found, which is a weighted sum of the exponential functions. The average energy for the beamforming phase is given by

$$\mathbb{E}\left[E_{\text{bf}}\right] = R_b N_0 \mathsf{SNR}_{\mathsf{S}} \int_0^{\infty} \frac{1}{\alpha} f_{[q]}(\alpha) d\alpha. \tag{7.12}$$

To compute $\mathbb{E}\left[E_{\text{bf}}\right]$, a numerical method in [83] can be used. Note that $f_{[q]}(0) = 0$ if $q < n$, therefore, $\mathbb{E}\left[E_{\text{bf}}\right]$ is finite, but increases with q. The case of selecting all the available sensors ($q = n$) is not practical under the assumption of Rayleigh fading as the infinite amount of energy would be required for the beamforming phase.

Finally, we now consider the cooperative transmission phase. Define $\alpha_{[1:q]} = \sum_{i=1}^{q} \alpha_{[i]}$ and denote its PDF by $f_{[1:q]}(\alpha)$. From (7.11), we have

$$\alpha_{[1:q]} = \sum_{i=1}^{n-q+1} \frac{q}{n-i+1}\beta_i + \sum_{i=n-q}^{n} \beta_i. \tag{7.13}$$

Similar to $f_{[q]}(\alpha)$, the closed-form expression of $f_{[1:q]}(\alpha)$ can be found. The average energy for the cooperative transmission phase is given by

$$\mathbb{E}\left[E_{\text{co}}\right] = LN_0\mathsf{SNR}_\mathsf{D} \int_0^\infty \frac{1}{\alpha} f_{[1:q]}(\alpha) d\alpha. \tag{7.14}$$

It can be seen that, opposed to the case of using a single random sensor, $\mathbb{E}\left[E_{\text{co}}\right]$ is finite even for $q = 1$ (i.e., only the best sensor is activated). This shows the great benefit of the sensor selection to energy savings. In addition, since $\alpha_{[1:q]}$ is the weighted sum of n IID random variables as shown in (7.13), a diversity of order n can be achieved for any number of selected sensors with better channel gain powers, provided that there are n sensors available. Furthermore, the weights in (7.13) can be seen as as the cooperation or array gain. Note that the weights increase with q, therefore, more energy is saved in this phase if more sensors are selected.

7.5 Numerical and Simulation Results

Our simulation set up follows the assumption in the previous section, that is, all communication channels are independent and exponentially distributed with the unit mean power (i.e., the Rayleigh fading). Both the SNRs required at the FC and the sensors are assumed to be $\mathsf{SNR}_\mathsf{D} = \mathsf{SNR}_\mathsf{S} = 1$. In addition, the energy for the CSI acquisition is assumed to be fixed and constant, which is $E_{\text{bc}} = R_t P_t = 5$. Moreover, in the sensor selection phase, we assume that the transmission power for (ternary) feedback $P_f = 1$ and the number of symbols in the feedback packet $R_f = 1$. In the beamforming phase, the FC reports the sum of the channel gain powers from the selected sensors in an encoded packet of $R_b = 4$ symbols.

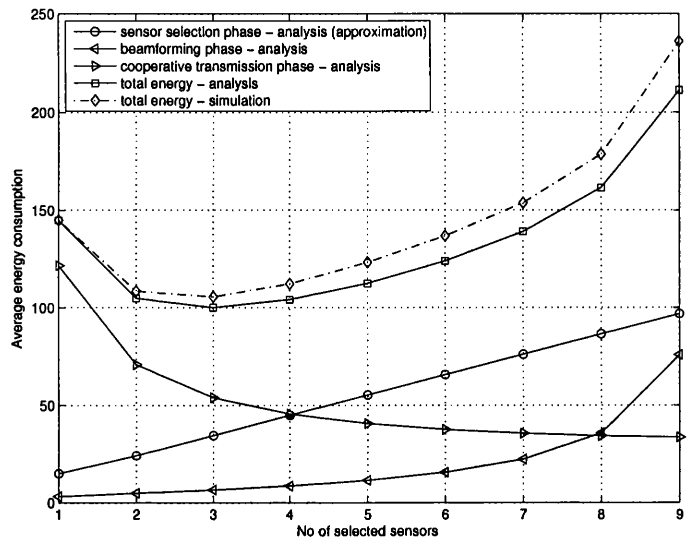

Figure 7.3: The energy consumption for the different number of selected sensors

When the packet length of $L = 300$ and there are $n = 10$ available sensors, Fig. 7.3 presents the average energy consumed per message for the different number of selected sensors (q varies from 1 to 9). It can be seen that the energy required for cooperative transmissions, $E_{\rm co}$, is reduced when more sensors are selected. However, the reduction decreases as q increases since the contribution of the worst sensors becomes less significant in comparison with the better ones. On the other hand, since the expected number of rounds and transmissions for the distributed sensor selection increases with q, the energy for the selection phase, $E_{\rm sel}$, is shown to grow with q. Moreover, because the higher power is required to report the sum of the channel gains (being inversely proportional to the qth largest gain amongst all the channels) when q increases, the energy for the beamforming phase, $E_{\rm bf}$, dramatically increases as q becomes larg. Therefore, the total energy would be a U-shape function of q as shown in Fig. 7.3. The simulation results clearly confirm this observation [1].

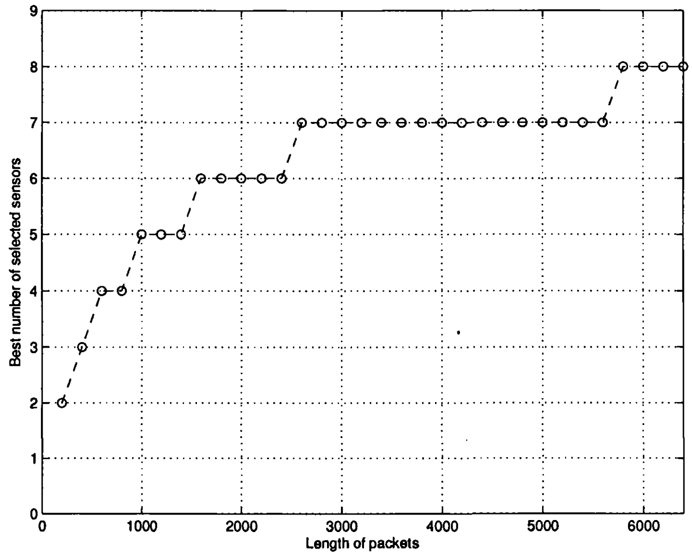

Figure 7.4: The optimal number of sensors to be selected for the different packet length

From the design point of view, a proper choice of q, the number of activated sensors, depends on the system parameters. For example, if the packet length L increases, i.e., more information is transmitted per request, while other parameters are fixed, the energy for the cooperative transmission, $[E_{\text{co}}]$, increases and becomes a dominant factor in the total energy. Therefore, q should be increased to exploit the benefits of the cooperative transmissions. On the other hand, if less information is requested, q should be decreased to reduce the overhead incurred in the sensor selection and the beamforming phases. This trend is showed in Fig. 7.4 where the optimal number of sensors is plotted against the different values of the packet length L numerically.

[1] As expected, there is a gap between the simulation and the analysis using the lower bound approximations. However, the gap only becomes significant when the number of selected sensors q is relatively large, the design rules are less affected.

7.6 Concluding Remarks

In this chapter, we have demonstrated application of the splitting based MSA developed in the previous chapters. Assuming a WSN in which the FC periodically collects information from the sensor nodes, we proposed a cooperative transmission scheme with four phases: CSI acquisition, sensor selection, beamforming, and cooperative transmission, and then discussed the energy consumption per message. Assuming also the TDD mode of communication, initially, each sensor estimates the condition of its channel to the FC. Based on the estimated channel coefficient, the sensors participate in a polling session so that the FC can select the most suitable sensors for cooperative transmission. A beamformer is then constructed by the selected sensors and finally, sensing data is cooperatively transmitted to the FC. Our analysis showed that the user selection can bring great benefits for the energy savings. In addition, if more sensors are selected to send information to the FC, the energy efficiency in the cooperative transmission phase is improved but the required overhead (the energy consumption) for the sensor selection and beamforming also increases. This trade-off and its the analyses in this chapter can be used as a guideline to decide the optimal number of active sensors for the distributed beamforming as illustrated by our numerical results.

Chapter 8

Conclusions and Future Works

In this thesis, we proposed a number of distributed scheduling schemes for the wireless communication systems assuming the cross layer design context. We considered an uplink random access network in which multiple users communicate with a common base station (BS) over a shared transmission medium. In addition, we performed a comprehensive study of the splitting based multiuser selection algorithms which are considered to be simple, effective, and scale well with the network size.

First, we investigated the effect of the channel memory in the *channel aware* ALOHA system. For a simple correlation model of the communication channels between the users and the BS, a reservation-type protocol was proposed. Various Markovian models were used to capture the behavior of the system. The average throughput of the system was obtained using the Markov Analysis technique and we showed that the proposed reservation protocol can achieve better performance by reducing the probability of transmission collisions.

Second, in order to improve the efficiency of the *Opportunistic Multichannel* ALOHA (OMC-ALHOA) scheme, we proposed a simple extension to the transmission policy that exploits the idle channels. The basic idea is to allow the users who have sufficiently good channel conditions to access these idle channels and achieve better resources utilization. The performance analysis showed that, theoretically, the maximum system throughput can be improved by up to 63% in the asymptotic case. Through the numerical results, it can be seen that a significant gain is achieved even when the system consists of a small number of users. Moreover, as the number of users increases, the system performance also improves at the same rate with the centralized scheme, which shows that both the multiuser and the multichannel diversities are preserved in the proposed scheme.

Third, we considered a generalized version of the splitting based multiuser selection algorithm (MSA) in a probabilistic view. In the asymptotic case, we showed that the average duration and the expected number of transmissions are given in the form of a functional equation, similar to the analysis of the collision resolution algorithm [12, 13, 14]. Subject to some *contraction conditions*, the solution of the functional equation can be obtained, which provided approximations of the expectations of both measures of interest in a system with many users. These results shed light to the design problems in choosing the parameters for the algorithm when considering the delay and the overhead jointly. A typical example is to optimize the parameters that minimize a weighted sum of the measures of interest. Illustrative examples supported by the numerical results were used to show the benefits of optimally choosing the parameters for the different design purposes. In addition, we derived the lower bounds of the expected number of slot and the expected number of transmissions required by the algorithms. These bounds capture the behavior of these measures as the number of users to be selected q increases, and they are particularly useful if q is considered as the main design parameter.

Last but not least, we demonstrated application of the generalized splitting based MSA in a study of the energy efficiency in a wireless sensor network with the distributed beamforming. Assuming the scenario that multiple sensors have identical sensing information and would like to cooperatively transmit signals to a fusion center, we proposed a transmission scheme consisting of four phases. Importantly, our proposal considers a new *sensor selection phase* that physically selects the best sensors for the cooperative transmission and also maximizes the energy efficiency. Our analysis has shown that there is a tradeoff between the energies required for the sensor selection plus the beamforming phases and the energies required for the cooperative transmission phase in deciding the number of sensors to be selected. This observation was captured by the numerical and simulation results, which provide a design guideline for the energy savings and prolonging the network lifetime.

From the results presented in this work, the following extensions are possible.

- In the reservation scheme that we studied in Chapter 3, we assume that the channel is reserved for as long as the reserving user is still in a "GOOD" condition. The proposed scheme does have a side effect. Once the user reserves the channel, other users have to wait for the channel to be released before they can start contending for the channel, thereby causing a delay. In a delay constrained application, this side effect is undesirable and shall be avoided. One possible extension is to study a congestion control scheme which takes into account the number of packets in the system. In this case, the users adaptively decide to reserve or release the channel depending on the length of the queues.

- In principle, the reservation-type protocol in Chapter 3 aims at reducing the probability of collision while the proposed scheme for a multichannel system in Chapter 4 improves the resource utilization by exploiting idle channels. Considering to reduce the collisions and exploit idle channels jointly could be an interesting study in a OMC-ALOHA framework.

- In this thesis, we only consider a standard collision model. There have been many other models and distributed schemes studied with the multipacket reception (MPR) capability. In future work, the schemes can be investigated in conjunction with the MPR.

- For the MSA, we noted that the computational complexity of the analytical formulas is rather high when the number of users to be selected is relatively large. Although the lower bounds were derived and were proved to be useful, it is desirable to capture the behavior of the main measures in more simple and (hopefully) linear approximate forms.

- Furthermore, we noted that there is a close connection between the MSA and the collision resolution algorithm (CRA). Therefore, other variations suggested for the CRA such as a V*-ary* splitting and *tree pruning* could be applied and shall be investigated with multiuser selection.

- Finally, we demonstrated the application of the novel MSA with an energy efficient distributed beamforming scheme in a wireless sensor network. A clear future research direction is to propose and study other practical application scenarios employing such distributed multiuser selection.

Appendix A

Markov Chain

A.1 Introduction

A Markov chain is a stochastic process with the Markov property, namely that, the conditional probability distribution of future states of the process depends only upon the present state but not on the sequence of events that preceded it. The changes of state of the system (the Markov chain) are called transitions and the probabilities associated with various state-changes are called the transition probabilities.

Usually, a Markov chain is defined as a random process which is being in a certain state at a given step (time) and the state of the process is changing randomly between steps (i.e., a discrete-time Markov chain). In this case, a formal definition for Markov chain can be as follows.

Definition 3 A Markov process is a sequence of random variables $X_1, X_2, X_3 \ldots$ taking valuable in state space $\mathcal{X}$ such that

$$\Pr\{X_{n+1} = x | X_1 = x_1, \ldots, X_n = x_n\} = \Pr\{X_{n+1} = x | X_n = x_n\}. \tag{A.1}$$

The state space $\mathcal{X}$ is often considered to be discrete (and countable). The terminology **Markov Chain** is, however, also used for stochastic process with

Markov property where the state space is continuous or "time" can take continuous values [84]. Note that, the discussion about Markov chain in this appendix only concentrates on the discrete-time discrete-state-space case, in which, a Markov chain can be completely characterized by a state space cX and all the transition probabilities.

Since the system (i.e., the Markov chain) changes randomly, it is generally impossible to predict with certainty the state of a Markov chain at a given point in the future. However, the statistical properties of the system's future can be predicted. In many applications, it is these statistical properties that are important.

A.2 Concepts and Properties

A.2.1 State Transition

Consider a discrete time Markov chain $\mathbf{X}$ that is being in state i initially. Let the single-step transition probability from state i to j is

$$p_{ij} = \Pr\{X_1 = j | X_0 = i\}, \tag{A.2}$$

and the probability of going from state i to state j in n steps is

$$p_{ij}^{(n)} = \Pr\{X_n = j | X_0 = i\}. \tag{A.3}$$

The n-step transition probabilities satisfy the Chapman-Kolmogorov equation [84], that is, for any k such that $0 < k < n$,

$$p_{ij}^{(n)} = \sum_{r \in \mathcal{X}} p_{ir}^{(k)} p_{rj}^{(n-k)}, \tag{A.4}$$

where $\mathcal{X}$ is the state space of the Markov chain.

Assume that the state space $\mathcal{X}$ is finite, the one step transition probability distribution can be represented by a matrix $\mathbf{P}$, called the transition matrix, with the (i, j)th element of $\mathbf{P}$ equal to p_{ij}. Since each row of $\mathbf{P}$ sums to one and all elements are non-negative, $\mathbf{P}$ is a right stochastic matrix.

The Markov chain is said to be (discrete) time homogeneous if

$$p_{ij} \quad = \Pr\{X_{k+1} = j | X_k = i\} \text{ and} p_{ij}^{(n)} \quad = \Pr\{X_{k+n} = j | X_k = i\}, \qquad \text{(A.5)}$$

that is, transition probabilities depend on elapsed time, not absolute time. In this case, the transition matrix $\mathbf{P}$ is the same after each step so the k-step transition probability can be computed as the k-th power of the transition matrix, $\mathbf{P}^k$.

A.2.2 Reducibility and Periodicity

A state j is said to be **accessible** from a state i (written $i \rightarrow j$) if a system started in state i has a non-zero probability of transitioning into state j at some point. Formally, state j is accessible from state i if there exists an integer $n > 0$ such that

$$\Pr\{X_n = j | X_0 = i\} = p_{ij}^{(n)} > 0. \qquad \text{(A.6)}$$

A state i is said to be gcommunicate with state j (written $i \leftrightarrow j$) if both $i \rightarrow j$ and $j \rightarrow i$. A set of states $\mathcal{C}$ is a communicating class if every pair of states in $\mathcal{C}$ communicates with each other, and no state in $\mathcal{C}$ communicates with any state not in $\mathcal{C}$. A communicating class is closed if the probability of leaving the class is zero, namely that if i is in $\mathcal{C}$ but j is not, then j is not accessible from i.

A state i is said to be gessential if for all j such that $i \rightarrow j$, it is also true that $j \rightarrow i$. A state i is **inessential** if it is not essential [85].

Furthermore, a Markov chain is said to be **irreducible** if its state space is a single communicating class. In other words, it is possible to get to any state from any state with an irreducible Markov chain.

A state i has **period** k if any return to state i must occur in multiples of k time steps. Formally, the period of a state is defined as

$$k = \gcd\{n : p_{ij}^{(n)} > 0\}, \tag{A.7}$$

where "gcd" is the greatest common divisor. Note that even though a state has period k, it may not be possible to reach the state in k steps. For example, suppose it is possible to return to the state in $\{6, 8, 10, 12, \ldots\}$ steps; k would be 2, even though 2 does not appear in this list.

If $k = 1$, then the state is said to be **aperiodic**, i.e., the returns to the state i can occur at irregular times. In other words, a state i is **aperiodic** if there exists n such that for all $n' \geq n$,

$$\Pr\{X_{n'} = i | X_0 = i\} = p_{ii}^{(n')} > 0. \tag{A.8}$$

Otherwise (k ¿ 1), the state is said to be **periodic** with period k.

Finally, a state i is called absorbing if it is impossible to leave this state. Therefore, the state i is absorbing if and only if $p_{ii} = 1$ and $p_{ij} = 0 \forall i \neq j$. If every state can reach an absorbing state, then the Markov chain is an absorbing Markov chain.

A.2.3 Recurrence and Ergodicity

A state i is said to be **transient** if, given that we start in state i, there is a non-zero probability that we will never return to i. Formally, let the random variable T_i be the first return time to state i (the *hitting time*):

$$T_i = \inf\{n \geq 1 : X_n = 1 | X_0 = i\}. \tag{A.9}$$

In this case, the number $f_{ii}^{(n)} = \{T_i = n\}$ is the probability that we return to state i for the first time after n steps. Therefore, state i is transient if

$$\Pr\{T_i < \infty\} = \sum_{n=1}^{\infty} f_{ii}^{(n)} < 1. \tag{A.10}$$

State i is **recurrent** (or **persistent**) if it is not transient. It can be shown that a state i is recurrent if and only if the expected number of visits to this state is infinite, i.e., $\sum_{n=1}^{\infty} p_{ii}^{(n)} = \infty$. Recurrent states have finite hitting time with probability 1.

Note that, even if the hitting time is finite with probability 1, it need not have a finite expectation. The mean recurrence time M_i at state i is the expected return time

$$M_i = E[T_i] = \sum_{n=1}^{\infty} n f_{ii}^{(}n). \tag{A.11}$$

State i is **positive recurrent** (or **non-null persistent**) if M_i is finite; otherwise, state i is **null recurrent** (or **null persistent**).

Finally, a state i is said to be **ergodic** if it is aperiodic and positive recurrent. If all states in an irreducible Markov chain are ergodic, then the chain is said to be ergodic. Nevertheless, a finite state irreducible Markov chain is ergodic if it has an aperiodic state. A chain that has the ergodic property if there's a finite number N such that any state can be reached from any other state in exactly N steps. In case of a fully connected transition matrix where all transitions have a non-zero probability, this condition is fulfilled with $N = 1$. A model with more than one state and just one out-going transition per state cannot be ergodic.

A.2.4 Steady State Analysis and Limiting Distribution

Recall from subsection A.2.1, if the Markov chain is a time-homogeneous Markov chain, the process is completely characterized by a state space $\mathcal{X}$ and a single, time-independent transition matrix $\mathbf{P}$. The non-negative vector $\mathbf{\Pi}$ is called a stationary distribution if its entries π_i are sum to 1 and

$$\pi_j = \sum_{i \in \mathcal{X}} \pi_i p_{ij}, \tag{A.12}$$

or in matrix notation $\mathbf{\Pi} = \mathbf{\Pi P}$. In other words, the stationary distribution $\mathbf{\Pi}$ is a normalized (meaning that the sum of its entries is 1) left eigenvector of the transition matrix associated with the eigenvalue 1. Furthermore, by linear substitution of the matrix notation,

$$\mathbf{\Pi} = \mathbf{\Pi P} = \underbrace{\mathbf{\Pi P}}_{\mathbf{\Pi}} \mathbf{P} = \dots, \tag{A.13}$$

$\mathbf{\Pi}$ can be alternatively viewed as a fixed point of the linear (hence continuous) transformation on the unit simplex associated with the matrix $\mathbf{P}$. As any continuous transformation in the unit simplex has a fixed point, a stationary distribution always exists, but is not guaranteed to be unique, in general. However, if the Markov chain is irreducible and aperiodic, then there is a unique stationary distribution $\mathbf{\Pi}$. Additionally, in this case $\mathbf{P}^k$ converges to a rank-one matrix in which each row is the stationary distribution $\mathbf{\Pi}$, that is,

$$\lim_{k \to \infty} \mathbf{P}^k = \mathbf{1\Pi}, \tag{A.14}$$

where $\mathbf{1}$ is the column vector with all entries equal to 1.

Note that, there has been no assumption on the starting distribution; the chain converges to the stationary distribution regardless of where it begins. Such $\mathbf{\Pi}$ is called the **equilibrium distribution** of the chain.

Appendix B

Introduction to the Background of the Mathematics in Chapter 6

B.1 Functional Equation

Functional equations are equations of unknown functions instead of unknown numbers, that is, an equation of the form $f(x, y, z, \ldots) = 0$, where $f(.)$ contains a finite number of independent variables, known functions, and unknown functions which are to be solved for. A notable example is the Cauchy's functional equation

$$f(x+y) = f(x) + f(y), \tag{B.1}$$

of which, solutions to it are called additive functions. Over the rational numbers, it can be shown using elementary algebra that there is a single family of solutions, namely $f(x) = cx$ for any arbitrary rational number c . Over the real numbers, this is still a family of solutions; however there can exist other solutions that are extremely complicated.

Strictly, a functional equation is any equation that specifies a function in implicit form that cannot be simply reduced to algebraic equations. Often, the equation relates the value of a function (or functions) at some point with its values at other points. Many properties of functions can be determined by studying the types of functional equations they satisfy. For example, the gamma function $\Gamma(.)$ satisfies the functional equations

$$\Gamma(z+1) = z\Gamma(z). \tag{B.2}$$

The Riemann zeta function also satisfies a functional equation, called the Riemann('s) functional equation

$$\zeta(s) = 2^s\pi^{(s-1)} \sin\left(\frac{\pi s}{2}\right)\Gamma(1-s)\zeta(1-s). \tag{B.3}$$

Solving functional equations can be very difficult but there are some common methods. The main approach to solve the elementary functional equations is substitution. Induction is a useful technique to use when the function is only defined for rational or integer values. In dynamic programming a variety of successive approximation methods can be used to solve functional equation, including methods based on fixed point iterations.

B.2 Semi-group

In mathematics, a **semi-group** is an algebraic structure consisting of a set H together with an *associative binary operation* ($\bullet$) that combines any two of its elements to form a third element. For example, in Chapter 6, we encountered a semi-group of linear substitutions generated by functions σ_i, $i \in \{1, 2\}$, where

$$\begin{aligned}\sigma_1(p_c) &= p_s p_c,\\ \sigma_2(p_c) &= (1-p_s)p_c.\end{aligned} \tag{B.4}$$

With this semi-group, operation is the composition of functions. For example, the simplest composition generates 3 elements of the semi-group, i.e.,

$$\begin{aligned}\sigma_1 \bullet \sigma_1(p_c) &= p_s p_s p_c = p_s^2 p_c,\\ \sigma_1 \bullet \sigma_2(p_c) &= \sigma_2 \bullet \sigma_1(p_c) p_s(1-p_s)p_c,\\ \sigma_2 \bullet \sigma_2(p_c) &= (1-p_s)(1-p_s)p_c = (1-p_s)^2 p_c.\end{aligned} \tag{B.5}$$

Originally, the term semi-group is the generalization of the term **group**. In order to qualify as a group, the set and the operation must satisfy four conditions called the group axioms, namely *closure, associativity, identity* and *invertibility*, The semi-group, however, does not have the last condition , i.e., every element do not have to have an inverse (hence the name semi-group). Furthermore, a semi-group might not have identity either. Those with an identity are often called **monoid**, such as in our example of linear substitutions of functions above. In this specific case, there exists an identity element $\epsilon(p_c) = p_c$ and that

$$\sigma^k \bullet \epsilon(p_c) = \epsilon \bullet \sigma^k(p_c) = \sigma^k(p_c). \tag{B.6}$$

Note that, when working with the semi-group H in Chapter 6, we dropped the notation of the operator $(\bullet)$ in composing the elements and, hence, slightly abused the standard.

B.3 Entire Function

B.3.1 Definition and Representation

An **entire function**, also called an **integral function**, is a function of one or more complex variables that is complex differentiable in a neighborhood of every point in the complex plane, except, possibly, at the point at infinity. In other words, an entire function is **holomorphic** over the whole complex domain [1].

Due to the fact that an entire function $f(z)$ is complex analytic it can be expanded in a power series such as

$$f(z) = \sum_{k=0}^{\infty} a_k z^k, \tag{B.7}$$

[1]The term **analytic function** is often used interchangeably with **holomorphic function**, although the term *analytic* is commonly used in a broader sense to describe any function (real, complex, or of more general type) that is equal to its Taylor series in a neighborhood of each point in its domain. It can be proved that any complex function differentiable (in the complex sense) in an open set is analytic. Thus, in *complex analysis*, the class of **complex analytic functions** coincides with the class of **holomorphic functions**.

where

$$a_k = \frac{f^{(k)}(0)}{k!} \text{ for } k \geq 0, \tag{B.8}$$

which converges in the whole complex plane, i.e.,

$$\lim_{k \to \infty} |a_k|^{\frac{1}{k}} = 0. \tag{B.9}$$

If $f(z) \neq 0$ everywhere, then $f(z) = e^{P(z)}$, where $P(z)$ is an entire function. If there are finitely many points $z_1, z_2, \ldots$ at which $f(z)$ vanishes (these points are called the zeros of the function), then $f(z) = (z - z_1) \ldots (z - z_k)e^{P(z)}$, where $P(z)$ is, again, an entire function. In the general case where $f(z)$ has infinitely many zeros $z_1, z_2, \ldots$ there is a product representation (The Weierstrass factorization theorem on infinite products [73])

$$f(z) = z^{\alpha} e^{P(z)} \prod_{k=1}^{\infty} \left(1 - \frac{z}{z_k}\right) \exp\left(\frac{z}{z_k} \ldots \frac{z^k}{k z_k^k}\right), \tag{B.10}$$

where $P(z)$ is an entire function, $\alpha = 0$ if $f(0) \neq 0$, and α is the multiplicity of the zero $z = 0$ if $f(0) = 0$.

B.3.2 Properties and Classification

Every entire function can be represented as a power series that converges uniformly on compact sets (Taylor series expansion), that is, for any complex number z_0,

$$f(z) = a_0 + a_1(z - z_0) + \ldots \tag{B.11}$$

If conversely, such a power series converges for every complex value z, then the sum of the series is an entire function. Note that, the Weierstrass factorization theorem asserts that any entire function can be represented by a product involving its zeros.

Based on the Taylor series expansion, the entire functions may be divided in two disjoint classes

1. **The entire rational functions**, i.e. polynomial functions; in their series expansion there is an n_0 such that $a_n = 0 \; \forall \; n \geq n_0$.

2. **The entire transcendental functions**; in their series expansion one has $a_n \neq 0 \; \forall \; n$.

Note that the sum, the product and the composition of any two entire functions are entire functions. The entire functions on the complex plane form an integral domain (the Prfer domain).

Liouville's theorem [73] states that any bounded entire function must be constant. Conversely, the non-constant entire functions are unbounded, i.e., if $f(z)$ is a non-constant entire function and if R and M are two arbitrarily great positive numbers, then there exist such points z that $|z| > R$ and $|f(z)| > M$. As a consequence of Liouville's theorem, any function that is entire over the whole complex plane and the point at infinity is constant. Thus any non-constant entire function must have a singularity at the complex point at infinity, either a pole for a polynomial or an essential singularity for a transcendental entire function. Specifically, by the CasoratiWeierstrass theorem [73], for any transcendental entire function $f(z)$ and any complex w, there is a sequence (z_m) with $\lim_{m \to \infty} = \infty$ and $lim_{m \to \infty} f(z_m) = w$, In a more descriptive way, f comes arbitrarily close to any complex value in every neighbourhood of infinity.

Bibliography

[1] International Organization for Standardization, *Information Technology - Open Systems Interconnection - Basic Reference Model*, June 1996. 2, 10, 30

[2] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," Oct. 2003. 2, 30

[3] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication.* Cambridge University Press, 2005. 2, 30

[4] H. F. Rashvand and Y. S. Kavian, *Using Cross-Layer Techniques for Communication Systems: Techniques and Applications.* 2

[5] R. Knopp and P. Humblet, "Information capacity and power control in single-cell multiuser communications," in *Proc. IEEE ICC' 95*, (Seatle, WA), 1996. 2, 31, 43, 56

[6] X. Qin and R. Berry, "Exploiting multiuser diversity for medium access control in wireless networks," in *Proc. IEEE INFOCOM 2003*, (San Francisco, CA), pp. 1084–1094, 2003. 3, 31, 33, 35, 36, 44, 47, 49

[7] K. Bai and J. Zhang, "Opportunistic multichannel ALOHA: Distributed multiaccess control scheme for ofdma wireless networks," *IEEE Trans. Veh. Technol.*, vol. 55, pp. 848–855, May 2006. 3, 4, 7, 44, 45, 46, 47, 48, 51

[8] X. Qin and R. Berry, "Opportunistic splitting algorithms for wireless networks," in *Proc. IEEE INFOCOM 2004*, vol. 3, (Hong Kong), Mar. 2004. 3, 5, 36, 47, 49, 56, 63, 86

[9] V. Shah, N. B. Mehta, and R. Yim, "Splitting algorithms for fast relay selection: Generalization, analysis and a unified view," *IEEE Trans. Wireless Commun.*, vol. 9, pp. 1525–1535, Apr. 2010. 3, 5, 57, 62, 63, 86

[10] J. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE Trans. Inform. Theory*, vol. 25, pp. 505–515, Sept. 1979. 5, 12, 19, 20, 21, 26, 57

[11] B. S. Tsybakov and V. A. Mikhailov, "Free synchronous packet access in a broadcast channel with feedback," *Probl. Pered. Inform.*, vol. 14, pp. 32–59, Oct–Dec. 1978. 5, 12, 19, 20, 21, 26, 57

[12] G. Fayolle, P. Flajolet, M. Hofri, and P. Jacquet, "Analysis of a stack algorithm for random multiple access communication," *IEEE Trans. Inform. Theory*, vol. 31, pp. 244–254, Mar. 1985. 5, 22, 25, 112

[13] G. Fayolle, P. Flajolet, and M. Hofri, "On a functional equation arising in the analysis of a protocol for a multi-access broadcast channel," *Adv. Appl. Prob.*, vol. 18, pp. 441–472, 1986. 5, 22, 25, 112

[14] W. Szpankowski, "On a recurrence equation arising in the analysis of conflict resolution algorithms," *Stoch. Models.*, vol. 3, pp. 89–114, 1987. 5, 112

[15] Y. Yu and G. B. Giannakis, "Opportunistic medium access for wireless networking adapted to decentralized CSI," *IEEE Trans. Wireless Commun.*, vol. 5, pp. 1445–1455, June 2006. 7, 31, 32, 33, 47

[16] P. Viswanath, D. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Trans. Inform. Theory*, pp. 1277–1294, June 2002. 7, 31, 43, 47

[17] H. Zimmermann, "OSI reference model - the OSI model of architecture for open systems interconnection," *IEEE Trans. Commun.*, vol. 28, pp. 425–432, Apr. 1980. 10, 30

[18] IEEE Standard for Local and Metropolitan Area Networks, *Overview and Architecture*, Feb. 2001. 11

[19] D. Bertsekas and R. Gallager, *Data Networks.* Eaglewood Cliffs, NJ: Prentice-Hall, 1992. 11, 16, 17, 26, 33, 47, 59

[20] R. Rom and M. Sidi, *Multiple Access Protocols: Performance and Analysis.* Berlin: Springer-Verlag, 1989. 11, 16, 26, 33, 47, 59

[21] N. Abramson, "The ALOHA system: Another alternative for computer communications," in *Proc. AFIPS Conf.*, vol. 37, pp. 281–285, Nov. 1970. 12, 13, 15, 32

[22] G. Fayolle, E. Gelenbe, and J. Labetulle, "Stability and optimal control of the packet switching broadcast channel," *J. Ass. Comput.*, vol. 24, pp. 375–386, Mar. 1977. 12, 17, 19

[23] European Telecommunication Standardization Institute (ETSI), *Digital Video Broadcasting (DVB): Interation Channel for Satellite Distributed System, EN 301 790 V1.4.1*, Sept. 2005. 12

[24] N. Abramson, "The throughput of packet broadcasting channels," *IEEE Trans. Commun.*, vol. 25, pp. 117–128, Jan. 1977. 14

[25] N. Abramson, "Development of the AlohaNet," *IEEE Trans. Inform. Theory*, vol. 31, pp. 119–123, Mar. 1985. 15

[26] N. Abramson, "VSAT data networks," in *Proc. IEEE*, vol. 78, pp. 1267–1274, July 1990. 15

[27] D. G. Jeong and W. S. Jeon, "Performance of an exponential backoff scheme for slotted-ALOHA protocol in local wireless environment," *IEEE Trans. Veh. Tech.*, vol. 44, pp. 470–479, Aug. 1995. 15

[28] M. L. Molle and G. C. Polyzos, "Conflict resolution algorithms and their performance analysis," tech. rep., Department of Computer Science and Engineering, University of California, San Diego, July 1993. 16, 20, 21, 22, 23, 24, 26, 47, 85

[29] B. Hajek and T. v. Loon, "Decentralized dynamic control of a multiaccess broadcast channel," *IEEE Trans. Automat. Contr.*, vol. 27, pp. 559–569, June 1982. 17

[30] L. P. Clare, "Control procedures for slotted ALOHA systems that achieve stability," in *Proc. AMC SIGCOMM'86 Symp.*, pp. 302–309, Aug. 1986. 17

[31] B. S. Tsybakov and V. A. Mikhailov, "Ergodicity of a slotted ALOHA system," *Probl. Pered. Inform.*, vol. 15, pp. 73–87, Mar. 1979. 18, 19

[32] R. Rao and A. Ephremides, "On the stability of interacting queues in a multi-access system," *IEEE Trans. Inform. Theory*, vol. 34, pp. 918–930, Sept. 1988. 18

[33] V. Anantharam, "Stability region of the finite-user slotted ALOHA protocol," *IEEE Trans. Inform. Theory*, vol. 37, pp. 535–540, May 1991. 18

[34] W. Szpankowski, "Stability conditions for some multiqueue distributed systems: Buffered random access systems," *Adv. Appl. Probab.*, vol. 26, pp. 498–515, 1994. 18

[35] W. Luo and A. Ephremides, "Stability of N interacting queues in random access systems," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1579–1587, July 1999. 18

[36] S. Adireddy and L. Tong, "Exploiting decentralized channel state information for random access," *IEEE Trans. Inform. Theory*, vol. 51, pp. 537–561, Feb. 2005. 18, 31

[37] V. Naware, G. Mergen, and L. Tong, "Stability and delay of finite-user slotted ALOHA with multipacket reception," *IEEE Trans. Inform. Theory*, vol. 51, pp. 2636–2656, July 2005. 18

[38] J. Luo and A. Ephremides, "On the throughput, capacity, and stability regions of random multiple access," *IEEE Trans. Inform. Theory*, vol. 52, pp. 2593–2607, July 2006. 18

[39] R. M. Loynes, "The stability of a queue with nonindependent inter-arrival and service times," in *Proc. Cambridge Philos. Soc.*, vol. 58, pp. 497–520, 1962. 18

[40] T. Berger, N. Mehravari, D. Towsley, and J. Wolf, "Random multiple-access communication and group testing," *IEEE Trans. Commun.*, vol. 32, pp. 769–779, July 1984. 19

[41] J. K. Wolf, "Born again group testing: Multiaccess communications," *IEEE Trans. Inform. Theory*, vol. 31, pp. 185–191, Mar. 1985. 19

[42] R. Dorfman, "The detection of defective members of large populations," *Ann. Math. Statist.*, vol. 14, pp. 436–440, Dec. 1943. 19

[43] M. Sobel and P. A. Groll, "Group testing to eliminate efficiently all defectives in a binomial sample," *Bell Syst. tech. Journal*, 1959. 19

[44] P. Ungar, "The cutoff point for group testing," *Comm. Pure Appl. Math.*, vol. 13, pp. 49–54, Feb. 1960. 19

[45] J. L. Massey, *Collision Resolution Algorithms and Random Access Communications.* New York: Springer-Verlag, G. Longo ed., 1981. 21, 26, 85

[46] B. S. Tsybakov and N. D. Vvedenskaya, "Random multiple access stack algorithm," *Probl. Pered. Inform.*, vol. 16, pp. 80–94, July–Sep 1980. 22, 25

[47] P. Mathys and P. Flajolet, "Q-ary collision resolution algorithms in random access system with free or blocked channel access," *IEEE Trans. Inform. Theory*, vol. 31, pp. 217–243, Mar. 1985. 22, 25, 26, 57

[48] R. Gallager, "Conflict resolution in random access broadcast networks," in *Proc. AFOSR Work. in Commun. Theory and App.*, pp. 74–76, Sept. 1978. 23, 24, 27, 57

[49] J. Mosely and P. A. Humblet, "A class of efficient contention resolution algorithms for multiple access," *IEEE Trans. Commun.*, vol. 33, pp. 145–151, Feb. 1985. 23, 27

[50] A. Cohen, M. Kam, and R. Conn, "Partitioning a sample using binary-type questions with ternary feedback," *IEEE Trans. Syst. Man, Cybern.*, vol. 25, pp. 1405–1408, Oct. 1995. 23, 58, 59, 60, 63, 86, 88

[51] B. S. Tsybakov and V. A. Mikhailov, "Random multiple packet access: Part and try algorithm," *Probl. Pered. Inform.*, vol. 16, pp. 65–79, Oct–Dec 1980. 24, 27, 57

[52] G. C. Polyzos and M. L. Molle, "A queuing theoretic approach to the delay analysis for the fcfs 0.487 conflict resolution algorithm," *IEEE Trans. Inform. Theory*, vol. 51, pp. 2636–2656, July 2005. 24

[53] G. Ruget, *Some Tools for the Study of Channel-Sharing Algorithms.* Udine, Italy: Springer-Verlag, G. Longo ed., 1981. 27

[54] D. N. C. Tse and S. V. Hanly, "Multiaccess fading channels: Part i: Polymatroid structure, optimal resource allocation and throughput capacities," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2796–2785, Nov. 1998. 31

[55] S. V. Hanly and D. N. C. Tse, "Multiaccess fading channels: Part ii: Delay limited capacities," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2816–2831, Nov. 1998. 31

[56] X. Liu, E. Chong, and N. Shroff, "Opportunistic transmission scheduling with resource sharing constraints in wireless networks," *IEEE Trans. Sel. Areas Commun.*, vol. 19, pp. 2053–2064, Oct. 2001. 31, 43

[57] M. Grossglauser and D. N. C. Tse, "Mobility increases the capacity of ad-hoc wireless networks," *IEEE/AMC Trans. Network.*, vol. 10, Aug. 2002. 31, 43

[58] P. Bender, P. Black, M. Grob, N. Sindhushayana, and A. Viterbi, "CDMA/HDR: a bandwidth efficient high speed wireless data service for nomadic users," *IEEE Commun. Mag.*, vol. 38, pp. 70–78, July 2000. 31

[59] W. Crowther, R. Rettberg, D. Walden, S. Ornstein, and F. Heart, "A system for broadcast communication: Reservation-ALOHA," in *Proc. 6th Hawaii Int. Conf. Syst. Sci.*, pp. 371–374, Jan. 1973. 32

[60] S. S. Lam, "Packet broadcast networks - a performance analysis of the R-ALOHA protocol," *IEEE Trans. Comput.*, vol. 29, pp. 596–603, July 1980. 32, 35, 36

[61] S. Tasaka, "Stability and performance of the R-ALOHA packet broadcast system," *IEEE Trans. Comput.*, vol. 32, pp. 717–726, Aug. 1983. 32, 35, 36

[62] D. J. Goodman, R. A. Valenzuela, K. T. Gayliard, and B. Ramamurthi, "Packet reservation multiple access for local wireless communications," *IEEE Trans. Commun.*, vol. 37, pp. 885–890, Aug. 1989. 32

[63] D. J. Goodman and S. X. Wei, "Efficiency of packet reservation multiple access," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 170–176, Feb. 1991. 32

[64] L. H. Ozarow, S. Shamai, and A. D. Wyner, "Information theoretic considerations for cellular mobile radio," *IEEE Trans. Veh. Tech.*, vol. 43, pp. 954–964, May 1994. 33

[65] H. S. Wang and N. Moayeri, "Finite-state markov channel - a useful model for radio communication," *IEEE Trans. Veh. Technol.*, vol. 44, pp. 163–171, Feb. 1995. 34

[66] Q. Zhang and S. A. Kassam, "OSI reference model - the OSI model of architecture for open systems interconnection," *IEEE Trans. Commun.*, vol. 47, pp. 1688–1692, Nov. 1999. 34

[67] D. Kazakos, L. F. Merakos, and H. Delic, "Random multiple access algorithms using a control mini-slot," *IEEE Trans. Comput.*, vol. 46, pp. 473–476, Apr. 1997. 36

[68] G. H. Golub and C. F. v. Loan, *Matrix Computations.* Johns Hopkins University Press, 1996. 40

[69] K. Arrow, L. Pesotchinsky, and M. Sobel, "On partitioning a sample with binary-type questions in lieu of collecting observations," *Journal of the American Statistical Association*, vol. 76, pp. 402–409, June 1981. 58, 59, 86

[70] T. To, D. To, and X. Wang, "On a functional equation arising in the analysis of a single user selection algorithm," in *Proc. IEEE GLOBECOM 2011*, (Houston, Texas), Dec. 2011. 59, 60, 73, 86

[71] D. Raz, Y. Shavitt, and L. Zhang, "Distributed council election," *IEEE/AMC Trans. Network.*, vol. 12, pp. 483–492, Mar. 2004. 60

[72] W. Rudi, *Principles of Mathematical Analysis.* New York: McGraw-Hill, 1976. 71, 78, 81, 84

[73] K. Knopp, *Theory of Functions.* 1996. 84, 124, 125

[74] M. Tummala, C. C. Wai, and P. Vincent, "Distributed beamforming in wireless sensor networks," in *Proc. Asilomar Conf.*, (Pacific Grove, CA), pp. 793–797, Oct.–Nov. 2005. 95

[75] H. Ochiai, P. Mitran, V. Poor, and V. Tarokh, "Collaborative beamforming for distributed wireless ad-hoc sensor networks," *IEEE Trans. Signal Process.*, vol. 53, pp. 4110–4124, Nov. 2005. 95

[76] R. Mudumbai, G. Barriac, and U. Madhow, "On the feasibility of distributed beamforming in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 1754–1763, May 2007. 95

[77] R. Mudumbai, D. R. Brown, U. Madhow, and V. Poor, "Distributed transmit beamforming: Challenges and recent progress," *IEEE Commun. Mag.*, vol. 47, pp. 102–110, Feb. 2009. 95

[78] D. R. Brown and V. Poor, "Time-slotted round-trip carrier synchronization for distributed beamforming," *IEEE Trans. Signal Process.*, vol. 56, pp. 5630–5643, Nov. 2008. 95

[79] J. Uher, T. Wysocki, and B. Wysocki, "Review of distributed beamforming," *Journal of Telecommunications and Information Technology*, 2011. 95

[80] M. Johnson, M. Mitzenmacher, and K. Ramchandran, "Distributed beamforming with binary signaling," in *Proc. IEEE ISIT*, pp. 890–894, July 2008. 96

[81] H. Jeon, J. Choi, H. Lee, and J. Ha, "Channel-aware energy efficient transmission strategies for large wireless sensor networks," *IEEE Signal Process. Let.*, vol. 17, pp. 643–646, July 2010. 97

[82] B. C. Arnold, N. Balakrishnan, and H. N. Nagaraja, *A First Course in Order Statistics.* Classics in applied mathematics 54 ed. 106

[83] R. Madan, N. B. Mehta, A. F. Molisch, and J. Zhang, "Energy-efficient cooperative relaying over fading channels with simple relay selection," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 3013–3025, Aug. 2008. 106

[84] A. Papoulis, *Probability, Random Variables and Stochastic Processes.* McGraw-Hill, 1991. 116

[85] D. A. Levin, *Markov Chains and Mixing Times.* 117