DISSERTATION

DISTRIBUTED MEDIUM ACCESS CONTROL FOR AN ENHANCED PHYSICAL-LINK LAYER INTERFACE

Submitted by

Faeze Heydaryanfroshani

Department of Electrical and Computer Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Fall 2020

Doctoral Committee:

Advisor: Rockey Luo

Liuqing Yang
Ali Pezeshki
Haonan Wang

ABSTRACT

DISTRIBUTED MEDIUM ACCESS CONTROL FOR AN ENHANCED PHYSICAL-LINK LAYER INTERFACE

Current wireless network architecture equips data link layer with binary transmission/idling options and gives the control of choosing other communication parameters to the physical layer. Such a network architecture is inefficient in distributed wireless networks where user coordination can be infeasible or expensive in terms of overhead. To address this issue, an enhancement to the physical-link layer interface is proposed. At the physical layer, the enhanced interface is supported by a distributed channel coding theory, which equips each physical layer user with an ensemble of channel codes. The coding theory allows each transmitter to choose an arbitrary code to encode its message without sharing such a decision with the receiver. The receiver, on the other hand, should decode the messages of interest or report collision depending on whether or not a predetermined reliability threshold can be met. Fundamental limits of the system is characterized asymptotically using a "distributed channel capacity" when the codeword length can be taken to infinity, and non-asymptotically using an achievable performance bound when the codeword length is finite.

The focus of this dissertation is to support the enhanced interface at the data link layer. We assume that each link layer user can be equipped with multiple transmission options each corresponds to a coding option at the physical layer. Each user maintains a transmission probability vector whose entries specify the probability at which the user chooses the corresponding transmission options to transmit its packets. We propose a distributed medium access control (MAC) algorithm for a time-slotted multiple access system with/without enhanced physical-link layer interface to adapt the transmission probability vector of each user to a desired equilibrium that maximizes a chosen network utility. The MAC algorithm is applicable to a general channel model and to a wide range of utility functions. The MAC algorithm falls into the stochastic approximation framework

with guaranteed convergence under mild conditions. We developed design procedures to satisfy these conditions and to ensure that the system should converge to a unique equilibrium. Simulation results are provided to demonstrate fast and adaptive convergence behavior of the the MAC algorithm as well as the near optimal performance of the designed equilibrium.

We then extend the distributed MAC algorithm to support hierarchical primary-secondary user structure in a random multiple access system. The hierarchical user structure is established in the following senses. First, when the number of primary users is small, channel availability is kept above a pre-determined threshold regardless of the number of secondary users that are competing for the channel. Second, when the number of primary users is large, transmission probabilities of the secondary users are automatically driven down to zero. Such a hierarchical structure is achieved without the knowledge of the numbers of primary and secondary users and without direct information exchange among the users.

Furthermore, we also investigate distributed MAC for a multiple access system with multiple non-interfering channels. We assume that users are homogeneous but the multiple channels can be heterogeneous. In this case, forcing all users to converge to a homogeneous transmission scheme becomes suboptimal. We extend the distributed MAC algorithm to adaptively assign each user to only one channel and to ensure a balanced load across different channels. While theoretical analysis of the extended MAC algorithm is still incomplete, simulation results show that the algorithm can help users to converge to a near optimal channel assignment solution that maximizes a given network utility.

ACKNOWLEDGEMENTS

I would like to thank my supervisor Dr. Rockey Luo who patiently guided me through my research, generously spend his time to teach me how to approach a research problem and encouraged me to overcome my problems. I would also like to thank Yanru Tang who collaborated with me for some part of this dissertation, and my committee members for their feedback. I am grateful to my parents and to my siblings especially my sister Zohre, who I missed them so much and also to my husband Ali Mofidi for his support and to my son who made my journey of PhD very tough but enjoyable.

DEDICATION

I would like to dedicate this dissertation to my beloved family especially my mother and my father who always supported me, my son who opened my eyes to a new world and my husband who supported me emotionally

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

Introduction

1.1 Motivation

Due to dramatic increasing demand of wireless communication and limited availability of communication resources such as transmission energy and bandwidth, efficient access to the shared wireless channels with multiple wireless users has become a critical issue. Classical information theory and classical network theory have been investigating this problem from different perspectives. Classical information theory provides guidance to design an efficient communication system by characterizing the fundamental limits of the systems such as channel capacity [1]. The assumption in deriving limits of the system is that users should coordinate with each other in order to choose the best possible set of communication parameters to optimize a utility function. Such a communication model is called "coordinated communication". Because classical information theory is originally developed for applications with long communication messages and a small number of users, the overhead of coordination among users is assumed to be negligible.

Communication networks containing varieties of devices with a broad range of applications have been widely adopted all over the world. To support a diversified range of devices and applications, a modularized system architecture is essential. Classical network theory suggested a layered architecture such as the Open Systems Interconnection (OSI) model to achieve desired modularization in communication networks [2]. A layered architecture classifies networking functions into abstract layers with clearly defined interfaces. Consequently, communication network design and optimization can be focused on one layer or a few number of neighboring layers without worrying about whether or not the result can fit into the whole system. The main concern of classical network theory is connecting a large number of different devices to support a broad range of applications. Energy and bandwidth performance optimization is often a secondary concern.

Due to rapid growth of mobile devices and smart sensors, wireless traffics are increasingly fragmental and bursty. Coordinating a large number of users to jointly choose the best set of communication parameters can become infeasible or expensive in terms of overhead. Consequently, wireless users send a significant proportion of messages via distributed protocols, where each user chooses its communication parameters individually. Unfortunately, efficient distributed communication is neither supported by classical information theory nor by classical network theory. On one hand, classical information theory does not provide support for efficient distributed communication because the assumption of joint channel coding design does not apply to the distributed communication model. On the other hand, classical network theory does not provide effective support for efficient distributed networking. In current network architecture where each data link user is equipped with a single transmission option, data link layer can only make decisions on whether a packet should be transmitted or not. Other communication parameters should be chosen at the physical layer. In distributed networks where communication optimization cannot be completely done at the physical layer, data link layer should get involved in communication adaptation. However, a single transmission option at the data link layer significantly limited effective exploitation of advanced wireless capabilities such as power and rate adaptations.

In current MAC protocols such as 802.11 DCF, users decrease their transmission probabilities in response to a collision event [3]. However, it is well-known in classical information theory that maximum sum throughput of a multiple access channel can be achieved by allowing all users to transmit in parallel with low information rates. Therefore, it can be more efficient to decrease the communication rates of the users instead of reducing their transmission probabilities. For instance, consider a time-slotted multiple access system with K users and one receiver. Each user transmits with transmission rate r and transmission power P. Assume multiple access channel with additive white Gaussian noise of power N_0. Assume that K users have packets to transmit. If each user chooses its rate to be slightly less than $r = \frac{1}{2}\log_2(1 + \frac{P}{N_0})$, then the channel can only support one user to transmit at a time. In this case, the maximum system throughput is upper-bounded by $r = \frac{1}{2}\log_2(1 + \frac{P}{N_0})$. However, if users can adapt their communication rates to slightly below

$r = \frac{1}{2K}\log_2(1+\frac{KP}{N_0})$, then all users can transmit in parallel with reliable message recovery. In this case sum throughput of the system is close to $\frac{1}{2}\log_2(1+\frac{KP}{N_0})$, which can be significantly higher than $r = \frac{1}{2}\log_2(1+\frac{P}{N_0})$.

1.2 The Enhanced Physical-Link Layer Interface

To enable advanced communication adaptation at the data link layer, we propose an enhanced physical-link layer interface in [4]. The enhanced interface equips a data link layer user with multiple transmission options, each of them represents a particular combination of values of communication parameters such as communication rate, transmission power and antenna beam. Advanced communication adaptation is therefore supported at the link layer by allowing each user to navigate through the provided transmission options according to the system condition.

To support the enhanced physical-link layer interface, we proposed a new channel coding theory in [5][4][6][7] for the distributed communication model at the physical layer. The coding theory equips a physical layer user with an ensemble of channel codes each of them corresponds to a transmission option at the data link layer. Each physical layer user encodes its message using a code possibly chosen according to a data link layer decision and transmits the codeword through the channel. To preserve modularity of the network architecture, we assume that a data link layer user is constrained by the provided transmission options, and a physical layer user, on the other hand, is not in control of the coding choices. Although the receiver knows the code ensemble of each user, coding choices of the users are not shared with the receiver or with the other users. The receiver decodes the received packets if a pre-determined error probability threshold can be met, and reports collision otherwise. Let coding choices of all users be listed in a vector, termed the "coding vector". An achievable region in the space of coding vectors is defined such that, asymptotically as codeword length is taken to infinity, message recovery is guaranteed for coding vectors inside the region and collision report is guaranteed for coding vectors outside the region. A "distributed channel capacity" is defined as the maximum achievable region which coincides with the Shannon capacity region without a convex hull, as explained in [8]. The new capacity notion is enabled by an extended definition of "communication error" from its classical meaning of erroneous decoding to a generalized meaning of failure in reporting the expected outcome and with packet collision added as an expected outcome under certain circumstances. A bound on error performance is derived for the case of a finite codeword length in [6].

1.3 Contribution

Due to lack of coordination among users, packet collision is a natural feature of distributed networks. Current MAC protocols use various mechanisms to resolve collision among users equipped with a single transmission option. For instance, Aloha protocols [9] require each user to transmit its packets randomly with a pre-determined probability. In tree splitting algorithms, after each collision event, users are split into two sets, with the first set consisting of users involved in the collision and with the second set consisting of users not involved in the collision. Users in the first set will be further split into subsets and only users in one subset should transmit till the collision is resolved [10][11]. In Back-off algorithms such as the 802.11 DCF protocol, each user with an available packet randomly transmits its packet according to a probability parameter [3][12] [13]. Users decrease their probability parameters in response to packet collision and increase them in response to successful transmission. With the enhanced interface equipping each user with multiple transmission options, the question is how the system should respond to successful transmission and packet collision. More specifically, how should a distributed medium access control algorithm be developed to adapt the transmission schemes of the users with a set of limited and often non-ideal options to maximize a utility function?

In Chapter 2, we review the basic results of the distributed channel coding theory proposed in [5][4][6], which supports the enhanced physical-link layer interface at the physical layer. A new definition of "communication error" is explained, and the performance limitations of the system in terms of asymptotic achievable regions are derived with this definition. In Chapter 3, we propose a distributed MAC framework with/without enhanced physical-link layer interface [14] in a multiple access network. The distributed MAC algorithm adapts the transmission schemes of the users according to the feedback received from the receiver to maximize a network utility function. Convergence conditions are characterized for the proposed algorithm to converge to a designed unique equilibrium, which should be close to optimal with respect to the chosen utility. In Chapter 4, we extend the proposed MAC algorithm to support a hierarchical primary-secondary user structure in a random multiple access network. With proposed algorithm, we show that hierarchical structure can be established without knowledge of number of primary and secondary users and without direct exchange of information between users. In Chapter 5, we propose an extended distributed MAC algorithm for a distributed multiple access network with multiple non-interfering channels to lead the system toward the balanced channel loads. While the convergence proof of this algorithm is not completed yet, we use simulation result to show its performance.

Chapter 2

Distributed Channel Coding

In this chapter, we review the basic results of distributed channel coding theory proposed in [5][4]. In a distributed network where users are not fully coordinated due to excessive overhead, distributed channel coding equips each physical layer user with an ensemble of channel codes. Each channel code corresponds to a particular combination of communication parameters such as transmission power and communication rate. When a message becomes available at a user, the user individually chooses one code to encode its message according to a link layer decision and then sends the codeword through the channel. Code ensembles, but not coding choices, of the users are assumed to be known at the receiver. Because users are not fully coordinated, collision may happen and reliable message decoding may not always be possible. It is the receiver's task to detect whether reliable message decoding can be achieved. Then the receiver should decode the messages of interest or report collision to the data link layer. The definition of communication error is extended from the classical meaning of erroneous message decoding to the new meaning of generating an outcome that is different from the expected one (including both correct message decoding and collision report depending on the coding choices of the users). With this extended definition, performance limitations in terms of achievable regions are derived in the asymptotic case when codeword length is taken to infinity.

2.1 Distributed Multiple Access Communication

Consider a time-slotted multiple access system with K transmitters, one receiver and a discrete-time memoryless channel. The duration of each time-slot equals the length of N symbols. We use a bold font variable to represent a vector whose entries contain the corresponding variables of all users. The channel is modeled by a conditional distribution $P_{Y|\boldsymbol{X}}$, where $\boldsymbol{X} = [X_1, \ldots, X_K]^T \in \boldsymbol{\mathcal{X}}$ is the channel input symbol vector of all users with $\boldsymbol{\mathcal{X}}$ being the vector of input alphabets and $Y \in \mathcal{Y}$ is the channel output symbol with $\mathcal{Y}$ being the output alphabet. We assume that channel input alphabet $\mathcal{X}_k$ is known to user k for $k = 1, \ldots, K$ and channel distribution function $P_{Y|\boldsymbol{X}}$ is known to the receiver. Whether the channel is known to the transmitters or not does not affect the results presented in this chapter.

Each physical layer transmitter is equipped with an ensemble of M random block codes, with each code corresponding to a transmission option at the data link layer. Let $\boldsymbol{\mathcal{G}}^{(N)} = [\mathcal{G}_1^{(N)}, \ldots, \mathcal{G}_K^{(N)}]^T$ be the vector of code ensembles of the users with $\mathcal{G}_k^{(N)}$ being the code ensemble of user k for $k = 1, \ldots, K$. Let $\boldsymbol{g}$ be a particular coding choices of the users. We say $\boldsymbol{g} \in \boldsymbol{\mathcal{G}}^{(N)}$ if its entry g_k satisfies $g_k \in \mathcal{G}_k^{(N)}$ for all k. The random block coding schemes of the users are described as follows. For $g_k \in \mathcal{G}_k^{(N)}$, let $\mathcal{L} = \{\mathcal{C}_{g_k\theta_k} : \theta_k \in \Theta_k^{(N)}\}$ be a library of codebooks for user k, indexed by a set $\Theta_k^{(N)}$. Each codebook contains $\lfloor e^{Nr_{g_k}} \rfloor$ codewords, where r_{g_k} is a pre-determined parameter called "communication rate" of the corresponding code. The jth symbol of the codeword corresponding to message ω_k in codebook $\mathcal{C}_{g_k\theta_k}$ is denoted by $[\mathcal{C}_{g_k\theta_k}(\omega_k)]_j$. At the beginning of each time-slot, each transmitter, say transmitter k, randomly generates a codebook index θ_k according to a distribution $\gamma_k^{(N)}$. The distribution $\gamma_k^{(N)}$ and codebooks $\mathcal{C}_{g_k\theta_k}$ should be chosen such that random variables $X_{g_k\omega_k j} : \theta_k \to [\mathcal{C}_{g_k\theta_k}(\omega_k)]_j$, for each j, ω_k and g_k are i.i.d according to an input distribution P_{g_kX}. Note that a random block code g_k described above is characterized by its communication rate r_{g_k} and its input distribution P_{g_kX}. With an abuse of the notation, we also regard $g_k = (r_{g_k}, P_{g_kX})$ as a variable representing a rate and distribution pair. We define "code space" as a space of the $\boldsymbol{g}$ vectors, which is also a space of the rate and distribution pairs. For any code ensemble vector $\boldsymbol{\mathcal{G}}^{(N)}$, we use $\boldsymbol{\mathcal{G}}$ to represent the corresponding vector in the code space.

At the beginning of each time-slot, a random codebook is generated for each user and for each code. We assume that the receiver knows the randomly generated codebooks, and this can be achieved by sharing the codebook generation algorithm with the receiver. Note that such algorithm sharing can be done offline and therefore does not require much online information exchanged with the receiver. Each physical user chooses one code from the ensemble to encode its message. Let $\boldsymbol{g}$ denote the coding choices of the users. Because $\boldsymbol{g}$ is determined by the data link layer, we regard it as "arbitrary" at the physical layer, and should be unknown to the receiver. Given $\boldsymbol{g}$, users encode a message vector $\boldsymbol{\omega}$ into an N-length sequence of input symbol vectors, denoted by $\boldsymbol{X}_{\boldsymbol{g}}^{(N)}(\boldsymbol{\omega})$. The codewords are then sent through the multiple access channel. Upon receiving the sequence of channel output sequences $\boldsymbol{Y}^{(N)} = [Y_1, \dots, Y_N]$, the receiver calculates an estimated message and code index vector pair $(\hat{\boldsymbol{\omega}}, \hat{\boldsymbol{g}})$ if a pre-determined error probability threshold can be met and reports collision otherwise.

We assume that the receiver should choose an "operation region" $\boldsymbol{\mathcal{R}}$ in the code space. The receiver intends to decode the received message if $\boldsymbol{g} \in \boldsymbol{\mathcal{R}}$, and to report collision if $\boldsymbol{g} \notin \boldsymbol{\mathcal{R}}$. Note that the receiver needs to make decoding and collision report decisions without knowing $\boldsymbol{g}$. Given operation region $\boldsymbol{\mathcal{R}}$, communication error probability conditioned on $\boldsymbol{g}$ being the actual code index vector for codeword length N, denoted by $P_e^{(N)}(\boldsymbol{g})$, is defined as follows,

$$P_e^{(N)}(\boldsymbol{g}) = \begin{cases} \max_{\boldsymbol{w}} Pr\{(\hat{\boldsymbol{w}}, \hat{\boldsymbol{g}}) \neq (\boldsymbol{w}, \boldsymbol{g}) | (\boldsymbol{w}, \boldsymbol{g})\}, & \forall \boldsymbol{g} \in \boldsymbol{R} \\ \max_{\boldsymbol{w}} 1 - Pr \left\{ \begin{array}{c} \text{"collision" or} \\ (\hat{\boldsymbol{w}}, \hat{\boldsymbol{g}}) = (\boldsymbol{w}, \boldsymbol{g}) \end{array} \middle| (\boldsymbol{w}, \boldsymbol{g}) \right\}, & \forall \boldsymbol{g} \notin \boldsymbol{R} \end{cases}. \tag{2.1}$$

Note that $P_e^{(N)}(\boldsymbol{g})$ is a function of $\boldsymbol{g}$.

Definition 1. *An operation region $\boldsymbol{\mathcal{R}}$ is said to be achievable if for every M and $\boldsymbol{\mathcal{G}}$, there exist decoding algorithms for the sequence of coding ensembles $\boldsymbol{\mathcal{G}}^{(N)} = \boldsymbol{\mathcal{G}}$ to satisfy*

$$\lim_{N \to \infty} P_e^{(N)}(\boldsymbol{g}) = 0, \qquad \forall \boldsymbol{g} \in \boldsymbol{\mathcal{G}}. \tag{2.2}$$

The following theorem is implied by the achievability definition.

Theorem 1. *For discrete memoryless multiple access channel $P_{Y|\boldsymbol{X}}$ with finite input and output alphabets, any subset $\tilde{\boldsymbol{\mathcal{R}}}$ of an operation region $\boldsymbol{\mathcal{R}}$, i.e. $\tilde{\boldsymbol{\mathcal{R}}} \subset \boldsymbol{\mathcal{R}}$ is also achievable.*

The next theorem characterizes the maximum achievable region.

Theorem 2. *For discrete memoryless multiple access channel $P_{Y|\boldsymbol{X}}$ with finite input and output alphabets, the region $\boldsymbol{C}_d$ in coding space specified in (2.3) is asymptotically achievable*

$$\boldsymbol{C}_d = \left\{ \boldsymbol{g} \left| \boldsymbol{g} = (\boldsymbol{r}_{\boldsymbol{g}}, \boldsymbol{P}_{\boldsymbol{g}\boldsymbol{X}}), \forall S \subseteq \{1, \ldots, K\}, \sum_{k \in S} r_{g_k} < I_{\boldsymbol{g}}(\boldsymbol{X}_S; Y | \boldsymbol{X}_{\bar{S}}) \right. \right\}, \tag{2.3}$$

where $\boldsymbol{X}_S$ *denotes channel input symbols of users in* S*,* $\boldsymbol{X}_{\bar{S}}$ *denotes channel input symbols of users not in* S*, and* $I_{\boldsymbol{g}}(\boldsymbol{X}_S; Y|\boldsymbol{X}_{\bar{S}})$ *denotes mutual information between* $\boldsymbol{X}_S$ *and* Y *given* $\boldsymbol{X}_{\bar{S}}$ *with respect to joint distribution* $P_{\boldsymbol{X}Y} = P_{Y|\boldsymbol{X}} \prod_{k=1}^{K} P_{g_k X_k}$*.*

Let $\boldsymbol{C}_d^c$ be the closure of $\boldsymbol{C}_d$. $\boldsymbol{C}_d$ is the maximum achievable region in the sense that for any achievable region $\boldsymbol{\mathcal{R}}$, we must have $\boldsymbol{\mathcal{R}} \subseteq \boldsymbol{C}_d^c$.

The proof of theorem 2 can be found in [5][4].

We define $\boldsymbol{C}_d$ as "distributed capacity" of multiple access channel $P_{Y|\boldsymbol{X}}$. $\boldsymbol{C}_d$ coincides with Shannon capacity of the same channel [15] in the following sense

$$\boldsymbol{C}^c = \text{convex hull}\left(\{\boldsymbol{r} | \exists \boldsymbol{g} \in \boldsymbol{C}_d^c, \boldsymbol{r}_{\boldsymbol{g}} = \boldsymbol{r}\}\right), \tag{2.4}$$

where $\boldsymbol{C}^c$ is the closure of $\boldsymbol{C}$.

Error performance bounds in the case of finite codeword length were obtained in [6]. The corresponding results are skipped here because it is not directly relevant to the focus of this dissertation.

Chapter 3

Distributed MAC Algorithm

Classical network architecture such as the OSI model assumes that each link layer user should be equipped with a single transmission option plus an idling option [16]. At any moment, a link layer user can only choose to idle or to transmit a packet. When communication cannot be fully optimized at the physical layer, which happens often in a distributed wireless network, data link layer must share the responsibility of transmission adaptation. However, the single transmission option setting significantly limited the capability of exploiting advanced wireless tools such as rate and power adaptations at the data link layer.

The new channel coding theory for distributed communication proposed in [5][6][4][7] provided the basic physical layer support for an enhancement to the physical-link layer interface [4][7], which allows each link layer user to be equipped with multiple transmission options. These options correspond to different codes at the physical layer, possibly representing different communication settings such as different transmission power and rate combinations. The interface enhancement enables data link layer protocols to exploit advanced wireless communication adaptations through the navigation of different transmission options. This is a much needed capability for mitigating architectural inefficiency at the bottom two layers of many wireless networks. However, to maintain a layered network architecture (or system modularity), a link layer user is constrained to the provided options for transmission adaptation. How should a user efficiently exploit the often limited options to optimize a network utility, is a key question that needs to be answered.

Distributed medium access control (MAC) protocols can be categorized into non-adaptive ALOHA protocols [9], splitting algorithms [10], and back-off approaches [12][13][3]. ALOHA protocols have been widely used to investigate fundamental network properties, such as achievable throughput and stability regions [17]. In splitting algorithms such as the FCFS algorithm [10], each user maintains a common virtual interval and a randomly generated identity value belonging to the interval. Users partition the interval and order the sub-intervals based upon a sequence of channel feedback messages. Transmission schedule of the users are determined accordingly. While splitting algorithms can often achieve a relatively high system throughput, their correct function depends on the assumptions of instant availability of noiseless channel feedback and correct reception of feedback sequence. Both of these conditions, unfortunately, can be violated in a wireless environment. Theoretical analysis of a splitting algorithm can be extremely challenging, especially when wireless-related factors such as channel noise, feedback error, and transmission delay are taken into account. Back-off algorithms, on the other hand, has proven to enjoy more trackable analysis [12][13][3]. In back-off algorithms such as the 802.11 DCF protocol, depending on packet availability, each user transmits its packets randomly according to an associated probability parameter. A user should decrease its transmission probability in response to a packet collision (or transmission failure) event, and increase its transmission probability in response to a transmission success event. Distributed probability adaptation in a back-off algorithm often falls into the framework of stochastic approximation algorithms [12][13], with rigorously developed mathematical and statistical tools available for its performance analysis. It is well known that convergence proof of these algorithms often hold in the existence of measurement noise and feedback delay [18]. Practical back-off algorithms can also be analyzed using Markov models to characterize the impact of discrete probability updates [3].

In [13], a stochastic approximation model was proposed for distributed networking over a collision channel with an unknown finite number of users, each having a saturated message queue. By targeting the transmission probability of each user as a function of a locally measurable system variable, such as the channel idling probability, it was shown that the system can be designed to converge to a unique stable equilibrium. In the case of throughput maximization with homogeneous users, it was proposed that idling probability of the channel should be controlled toward the asymptotically optimal value of $1/e$. This is similar to the proposal of controlling the total traffic level toward 1, as discussed in [12] using a stochastic approximation framework for a system with an infinite number of users. Most of the existing analysis of the splitting and the back-off algorithms either assumes a throughput optimization objective and/or a simple collision channel model.

While significant research efforts have been made to revise collision resolution algorithms to incorporate wireless-related physical layer properties, such as capture effect [19] and multi-packet reception [20], not much progress has been reported since the 1980s on integrating these extensions with the insightful stochastic approximation-based frameworks, such as those introduced in [12][13].

With the enhanced physical-link layer interface, a link layer user can be equipped with multiple transmission options. Link layer networking can face a set of channel models that is much richer and more complicated than the classical collision channel [21][22]. It is not immediately clear how collision resolution should be done in such a scenario. For example, if a user can adapt its transmission power and rate in addition to its transmission probability, what does "back-off" even mean in this case? Motivated by this and similar simple questions, in this chapter, we investigate the problem of distributed MAC in a wireless multiple access network with/without the enhanced physical-link layer interface. To maintain a relatively simple and trackable investigation, we assume that the network should have an unknown finite number of homogeneous users (transmitters), each being backlogged with a saturated message queue. Other than the user homogeneity assumption, our choice of problem formulation and analytical tools are similar to those presented in [13] for the collision channel. First, the assumption of saturated message queues is introduced to avoid the complication of random message arrivals. While bursty message arrival is rather an important character of distributed network systems [16][23], it is known to create coupling between transmission activities of the users. Such coupling often leads to open research problems in throughput and stability analysis of systems with a relatively small number of users [24][17]. Results obtained with the assumption of saturated message queues can often serve as achievable bounds to the corresponding results for systems with random message arrivals. Second, because each user only interacts with the receiver, the assumption of multiple access networking with homogeneous users mainly represents the communication environment envisioned by each link layer user. In other words, without further knowledge about the actual networking environment, a link layer protocol should be designed to help a user to get a fair share of the multiple access channel under the assumption of user homogeneity[1]. Furthermore, we consider extending the system model to the case of heterogeneous users in Chapter 4. Third, because users in a distributed network often access the channel opportunistically, it may not be easy to know how many users are actually active. We assume that each user should be able to calculate its optimal transmission scheme if the user number is known, but we would like to develop distributed algorithms to lead the system to a close-to-optimal operation point without the knowledge of the actual user number. The expectation is that, if fast adaptation algorithms can be developed accordingly, a system can keep track of the number of active users even if users frequently join/exit the communication party[2].

The rest of this chapter is organized as follows. In Section 3.1, we present a stochastic approximation framework for a class of distributed MAC algorithms with guaranteed convergence to a unique system equilibrium. While the results are more or less standard in the stochastic approximation literature, they characterize the key conditions for convergence and a key approach to simplify the equilibrium analysis. In Section 3.2, based on a general link layer channel model and a utility maximization objective, we present a distributed MAC algorithm that adapts the transmission scheme of each user according to two carefully designed functions. We show that, under a set of assumptions, the proposed MAC algorithm should lead the transmission schemes of all users to a designed system equilibrium. Next, in Section 3.3, we consider a simple scenario and present a closed-form approach to design the two key functions to satisfy the required assumptions and to place the system equilibrium at a point that is close to optimal with respect to a chosen symmetric network utility[3]. We then extend the design approach to the general scenario in Section 3.4 where a search-assisted approach is proposed to replace the closed-form approach to design part of the two key functions. Simulation results are provided in Section 5.2 to demonstrate both the optimality and the convergence properties of the proposed MAC algorithm.

---

[1]Note that user symmetry is widely assumed in many channel models such as the collision channel [16] and the multi-packet reception channel [20].

[2]This includes the case when users do not have saturated message queues.

[3]A network utility is “symmetric” if it requires that utility values achieved by different individual users should be equal.

3.1 A Stochastic Approximation Framework

Consider a distributed multiple access network with a memoryless channel and K homogeneous users (transmitters). Time is slotted. The length of each time slot equals the transmission duration of one packet. We assume that the number of users K should be unknown to the users and also unknown to the receiver. Each user is equipped with M transmission options plus an idling option, and is backlogged with a saturated message queue. We formulate the problem at the data link layer in the sense of constraining users to the provided transmission options. At the beginning of each time slot t, a user should either idle or randomly choose a transmission option to send a message, with corresponding probabilities being specified by an associated probability vector. Transmission decisions of the users are made individually, and they are shared neither among the users nor with the receiver. The M-length probability vector associated to user k, $k = 1, \ldots, K$, is denoted by $\boldsymbol{p}_k(t)$ for time slot t. We write $\boldsymbol{p}_k(t) = p_k(t)\boldsymbol{d}_k(t)$, with $0 \le p_k(t) \le 1$ being the probability that user k transmits a packet in time slot t, and with vector $\boldsymbol{d}_k(t)$ specifying the conditional probabilities for user k to choose each of the transmission options should it decide to transmit a packet. Entries of the $\boldsymbol{d}_k(t)$ vector satisfy $0 \le d_{km}(t) \le 1$ for $1 \le m \le M$, and $\sum_{m=1}^{M} d_{km}(t) = 1$. We term $p_k(t)$ the "transmission probability" of user k, and term $\boldsymbol{d}_k(t)$ the "transmission direction" vector of user k.

At the end of each time slot t, based upon available channel feedback, each user k derives a target probability vector $\tilde{\boldsymbol{p}}_k(t)$. User k then updates its transmission probability vector by

$$\boldsymbol{p}_k(t+1) = (1-\alpha(t))\boldsymbol{p}_k(t) + \alpha(t)\tilde{\boldsymbol{p}}_k(t) = \boldsymbol{p}_k(t) + \alpha(t)(\tilde{\boldsymbol{p}}_k(t) - \boldsymbol{p}_k(t)), \tag{3.1}$$

where $\alpha(t) > 0$ is a step size parameter of time slot t. Let $\boldsymbol{P}(t) = [\boldsymbol{p}_1^T(t), \boldsymbol{p}_2^T(t), \ldots, \boldsymbol{p}_K^T(t)]^T$ denote an MK-length vector that consists of the transmission probability vectors of all users in time slot t. Let $\tilde{\boldsymbol{P}}(t) = [\tilde{\boldsymbol{p}}_1^T(t), \tilde{\boldsymbol{p}}_2^T(t), \ldots, \tilde{\boldsymbol{p}}_K^T(t)]^T$ denote the corresponding target vector. According to (3.1), $\boldsymbol{P}(t)$ is updated by

$$\boldsymbol{P}(t+1) = \boldsymbol{P}(t) + \alpha(t)(\tilde{\boldsymbol{P}}(t) - \boldsymbol{P}(t)). \tag{3.2}$$

Probability adaptation given in (3.2) falls into the stochastic approximation framework [18][25][26], where the target probability vector $\tilde{\boldsymbol{P}}(t)$ is often calculated from noisy estimates of certain system variables, e.g., the channel idling probability.

Define $\hat{\boldsymbol{P}}(t) = [\hat{\boldsymbol{p}}_1^T(t), \hat{\boldsymbol{p}}_2^T(t), \ldots, \hat{\boldsymbol{p}}_K^T(t)]^T$ as the "theoretical value" of $\tilde{\boldsymbol{P}}(t)$ under the assumption that there is no measurement noise and no feedback error in time slot t. Let $E_t[\tilde{\boldsymbol{P}}(t)]$ be the conditional expectation of $\tilde{\boldsymbol{P}}(t)$ given system state at the beginning of time slot t. The difference between $E_t[\tilde{\boldsymbol{P}}(t)]$ and $\hat{\boldsymbol{P}}(t)$ is defined as the bias in the target probability vector calculation, denoted by $\boldsymbol{G}(t)$.

$$\boldsymbol{G}(t) = E_t[\tilde{\boldsymbol{P}}(t)] - \hat{\boldsymbol{P}}(t). \tag{3.3}$$

We assume that, given the communication channel, both $\hat{\boldsymbol{P}}(t) = \hat{\boldsymbol{P}}(\boldsymbol{P}(t))$ and $\boldsymbol{G}(t) = \boldsymbol{G}(\boldsymbol{P}(t))$ should only be functions of $\boldsymbol{P}(t)$, which is the transmission probability vector in time slot t.

The following two conditions are typically required for the convergence of a stochastic approximation algorithm [18][25][26].

Condition 1. *(Mean and Bias) There exists a constant $K_m > 0$ and a bounding sequence $0 \leq \beta(t) \leq 1$, such that*

$$\|\boldsymbol{G}(\boldsymbol{P}(t))\| \leq K_m \beta(t), \tag{3.4}$$

where $\|.\|$ denotes the second order norm. We assume that $\beta(t)$ is controllable in the sense that one can design protocols to ensure $\beta(t) \leq \epsilon$ for any chosen $\epsilon > 0$ and for large enough t.

Condition 2. *(Lipschitz Continuity) There exists a constant $K_l > 0$, such that*

$$\|\hat{\boldsymbol{P}}(\boldsymbol{P}_a) - \hat{\boldsymbol{P}}(\boldsymbol{P}_b)\| \leq K_l \|\boldsymbol{P}_a - \boldsymbol{P}_b\|, \textit{ for all } \boldsymbol{P}_a, \boldsymbol{P}_b. \tag{3.5}$$

According to stochastic approximation theory [18][26], if the above two conditions are satisfied, the step size sequence $\alpha(t)$ and the bounding sequence $\beta(t)$ are small enough, then trajectory of the transmission probability vector $\boldsymbol{P}(t)$ under distributed adaptation given in (3.2) can be approximated by the following associated ordinary differential equation (ODE) in a sense explained in [18][26],

$$\frac{d\boldsymbol{P}(t)}{dt} = -[\boldsymbol{P}(t) - \hat{\boldsymbol{P}}(t)], \tag{3.6}$$

where we used t to denote the continuous time variable. Because all entries of $\boldsymbol{P}(t)$ and $\hat{\boldsymbol{P}}(t)$ stay in the range of $[0,1]$, any equilibrium $\boldsymbol{P}^* = [\boldsymbol{p}_1^{*T}, \ldots, \boldsymbol{p}_K^{*T}]^T$ of the associated ODE must satisfy

$$\boldsymbol{P}^* = \hat{\boldsymbol{P}}(\boldsymbol{P}^*). \tag{3.7}$$

Suppose that the associated ODE given in (3.6) has a unique solution at $\boldsymbol{P}^*$, then the following convergence results can be obtained from the standard conclusions in the stochastic approximation literature.

Theorem 3. *For distributed transmission probability adaptation given in (3.2), assume that the associated ODE given in (3.6) has a unique stable equilibrium at $\boldsymbol{P}^*$. Suppose that $\alpha(t)$ and $\beta(t)$ satisfy the following conditions*

$$\sum_{t=0}^{\infty} \alpha(t) = \infty, \sum_{t=0}^{\infty} \alpha(t)^2 < \infty, \sum_{t=0}^{\infty} \alpha(t)\beta(t) < \infty. \tag{3.8}$$

Under Conditions 1 and 2, $\boldsymbol{P}(t)$ converges to $\boldsymbol{P}^*$ with probability one.

Theorem 3 is implied by [18, Theorem 4.3].

Theorem 4. *For distributed transmission probability adaptation given in (3.2), assume that the associated ODE given in (3.6) has a unique stable equilibrium at $\boldsymbol{P}^*$. Let Conditions 1 and 2 hold true. Then for any $\epsilon > 0$, there exists a constant $K_w > 0$, such that, for any $0 < \underline{\alpha} < \overline{\alpha} < 1$ satisfying the following constraint*

$$\exists T_0 \geq 0, \underline{\alpha} \leq \alpha(t) \leq \overline{\alpha}, \beta(t) \leq \sqrt{\overline{\alpha}}, \forall t \geq T_0, \tag{3.9}$$

$\boldsymbol{P}(t)$ converges weakly to $\boldsymbol{P}^*$ in the following sense

$$\limsup_{t\to\infty} Pr\left\{\|\boldsymbol{P}(t)-\boldsymbol{P}^*\| \geq \epsilon\right\} < K_w\overline{\alpha}. \tag{3.10}$$

Theorem 4 can be obtained by following the proof of [26, Theorem 2.3] with minor revisions.

For simplicity, we assumed the same step size sequence $\alpha(t)$ and the same bounding sequence $\beta(t)$ for all users. We also assumed that all users should update their transmission probability vectors synchronously in each time slot. However, according to the literature of stochastic approximation theory [18], convergence results stated in Theorems 3 and 4 should remain valid, even if different users use different step sizes and bounding sequences, so long as the step sizes and bounding sequences of all users satisfy the same constraints given in (3.8) and (3.9). Convergence results of Theorems 3 and 4 should also remain valid if users adapt their probability vectors asynchronously, so long as users update their probability vectors frequently enough [18]. Note that information on the asymptotic convergence rate of $\boldsymbol{P}(t) \to \boldsymbol{P}^*$ can be obtained from the eigenvalues of the Hessian matrix $\left.\frac{\partial(\hat{\boldsymbol{P}}(\boldsymbol{P})-\boldsymbol{P})}{\partial \boldsymbol{P}}\right|_{\boldsymbol{P}^*}$ [13]. However, convergence rate discussion is outside the scope of this dissertation.

Theorems 3 and 4 provided convergence guarantee for a class of distributed MAC algorithms. Within the presented stochastic approximation framework, the key question is how to design a distributed MAC algorithm to satisfy Conditions 1 and 2 and to place the unique equilibrium of the associated ODE at a point that maximizes a chosen utility. Because users are homogeneous, if equilibrium of the system is indeed unique, transmission probability vectors of the users at the equilibrium must be identical. We choose to enforce such a property by guaranteeing that all users should obtain the same target transmission probability vector in each time slot. This is achieved by the following design details.

We assume that, in each time slot, there is a virtual packet being transmitted through the channel. Virtual packets assumed in different time slots are identical. A virtual packet is an assumed packet whose coding parameters are known to the users and to the receiver, but it is not physically transmitted in the system, i.e., the packet is “virtual”. Without knowing the transmission/idling status of the users, we assume that the receiver can detect whether the reception of a virtual packet should be regarded as successful or not, and therefore can estimate its success probability [4][21]. For example, suppose that the link layer channel is a collision channel, and a virtual packet has the same coding parameters of a real packet. Then, virtual packet reception in a time slot should be regarded as successful if and only if no real packet is transmitted. Success probability of the virtual packet in this case equals the idling probability of the collision channel. For another example, if all packets including the virtual packet are encoded using random block codes, given the physical layer channel, reception of the virtual packet corresponds to a detection task that judges whether or not the vector transmission status of all real users should belong to a specific region. Such detection tasks and their performance bounds have been extensively investigated in the distributed channel coding literature [5][6][4][7].

Let $q_v(t)$ denote the success probability of the virtual packet in time slot t. We term $q_v(t)$ the "channel contention measure" because it is designed to serve as a measurement of the contention level of the link-layer multiple access channel. We assume that the receiver should obtain an estimate of $q_v(t)$ and feed it back to all transmitters. Note that, in the collision channel case when $q_v(t)$ equals the channel idling probability, feeding back an estimate of $q_v(t)$ may not be necessary. So long as each user k knows the success probability of its own packet, denoted by $q_k(t)$, idling probability of the channel can be calculated by $(1 - p_k(t))q_k(t)$. With a general link layer channel, however, such calculation of $q_v(t)$ at a transmitter is not always possible if an estimate of $q_v(t)$ is not fed back directly by the receiver [21]. Upon receiving an estimate of $q_v(t)$, each user calculates its target transmission probability vector as the same function of the $q_v(t)$ estimate. Denote the theoretical target transmission probability vector of a user by $\hat{\boldsymbol{p}}(q_v(t))$. The theoretical target transmission probability vector of all users is given by $\hat{\boldsymbol{P}}(t) = \mathbf{1} \otimes \hat{\boldsymbol{p}}(q_v(t))$, where $\mathbf{1}$ denotes a K-length vector of all 1's and $\otimes$ represents the Kronecker product. Consequently, according to (3.6), any equilibrium $\boldsymbol{P}^*$ of the ODE must take the form of $\boldsymbol{P}^* = \mathbf{1} \otimes \boldsymbol{p}^*$. Because q_v is a function of the transmission probability vectors of all users, we must have $\boldsymbol{P}^* = \mathbf{1} \otimes \boldsymbol{p}^* = \mathbf{1} \otimes \hat{\boldsymbol{p}}(\boldsymbol{p}^*)$, where $\hat{\boldsymbol{p}}(\boldsymbol{p}^*)$ denotes the derived target transmission probability vector of a user given that all users have the same transmission probability vector $\boldsymbol{p}^*$.

In a practical system, an estimate of $q_v(t)$ is likely to be corrupted by measurement noise. We assume that, if the transmission probability vectors of all users $\boldsymbol{P}$ is kept at a constant vector, and q_v is measured over an interval of Q time slots, then the measurement should converge to its true value with probability one as Q is taken to infinity. Other than this assumption, measurement noise is not involved in the discussion of the design objectives, i.e., to meet Conditions 1 and 2 and to place the unique system equilibrium at the desired point. Therefore, in the following three sections, we assume that $q_v(t)$ can be measured precisely and be fed back to the users. This leads to $\tilde{\boldsymbol{P}}(t) = \hat{\boldsymbol{P}}(t) = \mathbf{1} \otimes \hat{\boldsymbol{p}}(t)$. We will also skip time index t to simplify the notations.

3.2 Channel Model, Utility, and A Distributed MAC Algorithm

According to the distributed channel coding theory [5][6][4][7], given any combination of transmission status of the users, the receiver should be able to reliably detect the success/failure outcomes of the real and the virtual packets. These outcomes as functions of the transmission status of the users form the complete model of the link layer multiple access channel. While the complete channel model can be overly complicated, we require that the channel should satisfy the following sensitivity assumption.

Assumption 1. *(Channel Sensitivity) There exists a finite constant K_c, such that virtual packet reception should fail if the number of parallel real packet transmissions exceeds K_c.*

Because it is usually trivial to satisfy this assumption, in the rest of the dissertation, we will assume it should hold true.

Given the link layer multiple access channel and the number of users K, channel contention measure $q_v(\boldsymbol{P}, K)$ is a function of the transmission probability vectors of all users $\boldsymbol{P}$. Under Assumption 1, $q_v(\boldsymbol{P}, K)$ equals the summation of a finite number of terms each representing the probability of a particular transmission status combination of the users that can support the successful reception of the virtual packet[4]. Because each of these terms is a polynomial function of $\boldsymbol{P}$, we have the following property.

Theorem 5. *With Assumption 1, channel contention measure q_v is Lipschitz continuous in the transmission probability vectors of all users $\boldsymbol{P}$. That is, there exists a finite constant K_{qc}, such that for any number of users K and any transmission probability vectors $\boldsymbol{P}_a$, $\boldsymbol{P}_b$, the following inequality should hold true.*

$$|q_v(\boldsymbol{P}_a, K) - q_v(\boldsymbol{P}_b, K)| \leq K_{qc} \|\boldsymbol{P}_a - \boldsymbol{P}_b\|. \tag{3.11}$$

Proof of the theorem is skipped.

In the rest of the dissertation, we will simplify the complete link layer channel model into two sets of channel parameter functions, $\{C_{rij}(\boldsymbol{d})\}$ and $\{C_{vj}(\boldsymbol{d})\}$. Assume that all users should have the same transmission direction vector $\boldsymbol{d}$. We define $\{C_{rij}(\boldsymbol{d})\}$ for $1 \leq i \leq M$ and $j \geq 0$ as the "real channel parameter function set". $C_{rij}(\boldsymbol{d})$ is the conditional success probability of a real packet corresponding to the ith transmission option, should the packet be transmitted in parallel with j other real packets. We also define $\{C_{vj}(\boldsymbol{d})\}$ for $j \geq 0$ as the "virtual channel parameter function set". $C_{vj}(\boldsymbol{d})$ is the success probability of the virtual packet should it be transmitted in parallel with j real packets. We assume that $C_{vj}(\boldsymbol{d}) \geq C_{v(j+1)}(\boldsymbol{d})$ should hold for all $j \geq 0$ and for any $\boldsymbol{d}$. That is, with users having the same transmission direction vector $\boldsymbol{d}$, if the number of parallel real packet transmissions increases, the chance of a virtual packet getting through the channel should not increase. Let $\epsilon_v \geq 0$ be a pre-determined constant. We define $J_{\epsilon_v}(\boldsymbol{d})$ as the smallest integer such that $C_{vJ_{\epsilon_v}}(\boldsymbol{d})$ is strictly larger than $C_{v(J_{\epsilon_v}+1)}(\boldsymbol{d}) + \epsilon_v$, i.e.,

$$J_{\epsilon_v}(\boldsymbol{d}) = \arg\min_{j} C_{vj}(\boldsymbol{d}) > C_{v(j+1)}(\boldsymbol{d}) + \epsilon_v. \tag{3.12}$$

[4]For example, a particular term can represent the probability that K_0 users idle, K_1 users transmit with the 1st option, K_2 users transmit with the 2nd option, etc, under the constraints that $\sum_{i=0}^{M} K_i = K$ and $\sum_{i=1}^{M} K_i \leq K_c$.

By definition, $J_{\epsilon_v}(\boldsymbol{d})$ is a function of $\boldsymbol{d}$. Note that the value of ϵ_v needs to be carefully chosen to guarantee the existence of $J_{\epsilon_v}(\boldsymbol{d})$ for all $\boldsymbol{d}$. With Assumption 1, we should have $C_{vj}(\boldsymbol{d}) = 0$ for all $j > K_c$. If the virtual packet is designed properly, we should also have $C_{v0} > 0$, where C_{v0} is not a function of $\boldsymbol{d}$. Therefore, the existence of $J_{\epsilon_v}(\boldsymbol{d})$ is guaranteed if ϵ_v is chosen to satisfy $0 \leq \epsilon_v < \frac{C_{v0}}{K_c}$. Because both $\{C_{rij}(\boldsymbol{d})\}$ and $\{C_{vj}(\boldsymbol{d})\}$ can be derived from the physical layer channel model and the coding parameters of the packets [5][6][4][7], we assume that they should be known at the transmitters and at the receiver. Note that, while $\{C_{vj}(\boldsymbol{d})\}$ depends on the coding detail of the virtual packet, virtual packet is not involved in the calculation of $\{C_{rij}(\boldsymbol{d})\}$.

With the simplified channel model, given the number of users K and under the assumption that all users should have the same transmission probability vector $\boldsymbol{p} = p\boldsymbol{d}$, we write channel contention measure $q_v(\boldsymbol{p}, K)$ as a function of $\boldsymbol{p}$ and K. In this case, $q_v(\boldsymbol{p}, K)$ can be calculated by

$$q_v(\boldsymbol{p}, K) = \sum_{j=0}^{K} \binom{K}{j} p^j (1-p)^{K-j} C_{vj}(\boldsymbol{d}). \tag{3.13}$$

We assume that users intend to maximize a symmetric utility function. Under the assumption that all users should have the same transmission probability vector, the utility function $U(K, \boldsymbol{p}, \{C_{rij}(\boldsymbol{d})\})$ is defined as a function of the number of users K, the common transmission probability vector $\boldsymbol{p} = p\boldsymbol{d}$, and the real channel parameter function set $\{C_{rij}(\boldsymbol{d})\}$. For example, suppose that users intend to maximize the symmetric sum throughput of the network. If the ith transmission option has a communication rate of r_i (bits/time slot), then the utility function should be given by

$$U(K, \boldsymbol{p}, \{C_{rij}(\boldsymbol{d})\}) = K \sum_{i=1}^{M} d_i r_i \sum_{j=0}^{K-1} \binom{K-1}{j} p^{j+1} (1-p)^{K-1-j} C_{rij}(\boldsymbol{d}). \tag{3.14}$$

Next, we present a distributed MAC algorithm that adapts the transmission probability vectors of the users based on the estimated q_v fed back from the receiver and according to two carefully designed functions, both are functions of an estimated number of users $\hat{K}$. The first function is the "theoretical transmission probability vector" function $\boldsymbol{p}^*(\hat{K})$, which denotes the theoretical transmission probability vector of a user if the number of users equals $\hat{K}$. The second function is the "theoretical channel contention measure" function $q_v^*(\hat{K})$, which denotes the theoretical success probability of the virtual packet if the number of users of the system equals $\hat{K}$ and all users have the same transmission probability vector $\boldsymbol{p}^*(\hat{K})$.

Assumption 2. *(Estimation Continuity) $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K})$ should be defined for both integer and non-integer $\hat{K}$ values. Their limits as $\hat{K} \to \infty$, denoted by $\boldsymbol{p}^*(\infty) = \lim_{\hat{K}\to\infty} \boldsymbol{p}^*(\hat{K})$ and $q_v^*(\infty) = \lim_{\hat{K}\to\infty} q_v^*(\hat{K})$, should be well defined. For all integer-valued $\hat{K}$, the following equality should be satisfied*

$$q_v^*(\hat{K}) = q_v(\boldsymbol{p}^*(\hat{K}), \hat{K}). \tag{3.15}$$

Assumption 3. *(Contention Monotonicity) $q_v^*(\hat{K})$ should be non-increasing in $\hat{K}$. There exists a positive constant $K_{\min}$, such that $q_v^*(\hat{K})$ should be strictly decreasing for $\hat{K} > K_{\min}$, and $\boldsymbol{p}^*(\hat{K})$ should remain a constant vector for $\hat{K} \leq K_{\min}$.*

We are now ready to present the distributed MAC algorithm.

Distributed MAC Algorithm:

1. Each user initializes its transmission probability vector.

2. Let $Q > 0$ be a pre-determined integer. Over an interval of Q time slots, the receiver measures the success probability of the virtual packet, denoted by q_v, and feeds q_v back to all users.

3. Upon receiving q_v, each user derives an estimated number of users $\hat{K}$ by solving the following equation.

$$q_v^*(\hat{K}) = q_v, \qquad \text{s.t. } \hat{K} \geq K_{\min}. \tag{3.16}$$

   If a $\hat{K}$ satisfying (3.16) cannot be found, users set $\hat{K} = K_{\min}$ if $q_v > q_v^*(K_{\min})$, or set $\hat{K} = \infty$ otherwise. Each user then sets the target transmission probability vector at $\hat{\boldsymbol{p}} = \boldsymbol{p}^*(\hat{K})$.

4. Each user, say user k, updates its transmission probability vector by

$$\boldsymbol{p}_k = (1-\alpha)\boldsymbol{p}_k + \alpha\hat{\boldsymbol{p}}, \tag{3.17}$$

where α is the step size parameter for user k.

5. The process is repeated from Step 2 till transmission probability vectors of all users converge.

To prove convergence of the distributed MAC algorithm, we need two additional assumptions presented below.

Assumption 4. *(Target Continuity) Given q_v, let the target transmission probability vector $\hat{\mathbf{p}}$ be determined as in Step 3 of the distributed MAC algorithm. $\hat{\mathbf{p}}(q_v)$ as a function of q_v should be Lipschitz continuous in q_v. That is, there exists a constant K_{qp}, such that for any q_{v1} and q_{v2}, the following inequality should hold*

$$\|\hat{\boldsymbol{p}}(q_{v1}) - \hat{\boldsymbol{p}}(q_{v2})\| \leq K_{qp}|q_{v1} - q_{v2}|. \tag{3.18}$$

Assumption 5. *(Equilibrium Uniqueness) For any number of users $K > K_{\min}$, equation $q_v^*(\hat{K}) = q_v(\boldsymbol{p}^*(\hat{K}), K)$ should have a unique solution at $\hat{K} = K$. For any number of users $K \leq K_{\min}$, equation $q_v^*(\hat{K}) = q_v(\boldsymbol{p}^*(\hat{K}), K)$ should hold for all $\hat{K} \leq K_{\min}$.*

Convergence property of the proposed distributed MAC algorithm is stated in the following theorem.

Theorem 6. *Consider the K-user multiple access network presented in this section. Under Assumptions 1-5, and with the proposed MAC algorithm, the associated ODE given in (3.6) has a unique equilibrium at $\boldsymbol{P}^* = \mathbf{1} \otimes \boldsymbol{p}^*(K)$. The probability target $\hat{\mathbf{p}}(\boldsymbol{P})$ as a function of the transmission probability vectors of all users $\boldsymbol{P}$ satisfies Conditions 1 and 2. Consequently, transmission probability vectors of all users should converge to $\boldsymbol{P}^* = \mathbf{1} \otimes \boldsymbol{p}^*(K)$ in the sense specified in Theorems 3 and 4.*

Proof of Theorem 6 is given in Appendix A.1.

Note that the distributed MAC algorithm guides the adaptation of transmission probability vectors of all users by trying to maintain channel contention measure at an appropriate level. System equilibrium can be designed as a function of the number of users K even though the actual value of K is unknown. While we have not yet provided any optimality argument on how $\boldsymbol{p}^*(\hat{K})$ should be designed to maximize a chosen utility $U(K, \boldsymbol{p}, \{C_{rij}(\boldsymbol{d})\})$, because $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K})$ functions need to satisfy the required assumptions, it is quite clear that system equilibrium cannot be designed freely.

3.3 A Closed-Form Design Approach with Pre-fixed Transmission Direction

In this section, we consider a simple scenario when all users have the same pre-fixed transmission direction vector $\boldsymbol{d}$. We write the theoretical transmission probability vector function $\boldsymbol{p}^*(\hat{K})$ as

$$\boldsymbol{p}^*(\hat{K}) = p^*(\hat{K})\boldsymbol{d}, \tag{3.19}$$

where $p^*(\hat{K})$ is the "theoretical transmission probability" function that needs to be designed. It is easy to see that the problem becomes equivalent to the case when each user only has a single transmission option, as investigated in [21][22]. We will review the closed-form approach presented in [21][22] to design $p^*(\hat{K})$ and $q_v^*(\hat{K})$ functions to maximize the chosen network utility and to satisfy Assumptions 2-5. Most of the design parameters presented in this section should be functions of $\boldsymbol{d}$. However, for the sake of simple presentation, we will skip $\boldsymbol{d}$ in some of the notations.

With a fixed transmission direction vector $\boldsymbol{d}$, for most of the utility functions of interest, such as the sum throughput function given in (3.14), an asymptotically optimal solution should keep the expected load of the channel at a constant [13][20]. Therefore, if p_K^* is the optimal transmission probability for user number K, we should have $\lim_{K\to\infty} Kp_K^* = x^*$ with $x^* > 0$ being obtained by the following asymptotic utility optimization.

$$x^* = \arg\max_{x} \lim_{K\to\infty} U\left(K, \frac{x}{K}\boldsymbol{d}, \{C_{rj}(\boldsymbol{d})\}\right). \tag{3.20}$$

Without knowing the actual number of users K, we design $p^*(\hat{K})$ as

$$p^*(\hat{K}) = \min\left\{p_{\max}, \frac{x^*}{\hat{K}+b}\right\}, \tag{3.21}$$

where $b \geq 1$ is a pre-determined design parameter, and $p_{\max}$ is given by

$$p_{\max} = \min\left\{1, \frac{x^*}{J_{\epsilon_v}(\boldsymbol{d})+b}\right\}, \tag{3.22}$$

with $J_{\epsilon_v}(\boldsymbol{d})$ being defined in (3.12). According to Theorem 6, such a design implies that we intend to set system equilibrium at $\boldsymbol{P}^* = \mathbf{1} \otimes p^*(K)\boldsymbol{d}$. As shown in [21][22], this equilibrium setting is not only asymptotically optimal as $K \to \infty$, but also often close to optimal for small K values.

The $q_v^*(\hat{K})$ function, on the other hand, can be calculated as $q_v^*(\hat{K}) = q_v(p^*(\hat{K})\boldsymbol{d}, \hat{K})$ for integer-valued $\hat{K}$. For non-integer-valued $\hat{K}$, $q_v^*(\hat{K})$ is designed as follows. Let $N = \lfloor \hat{K} \rfloor$ be the largest integer below $\hat{K}$. Define $q_N(\boldsymbol{p})$ and $q_{N+1}(\boldsymbol{p})$ as

$$q_N(\boldsymbol{p}) = q_v(\boldsymbol{p}, N), \quad q_{N+1}(\boldsymbol{p}) = q_v(\boldsymbol{p}, N+1). \tag{3.23}$$

$q_v^*(\hat{K})$ is designed as a linear interpretation between $q_N(p^*(\hat{K})\boldsymbol{d})$ and $q_{N+1}(p^*(\hat{K})\boldsymbol{d})$.

$$q_v^*(\hat{K}) = \frac{p^*(\hat{K}) - p^*(N+1)}{p^*(N) - p^*(N+1)} q_N(p^*(\hat{K})\boldsymbol{d}) + \frac{p^*(N) - p^*(\hat{K})}{p^*(N) - p^*(N+1)} q_{N+1}(p^*(\hat{K})\boldsymbol{d}). \tag{3.24}$$

With $p^*(\hat{K})$ and $q_v^*(\hat{K})$ functions designed in (3.21) and (3.24), respectively, Assumption 2 is satisfied. According to the following theorem, Assumption 3 should hold true if design parameter b in (3.21) is chosen appropriately.

Theorem 7. *[21, Theorem 4] Let $x^* > 0$ and $b \geq \max\{1, x^* - \gamma_{\epsilon_v}\}$ with γ_{ϵ_v} being defined as*

$$\gamma_{\epsilon_v} = \min_{\hat{N}, \hat{N} \geq J_{\epsilon_v}(\boldsymbol{d}), \hat{N} \geq x^* - b} \frac{\sum_{j=0}^{\hat{N}} j \binom{\hat{N}}{j} \left(\frac{p^*(\hat{N}+1)}{1-p^*(\hat{N}+1)}\right)^j (C_{vj}(\boldsymbol{d}) - C_{v(j+1)}(\boldsymbol{d}))}{\sum_{j=0}^{\hat{N}} \binom{\hat{N}}{j} \left(\frac{p^*(\hat{N}+1)}{1-p^*(\hat{N}+1)}\right)^j (C_{vj}(\boldsymbol{d}) - C_{v(j+1)}(\boldsymbol{d}))}, \tag{3.25}$$

where $\hat{N}$ only takes integer values. $q_v^*(\hat{K})$ defined in (3.24) is non-increasing in $\hat{K}$. Furthermore, if $b > \max\{1, x^* - \gamma_{\epsilon_v}\}$ holds with strict inequality, then $q_v^*(\hat{K})$ is strictly decreasing in $\hat{K}$ for $\hat{K} \geq J_{\epsilon_v}(\boldsymbol{d})$.

According to [21, Theorem 4], Assumption 4 should also hold true. Furthermore, because $p^*(\hat{K})$ is non-increasing in $\hat{K}$, given the number of users K, channel contention measure $q_v(p^*(\hat{K})$ $\boldsymbol{d}, K)$ as a function of $\hat{K}$ is non-decreasing in $\hat{K}$. According to [21, Theorem 3], $q_v(p^*(\hat{K})\boldsymbol{d}, K)$ is strictly increasing in $\hat{K}$ for all $K > K_{\min}$ and $\hat{K} \geq \max\{J_{\epsilon_v}(\boldsymbol{d}), x^* - b\}$. Consequently, Assumption 5 should hold true due to the monotonicity properties of $q_v(p^*(\hat{K})\boldsymbol{d}, K)$ and $q_v^*(\hat{K})$.

It is important to note that system design also includes the design of the virtual packet, which affects the virtual channel parameter function set. In the case of a fixed $\boldsymbol{d}$, virtual packet should be chosen to support reasonable sensitivity of channel contention measure to the variation of number of users. As explained in [21, Section 3], a general principle is to choose a virtual packet design such that $J_{\epsilon_v}(\boldsymbol{d})$ and γ_{ϵ_v} are both slightly less than x^* and therefore $b \geq \max\{1, x^* - \gamma_{\epsilon_v}\}$ can take a value close to 1. Also as explained in [21, Section 4], when the receiver does not feedback q_v and each user only knows the success/failure status of its own packets, the distributed MAC algorithm can be revised to use an interpreted channel contention measure and, according to computer simulations, the system can still converge to the same designed system equilibrium.

3.4 A Search-Assisted Design Approach

In this section, we consider the general scenario when transmission direction vectors of the users are not fixed. To understand the challenges in the design of $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K})$ functions, we first take a look at a simple example.

Example 1. *Consider a time-slotted multiple access network over a multi-packet reception channel. Each user is equipped with two transmission options respectively labeled as the high-rate option and the low-rate option. If all packets are encoded using the low-rate option, then the channel can support the parallel transmissions of no more than 12 packets. We assume that one packet from the high-rate option is equivalent to the combination of 4 low-rate packets. That is, the chan-*

nel can support the parallel transmissions of n_h high-rate packets plus n_l low-rate packets if and only if $\frac{1}{3}n_h + \frac{1}{12}n_l \leq 1$. The utility function is chosen to be the sum system throughput. Suppose that all users should hold the same transmission probability vector $\boldsymbol{p} = [p_h, p_l]^T$ where p_h and p_l denote the probabilities of a user choosing the high-rate option and the low-rate option, respectively. We obtain the optimum probability vector as $\boldsymbol{p}^* = [p_h^*, p_l^*]^T = \arg\max_{\boldsymbol{p}} U(K, \boldsymbol{p}, \{C_{rij}(\boldsymbol{d})\})$. Figure 3.1 illustrates p_h^* and p_l^* as functions of the number of users. We can see that, if we write $\boldsymbol{p}^* = p^*\boldsymbol{d}^*$,

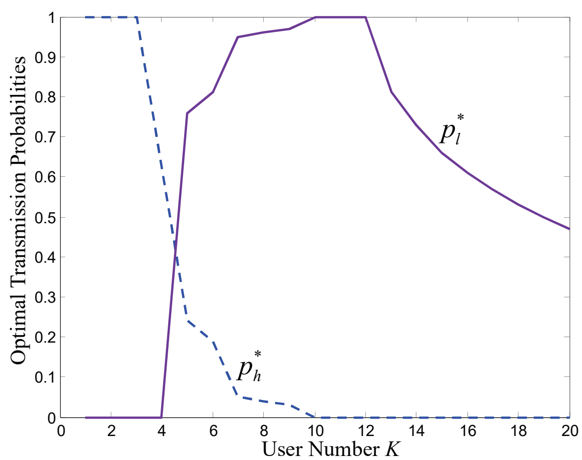

Figure 3.1: Optimal transmission probabilities of a K-user multiple access system with each user having two transmission options.

then $\boldsymbol{d}^*$ is fixed at $\boldsymbol{d}^* = [1, 0]^T$ for $K \leq 4$, and is fixed at $\boldsymbol{d}^* = [0, 1]^T$ for $K \geq 10$. $\boldsymbol{d}^*$ transits from $[1, 0]^T$ to $[0, 1]^T$ in the region of $4 \leq K \leq 10$.

According to the above observation, we assume that the theoretical transmission probability vector function $\boldsymbol{p}^*(\hat{K}) = p^*(\hat{K})\boldsymbol{d}^*(\hat{K})$ should be designed to satisfy the following properties termed the "Head and Tail Condition".

Condition 3. *(Head and Tail) Let $\epsilon_v > 0$ be a pre-determined constant. Let $J_{\epsilon_v}(\boldsymbol{d})$ be defined in (3.12). There exist two integer-valued constants $0 < \underline{K} \leq \overline{K}$, such that,*

1. *$\underline{K} \geq J_{\epsilon_v}(\boldsymbol{d}^*(\underline{K}))$ and $\boldsymbol{d}^*(\hat{K}) = \boldsymbol{d}^*(\underline{K})$ for $\hat{K} \leq \underline{K}$.*

2. *$\overline{K} > J_{\epsilon_v}(\boldsymbol{d}^*(\overline{K}))$ and $\boldsymbol{d}^*(\hat{K}) = \boldsymbol{d}^*(\overline{K})$ for $\hat{K} \geq \overline{K}$.*

The Head and Tail Condition indicates that, when $\hat{K}$ is either small enough or large enough, $\boldsymbol{d}^*(\hat{K})$ should stop changing in $\hat{K}$. Consequently, in the "Head" regime defined as $\hat{K} \leq \underline{K}$, and in the "Tail" regime defined as $\hat{K} \geq \overline{K}$, $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K})$ functions should be designed using the closed-form approach specified in Section 3.3.

Now consider the regime of $\underline{K} \leq \hat{K} \leq \overline{K}$. Because we usually have $\boldsymbol{d}^*(\underline{K}) \neq \boldsymbol{d}^*(\overline{K})$, if the designed equilibrium needs to be close to optimal, then theoretical transmission probability vector function $\boldsymbol{p}^*(\hat{K})$ designed for $\underline{K} \leq \hat{K} \leq \overline{K}$ should involve a transition of the transmission direction vector from $\boldsymbol{d}^*(\underline{K})$ to $\boldsymbol{d}^*(\overline{K})$. Unfortunately, due to generality of the system model, when users change their transmission direction vectors, it is difficult to argue whether the outcome should increase/decrease the channel contention measure. Consequently, it becomes difficult to argue for monotonicity properties on channel contention measure functions $q_v(\boldsymbol{p}^*(\hat{K}), K)$ and $q_v^*(\hat{K})$. To overcome such a challenge, we switch to a search-assisted approach whose basic idea is illustrated as follows. We will first choose several integer-valued $\hat{K}$ points, termed "Pinpoints", and assume that $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K}) = q_v(\boldsymbol{p}^*(\hat{K}), \hat{K})$ should be manually determined for the pinpoints. After that, an interpolation approach will be used to connect the pinpoints and to determine $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K})$ functions for all $\underline{K} \leq \hat{K} \leq \overline{K}$. Key objective of the pinpoints selection and their corresponding design is to make sure that the theoretical transmission probability vector function $\boldsymbol{p}^*(\hat{K})$ is close to optimal in terms of network utility optimization at equilibrium for all $\underline{K} \leq \hat{K} \leq \overline{K}$. Key objective of the interpolation approach, on the other hand, is to make sure that $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K})$ functions designed to connect the pinpoints should satisfy the monotonicity and continuity requirements presented in Assumptions 2-5.

We require that the following condition should be satisfied by the pinpoints.

Condition 4. *(Pinpoints) Let $\hat{K}_i$ for $i = 0, \ldots, L$ be $L + 1$ integers such that $\underline{K} = \hat{K}_0 < \hat{K}_1 < \cdots < \hat{K}_L = \overline{K}$. For $i = 0, \ldots, L$ and $0 \leq \lambda < 1$, define*

$$\hat{K}_{i\lambda} = (1-\lambda)\hat{K}_{i-1} + \lambda\hat{K}_i,$$
$$\boldsymbol{d}^*_{i\lambda} = (1-\lambda)\boldsymbol{d}^*(\hat{K}_{i-1}) + \lambda\boldsymbol{d}^*(\hat{K}_i),$$
$$q^*_{vi\lambda} = (1-\lambda)q^*_v(\hat{K}_{i-1}) + \lambda q^*_v(\hat{K}_i). \tag{3.26}$$

1. *There exists a positive constant ϵ_q to satisfy $q^*_v(\hat{K}_{i-1}) - q^*_v(\hat{K}_i) \geq \epsilon_q$, for all $i = 1, \ldots, L$.*

2. *There exists a constant $\epsilon_v > 0$, such that for all $i = 1, \ldots, L$ and for all $0 \leq \lambda < 1$, we have $\hat{K}_{i\lambda} > J_{\epsilon_v}(\boldsymbol{d}^*_{i\lambda})$, where $J_{\epsilon_v}(\boldsymbol{d}^*_{i\lambda})$ is defined in (3.12).*

3. *There exist $0 < \underline{p} < \overline{p} < 1$ to satisfy $\underline{p} \leq p(\hat{K}_i) \leq \overline{p}$ for all $i = 1, \ldots, L$.*

4. *Let $N = \lfloor \hat{K} \rfloor$. Define $q_N(\boldsymbol{p})$ and $q_{N+1}(\boldsymbol{p})$ as in (3.23). Extend the definition of $q_v(\boldsymbol{p}, \hat{K})$ for non-integer-valued $\hat{K}$ as*

$$q_v(\boldsymbol{p}, \hat{K}) = (N + 1 - \hat{K})q_N(\boldsymbol{p}) + (\hat{K} - N)q_{N+1}(\boldsymbol{p}), \tag{3.27}$$

The following inequality should be satisfied for all $i = 1, \ldots, L$ and for all $0 \leq \lambda < 1$.

$$q_v(\overline{p}\boldsymbol{d}^*_{i\lambda}, \hat{K}_{i\lambda}) \leq q^*_{vi\lambda} \leq q_v(\underline{p}\boldsymbol{d}^*_{i\lambda}, \hat{K}_{i\lambda}). \tag{3.28}$$

With $\boldsymbol{p}^*(\hat{K})$ being designed for the $L+1$ pinpoints, we propose the following interpolation approach to complete $\boldsymbol{p}^*(\hat{K})$ and $q^*_v(\hat{K})$ functions for $\underline{K} \leq \hat{K} \leq \overline{K}$.

Interpolation Approach Assume that $\boldsymbol{p}^*(\hat{K})$ is designed for $\hat{K}_i$, $i = 0, ...L$, with $\underline{K} = \hat{K}_0 < \hat{K}_1 < \cdots < \hat{K}_L = \overline{K}$, to satisfy Condition 4. For $i = 1, \ldots, L$ and for all $0 \leq \lambda < 1$, let $\hat{K}_{i\lambda}$, $\boldsymbol{d}^*_{i\lambda}$ and $q^*_{vi\lambda}$ be defined in (3.26). Let $q_v(\boldsymbol{p}, \hat{K})$ be defined in (3.27). We choose $p^*(\hat{K}_{i\lambda})$ to satisfy the following equality.

$$q_v(p^*(\hat{K}_{i\lambda})\boldsymbol{d}^*_{i\lambda}, \hat{K}_{i\lambda}) = q^*_{vi\lambda}. \tag{3.29}$$

This leads to $\boldsymbol{p}^*(\hat{K}_{i\lambda}) = p^*(\hat{K}_{i\lambda})\boldsymbol{d}^*_{i\lambda}$. Note that the existence of a solution to (3.29) is guaranteed by Item 4 of Condition 4.

Effectiveness of the Interpolation Approach is stated in the following theorem.

Theorem 8. *Assume that $\boldsymbol{p}(\hat{K})$ is designed for a set of $L+1$ pinpoints $\{\hat{K}_i\}$, $i = 0, \ldots, L$, with $\underline{K} = \hat{K}_0 < \hat{K}_1, \ldots, < \hat{K}_L = \overline{K}$, to satisfy Condition 4. After completing the functions using the Interpolation Approach, $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K})$ functions satisfy Assumptions 2-5 for $\underline{K} \leq \hat{K} \leq \overline{K}$.*

The proof of Theorem 8 is given in appendix A.2.

Note that the search-assisted design approach can also be adopted in the simple scenario when either all users have the same pre-fixed $\boldsymbol{d}$ vector or each user has a single transmission option. When there is a noticeable gap between the optimal performance, in terms of network utility maximization at equilibrium, and the performance of the $\boldsymbol{p}^*(\hat{K})$ function designed using the closed-form approach, one can adjust $\boldsymbol{p}^*(\hat{K})$ at carefully selected pinpoints to further improve its optimality. Also note that when users have multiple transmission options, the system should choose a virtual packet design such that channel contention measure is reasonably sensitive to the change of number of users for all transmission option choices. While we did not provide theoretical guidance on virtual packet design and pinpoint selections for the general scenario, in the next section, we will use several examples to show that coming up with a reasonably good design should not be a difficult task.

3.5 Simulation Results

In this section, we provide computer examples to illustrate both optimality and convergence properties of the proposed MAC algorithm.

Example 1: In [13], a similar stochastic approximation model was proposed for the maximization of symmetric sum throughput of a distributed multiple access network over a collision channel. Assume that there are K users each having a single transmission option and a saturated message queue. If K is known, the optimum solution that maximizes the symmetric sum throughput is to set the transmission probabilities of all users at $p_{\text{opt}} = \frac{1}{K}$ [13]. In [13], under the assumption of an unknown number of users and due to the constraint of certain monotonicity properties, it was suggested that equilibrium of the distributed MAC algorithm should be set at p_a, which is obtained by solving the following equation.

$$eP(\text{idle}) - 1 - 0.5\sqrt{p_a} = 0, \quad P(\text{idle}) = (1 - p_a)^K, \tag{3.30}$$

where $P(\text{idle})$ is the idling probability of the channel that can be measured without knowing the value of K.

Let us follow the design guideline presented in Section 3.3 of this dissertation. With the collision channel model, the real channel parameter set $\{C_{rj}\}$ is given by $C_{r0} = 1$ and $C_{rj} = 0$ for $j > 0$. With the utility chosen to be the symmetric sum throughput, we get from (3.20) that $x^* = 1$. Assume that a virtual packet should have the same coding parameters of a real packet. Consequently, the virtual channel parameter set $\{C_{vj}\}$ is given by $C_{v0} = 1$ and $C_{vj} = 0$ for $j > 0$. Choose $\epsilon_v = 0.01$, we get $\gamma_{\epsilon_v} = J_{\epsilon_v} = 0$. Therefore, we can set $b = 1.01 > x^* - \gamma_{\epsilon_v}$. This leads to an equilibrium with $p^* = \frac{1}{K+1.01}$.

In Figure 3.2, we illustrate the achieved sum throughput of the system in packet/slot as a function of the number of users under the optimal transmission probability, at the equilibrium of the proposed distributed MAC algorithm, and at the equilibrium of the approach suggested in [13]. It

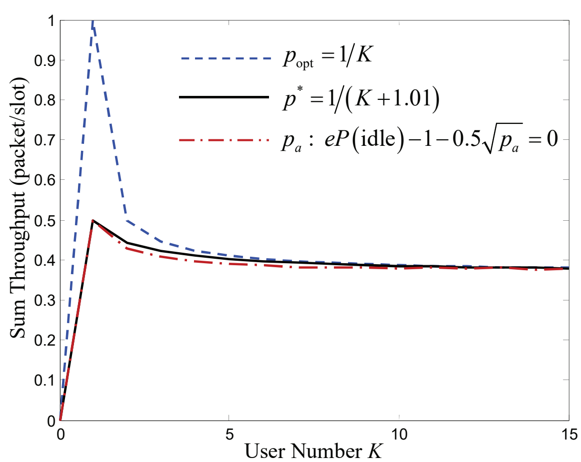

Figure 3.2: Sum throughput as a function of the user number for a multiple access network with a collision channel.

can be seen that, for the classical scenario presented in [12], the distributed MAC algorithm proposed in this dissertation can achieve a throughput performance better than the approach proposed in [12], although the improvement is indeed marginal.

Example 2: In this example, we consider distributed multiple access networking over a simple fading channel. Assume that the system has K users and one receiver. Each user only has a single transmission option. In each time slot, with a probability of 0.3, the channel can support no more than $M_1 = 4$ parallel real packet transmissions, and with a probability of 0.7, the channel can support no more than $M_2 = 6$ parallel real packet transmissions[5]. In this case, the real channel parameter set $\{C_{rj}\}$ is given by $C_{rj} = 1$ for $j < 4$, $C_{rj} = 0.7$ for $4 \leq j < 6$, and $C_{rj} = 0$ for $j \geq 6$. Assume that users intend to maximize the symmetric system throughput weighted by a transmission energy cost of $E = 0.3$. With K users all transmitting at the same probability of p, system utility $U(K, p, \{C_{rj}\})$ is given by

$$U(K, p, \{C_{rj}\}) = -EKp + \sum_{j=0}^{K-1} K \binom{K-1}{j} p^{j+1}(1-p)^{K-1-j} C_{rj}. \tag{3.31}$$

Correspondingly, x^* can be obtained from asymptotic utility optimization (3.20) as $x^* = 3.29$. Assume that a virtual packet should have the same coding parameters as those of a real packet. The virtual channel parameter set $\{C_{vj}\}$ is therefore identical to the real channel parameter set, i.e., $C_{vj} = C_{rj}$ for all $j \geq 0$. Choose $\epsilon_v = 0.01$, we have $\gamma_{\epsilon_v} = J_{\epsilon_v} = 3$. Therefore, we can set $b = 1.01$.

In Figure 3.3, we illustrate three utilities, all as functions of user number K. The solid curve represents the utility achieved by the proposed MAC algorithm at the designed equilibrium, with all users transmitting at a probability of $p^* = \min\left\{p_{\max}, \frac{x^*}{K+b}\right\}$. The dashed curve represents the optimum utility under the assumption that number of users K is known. Note that the optimum utility is not necessarily achievable without the knowledge of K. The dash-dotted curve represents the utility if we maintain the channel idling probability at its asymptotically optimal value of

[5]Such a channel can appear if there is an interfering user that transmits a packet with a probability of 0.3 in each time slot. One packet from the interfering user is equivalent to the combination of two packets from a regular user.

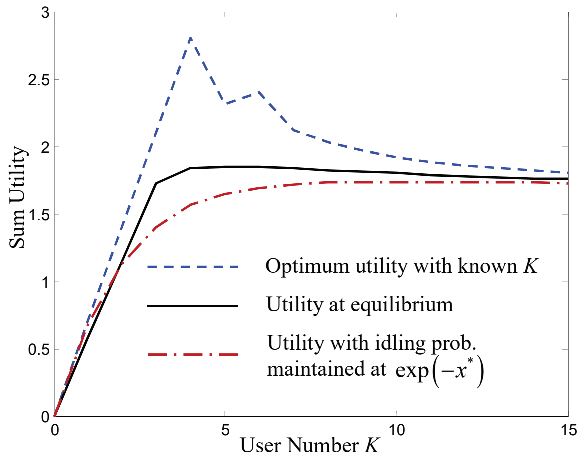

Figure 3.3: Sum utility as a function of the user number for a multiple access network over a simple fading channel.

$\exp(-x^*)$, or equivalently, if we set the transmission probabilities of all users at $1 - \exp\left(-\frac{x^*}{K}\right)$. This is an intuitive extension to the key idea suggested in [13], although a general channel model was not discussed in [13][6].

Next, we assume that the system has $K = 8$ users. Transmission probabilities of all users are initialized at 0. In each time slot, a channel state flag is randomly generated to indicate whether the channel can support the parallel transmissions of no more than 4 or 6 packets. Each user also randomly determines whether a packet should be transmitted according to its own transmission probability parameter. Whether the real packets and the virtual packet can go through the channel or not is then determined using the corresponding channel model. We use the following exponential moving average approach to measure q_v. q_v is initialized at $q_v = 1$. In each time slot, q_v is updated by $q_v = (1 - \frac{1}{300})q_v + \frac{1}{300}I_v$, where $I_v \in \{0, 1\}$ is an indicator of the success/failure reception status of the virtual packet in the current time slot. While this is different from the approach proposed in the distributed MAC algorithm, simulations show that an exponential averaging measurement of q_v can often lead the system to convergence in a relatively small number of time slots. The rest of probability updates proceeds according to the proposed distributed MAC algorithm with a

[6]Note that, other suggestions of maintaining certain variable at its asymptotically optimal value, as discussed in [13], do not give a better performance in this example.

constant step size of $\alpha = 0.05$. Convergence behavior of the sum utility is illustrated in Figure 3.4, where sum utility is also measured using the same exponential moving average approach except that initial value of the utility is set at 0.

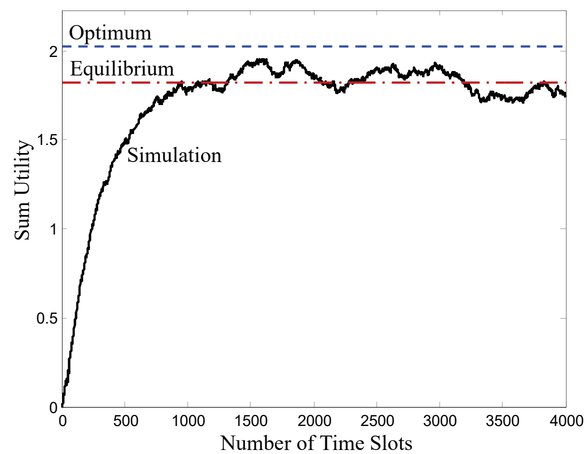

Figure 3.4: Convergence in sum utility of a system with $K = 8$ users.

Example 3: In this example, we use the system introduced in Example 1 of Section 3.4 to illustrate the design procedure of the $\boldsymbol{p}^*(\hat{K})$ function when users have multiple transmission options. First, we consider the "Head" and the "Tail" regimes when $\hat{K}$ is either small or large in value. We will add subscript "H" (or "T") to parameters of the "Head" (or the "Tail") regime. Without specifying the values of $\underline{K}$ and $\overline{K}$, we first determine the optimal transmission direction vectors in these two regimes as $\boldsymbol{d}_H = [1, 0]^T$ and $\boldsymbol{d}_T = [0, 1]^T$. In other words, users should only use the high rate option in the "Head" regime and only use the low rate option in the "Tail" regime. In the "Head" regime, the channel can support the parallel transmissions of no more than 3 high rate packets. The real channel parameter set of the equivalent single option system is given by $\{C_{rj}\}_H$ with $C_{rj} = 1$ for $j \leq 2$ and $C_{rj} = 0$ otherwise. By following the design guideline of Section 3.3, we get $x_H^* = \arg\max_x (x + x^2 + \frac{x^3}{2})e^{-x} = 2.27$. We design the virtual packet to be equivalent to a real high rate packet. Consequently, the virtual channel parameter set of the equivalent single option system is given by $\{C_{vj}\}_H = \{C_{rj}\}_H$. Choose $\epsilon_v = 0.01$, we get $\gamma_{\epsilon_v H} = J_{\epsilon_v H} = 2$, and $b_H = 1.01$. In the "Tail" regime, on the other hand, the channel can support the parallel transmissions of no more than 12 low rate packets. The real channel parameter set of the equivalent single option system is given by $\{C_{rj}\}_T$ with $C_{rj} = 1$ for $j \leq 11$ and $C_{rj} = 0$ otherwise. This leads to $x_T^* = \arg\max_x \sum_{i=0}^{11} \frac{x^{i+1}}{i!} e^{-x} = 8.82$. Because we already chose the virtual packet to be equivalent to a high rate real packet, virtual channel parameter set of the equivalent single option system in this case is given by $\{C_{vj}\}_T$ with $C_{vj} = 1$ for $j \leq 8$ and $C_{vj} = 0$ otherwise. Therefore, with $\epsilon_v = 0.01$, we have $\gamma_{\epsilon_v T} = J_{\epsilon_v T} = 8$. Luckily, this supports $b_T = 1.01$.

Next, we determine the values of $\underline{K}$ and $\overline{K}$. We first compare two schemes named the "high rate option only" scheme and the "low rate option only" scheme. In the "high rate option only" scheme, we fix $\boldsymbol{d}^*(\hat{K})$ at $[1, 0]^T$ for all $\hat{K}$, and set $p^*(\hat{K}) = \min\left\{p_{\max H}, \frac{x_H^*}{\hat{K}+b_H}\right\}$, where $p_{\max H} = \frac{x_H^*}{J_{\epsilon_v H}+b_H}$. In the "low rate option only" scheme, we fix $\boldsymbol{d}^*(\hat{K})$ at $[0, 1]^T$ for all $\hat{K}$, and set $p^*(\hat{K}) = \min\left\{p_{\max T}, \frac{x_T^*}{\hat{K}+b_T}\right\}$, where $p_{\max T} = \frac{x_T^*}{J_{\epsilon_v T}+b_T}$. By comparing utility values and theoretical channel contention measures of the two schemes, we choose $\underline{K} = 4$ and $\overline{K} = 10$.

Now consider the "Pinpoints Condition" for $\underline{K} \leq \hat{K} \leq \overline{K}$. For transmission direction vectors $\boldsymbol{d}$ satisfying $d_1 > 0$, with a small enough ϵ_v, we generally have $J_{\epsilon_v} = 2$. Therefore, so long as $\boldsymbol{d}^*(\hat{K})$ does not transit too quickly to $[0, 1]^T$, the condition of $\hat{K} > J_{\epsilon_v}(\boldsymbol{d}^*(\hat{K}))$ should hold true. Consequently, only two other key conditions need to be satisfied. The first condition is that $q_v^*(\hat{K})$ of the selected pinpoints must be strictly decreasing in $\hat{K}$. The second condition is that $p^*(\hat{K})$ found in the Interpolation Approach should be bounded away from 0 and 1. In addition, from the optimal scheme, we can see that $\boldsymbol{d}^*(\hat{K})$ should transit toward $[0, 1]^T$ faster than a linear transition from $\hat{K} = \underline{K}$ to $\hat{K} = \overline{K}$. With these considerations, we choose the following 4 pinpoints. At the edge of the "Head" and the "Tail" regimes, we have $\hat{K}_0 = \underline{K} = 4$ with $\boldsymbol{p}^*(4) = \frac{x_H^*}{\underline{K}+b_H}[1, 0]^T$ and $\hat{K}_3 = \overline{K} = 10$ with $\boldsymbol{p}^*(10) = \frac{x_T^*}{\overline{K}+b_T}[0, 1]^T$. We also choose other two pinpoints at $\hat{K}_1 = 5$ and $\hat{K}_2 = 6$. We set transmission directions vectors $\boldsymbol{d}^*(5)$ and $\boldsymbol{d}^*(6)$ to be equal to the corresponding optimal transmission direction vectors, i.e., direction vectors extracted from the optimal $\boldsymbol{p}$ vectors that maximize the sum throughput at $K = 5$ and $K = 6$, respectively. Transmission probabilities of these two pinpoints are chosen such that the resulting $q_v^*(\hat{K})$ equals $\frac{\overline{K}-\hat{K}}{\overline{K}-\underline{K}} q_v^*(\underline{K}) + \frac{\hat{K}-\underline{K}}{\overline{K}-\underline{K}} q_v^*(\overline{K})$. Note that, the purpose of designing pinpoints $\hat{K}_1 = 5$ and $\hat{K}_2 = 6$ is to help $\boldsymbol{d}^*(\hat{K})$ to transit quickly toward $[0, 1]^T$. The rest of the $\boldsymbol{p}^*(\hat{K})$ function is completed using the Interpolation Approach for $\underline{K} \leq \hat{K} \leq \overline{K}$. Theoretical channel contention measure $q_v^*(\hat{K})$ of the designed system is illustrated in Figure 3.5.

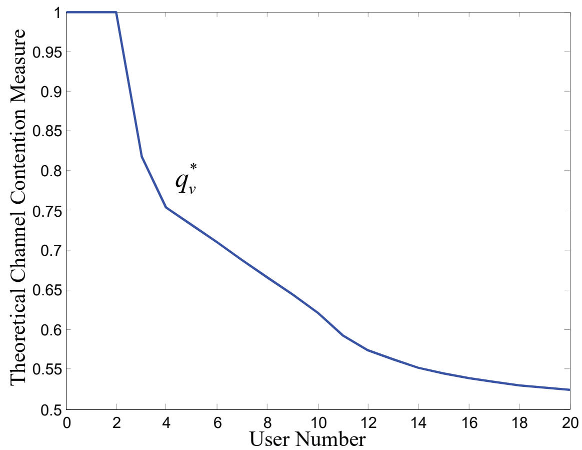

Figure 3.5: Theoretical channel contention measure q_v^* as a function of the user number.

In Figure 3.6, we illustrate the theoretical sum throughput of the network as functions of the number of users K when the transmission probability vectors of all users are set at the following four different vectors: optimal $\boldsymbol{p}(K)$ that maximizes the sum throughput, designed $\boldsymbol{p}^*(K)$, $\boldsymbol{p}^*(K)$ from the high rate option only scheme, and $\boldsymbol{p}^*(K)$ from the low rate option only scheme. Note again that the optimal sum throughput is not necessarily achievable without the knowledge of K. Assume that the high rate only scheme and the low rate only scheme should be reasonably good for the "Head" and the "Tail" regimes, respectively. It can be seen from Figure 3.6 that, with the help of the designed $\boldsymbol{p}^*(\hat{K})$ and $q_v^*(\hat{K})$ functions, the system can take advantage of the multiple transmission options and maintain a reasonably good performance in term of sum throughput for all user number values.

Next, we illustrate the convergence property of the proposed distributed MAC algorithm. Assume that the system has 8 users initially. Transmission probability vectors of all users are initialized at $[0, 0]^T$. In each time slot, according to its own transmission probability vector, each user randomly determines whether a packet should be transmitted or not, and if the answer is positive,

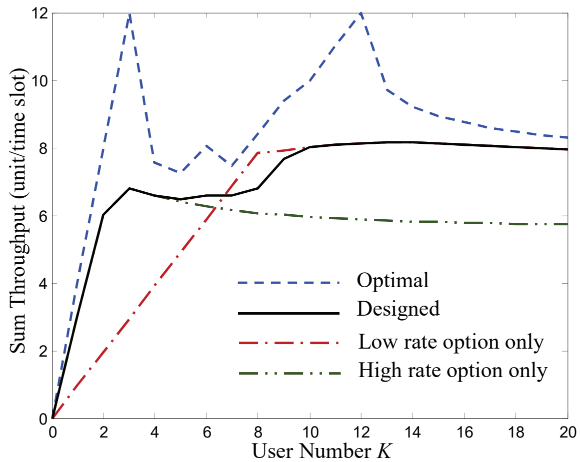

Figure 3.6: Sum throughput of the system as functions of the user number under different transmission probability vector settings.

which transmission option should be used. The receiver measures q_v using the following exponential moving average approach. q_v is initialized at $q_v = 1$. In each time slot, an indicator variable $I_v \in \{0, 1\}$ is used to represent the success/failure status of the virtual packet reception. q_v is then updated by $q_v = (1 - \frac{1}{300})q_v + \frac{1}{300}I_v$, and is fed back to the users at the end of each time slot. Each user then adapts its transmission probability vector according to the proposed MAC algorithm with a constant step size of $\alpha = 0.05$.

We assume that the system experiences three stages. At Stage one, the system has 8 users. The system enters Stage two at the 3001st time slot, when 6 more users enter into the system with their transmission probability vectors initialized at $[0, 0]^T$. Then at the 6001st time slot, the system enters Stage three when 8 users exit the system. Convergence behavior in sum throughput of the system is illustrated in Figure 3.7. The corresponding optimal throughput and the theoretical throughput at the designed equilibrium are provided as references. In Figure 3.8, we also illustrate entries of the target transmission probability vector calculated by the users together with the corresponding theoretical values. Note that the simulated throughput and probability values presented in the figures are measured using the same exponential averaging approach explained above. From Figures 3.7 and 3.8, we can see that the proposed MAC algorithm can indeed help users to adapt to the changes of stages and to adjust their transmission probability vectors to the new equilibrium.

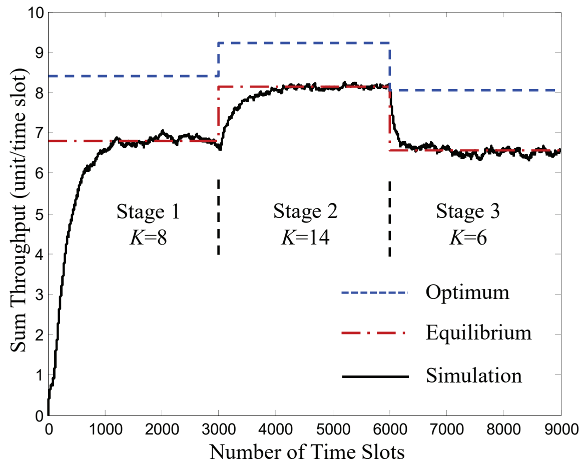

Figure 3.7: Convergence in sum throughput of the system. User number changed from 8 to 14 and then to 6 over the three stages.

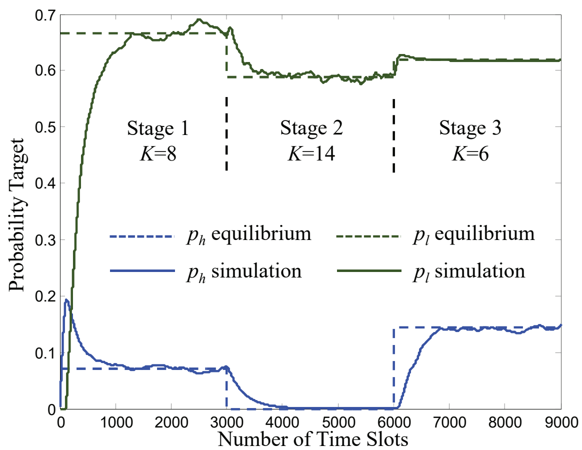

Figure 3.8: Entries of the transmission probability vector target and their corresponding theoretical values.

According to the Head and Tail Condition, the system degrades to an equivalent single option system when $K \leq \underline{K}$ and $K \geq \overline{K}$. It is generally expected that transmission direction vectors of the "Head" and the "Tail" regimes should be different, i.e., $\boldsymbol{d}(\underline{K}) \neq \boldsymbol{d}(\overline{K})$. In Example 3.5, we found one virtual packet design that supports both $b_H = 1.01$ in the "Head" regime and $b_T = 1.01$ in the "Tail" regime. One may think that such a lucky result should be rare. Surprisingly, according to our observations, in most of the problems of interest, even though one may not always be able to get the perfect result of $b_H = b_T \approx 1$, a single virtual packet can often be designed to support close to ideal values on $J_{\epsilon_v H}$, b_H, $J_{\epsilon_v T}$, and b_T. While it is possible to extend the system design and to improve design flexibility by including the transmissions of multiple (different) virtual packets in each time slot, because performance improvement provided by such an extension is often marginal, we choose to skip the corresponding discussions in this dissertation.

3.6 Conclusion

We investigated distributed multiple access networking with an unknown finite number of homogeneous users. An enhanced physical-link layer interface is considered where each link layer user can be equipped with multiple transmission options. With a generally modeled link layer channel, we proposed distributed MAC algorithms to adapt the transmission schemes of the users to maximize a chosen symmetric network utility. Convergence property of the proposed MAC algorithms is proven under quite mild conditions. While there is no theoretical guarantee on the optimality of the proposed MAC algorithms, simulation results suggest that performances of the proposed MAC algorithms are often not too far from optimal.

Chapter 4

Distributed MAC Algorithm for Hierarchical Users

Diversity of wireless devices and applications often requires wireless networks to provide differentiated services to users in the sense of supporting user groups with different priority levels. Take the enhanced DCF (EDCF) protocol in 802.11e for example [27]. Users (or traffics) in 802.11e EDCF can be assigned to four different priority levels with different adaptation schemes on their backoff windows. A high priority user generally maintains a backoff window smaller in size than that of a low priority user. Consequently, when messages are available, transmission probability of a high priority user is always larger in value than that of a low priority user. This gives high priority users an advantage over low priority users in getting their packets through the shared wireless channel. However, when the system has a large number of users whose transmission activities cause a significant level of contention, packet transmission success probability of each user can still be driven down close to zero, irrespective of the priority level of the user.

Recent trend of dynamic spectrum access (DSA) created the new demand of supporting hierarchical user structure in wireless systems [28][29]. Take channel sharing with the primary-secondary user structure for example. It is expected that secondary users should access the channel only if they can guarantee no disturbance to communication activities of the primary users. Existing DSA literature often assumes that secondary users should be able to identify whether an existing transmission should belong to a primary user or not. Disruptive interference is often avoided with online coordinations between primary and secondary users or within the secondary user group. There is little discussion on how to support hierarchical user groups in a random access environment, where users do not exchange information with each other directly and packet collision is part of the natural transmission outcomes.

In this chapter, we extend the distributed MAC framework proposed in Chapter 3 to support hierarchical user groups in a time-slotted random multiple access system [30]. The hierarchical user structure is established in the following senses. First, when the number of primary users is small, the MAC protocol guarantees that transmission success probability of each primary user will stay above a pre-determined threshold no matter how many secondary users are competing for the channel. Second, when the number of primary users is large, the MAC protocol guarantees that transmission probability of each secondary user will be driven down to zero. However, the MAC protocol does not reject channel access to any primary user even though transmission activities of the primary users naturally lead to a low packet transmission success probability. We introduce the distributed MAC framework first for random access systems where each user only has a single transmission option. The MAC framework is then extended to systems where each user is equipped with multiple transmission options. Simulation results are provided to demonstrate performances of the distributed MAC algorithms with various system settings.

4.1 Supporting Hierarchical Users with Single Transmission Option

In this section, we will show that the distributed MAC algorithm presented in Chapter 3 can be extended to support hierarchical user groups in random multiple access systems. To simplify the discussion, we assume that each user only has a single transmission option. In this case, each user is associated with a scalar transmission probability parameter. The two parameter sets used to model the link layer multiple access channel are simplified to $\{C_{rj}\}$, termed the "real channel parameter set", and $\{C_{vj}\}$, termed the "virtual channel parameter set" [8, Section 4.2]. If all users have the same transmission probability p, and the system actually has K users, "channel contention measure", which is the success probability of the virtual packet, is given by

$$q_v(p, K) = \sum_{j=0}^{K} \binom{K}{j} p^j (1-p)^{K-j} C_{vj}. \tag{4.1}$$

Assume that the system now has K_p primary users and K_s secondary users. The values of K_p and K_s are unknown to the users as well as to the receiver. However, each user still views the system as one with homogeneous users. On one hand, a primary user, for example user k_p, intends to maximize a utility of $U_p(\hat{K}, p_p, \{C_{rj}\})$, under the assumption that the system contains $\hat{K}$ homogeneous primary users and all users have the same transmission probability p_p. Based on this objective, as explained in Section 3.3, user k_p should design its desired transmission probability function as

$$p_p^*(\hat{K}) = \frac{x_p^*}{\max\{\hat{K}, \hat{K}_{p\min}\} + b_p}, \tag{4.2}$$

where $\hat{K}_{p\min}$ and b_p are design parameters whose values are determined by following the guideline given in Section 3.3, and x_p^* is obtained from the following asymptotic utility optimization.

$$x_p^* = \arg\max_x \lim_{\hat{K}\to\infty} U_p\left(\hat{K}, \frac{x}{\hat{K}}, \{C_{rj}\}\right). \tag{4.3}$$

With the $p_p^*(\hat{K})$ function given by (4.2), and the $q_v(p, K)$ function defined by (4.1), the "theoretical channel contention measure" function $q_{vp}^*(\hat{K})$ for the primary users is given by

$$\begin{aligned} q_{vp}^*(\hat{K}) = & \frac{p_p^*(\hat{K}) - p_p^*(\lfloor\hat{K}\rfloor + 1)}{p_p^*(\lfloor\hat{K}\rfloor) - p_p^*(\lfloor\hat{K}\rfloor + 1)} q_v(p_p^*(\hat{K}), \lfloor\hat{K}\rfloor) \\ & + \frac{p_p^*(\lfloor\hat{K}\rfloor) - p_p^*(\hat{K})}{p_p^*(\lfloor\hat{K}\rfloor) - p_p^*(\lfloor\hat{K}\rfloor + 1)} q_v(p_p^*(\hat{K}), \lfloor\hat{K}\rfloor + 1), \end{aligned} \tag{4.4}$$

where $\lfloor\hat{K}\rfloor$ represents the largest integer below $\hat{K}$.

On the other hand, a secondary user, for example user k_s, intends to maximize a utility of $U_s(\hat{K}, p_s, \{C_{rj}\})$, under the assumption that the system contains $\hat{K}$ homogeneous secondary users and all users have the same transmission probability p_s. Based on this objective, user k_s should design its desired transmission probability function as

$$p_s^*(\hat{K}) = \frac{x_s^*}{\max\{\hat{K}, \hat{K}_{s\min}\} + b_s}, \tag{4.5}$$

where $\hat{K}_{s\min}$ and b_s are design parameters whose values are determined by following the guideline given in Section 3.3. The value of x_s^*, however, is determined differently from that of the primary users. Let $D_s \geq 1$ be a pre-determined discount factor. x_s^* for the secondary users is obtained by an asymptotic utility optimization, but is also upper bounded by x_p^*/D_s.

$$x_s^* = \min\left\{\arg\max_x \lim_{\hat{K}\to\infty} U_s\left(\hat{K}, \frac{x}{\hat{K}}, \{C_{rj}\}\right), \frac{x_p^*}{D_s}\right\}. \tag{4.6}$$

Under the assumption that the system contains $\hat{K}$ homogeneous secondary users and all users have the same transmission probability p_s, with $p_s^*(\hat{K})$ function given by (4.5) and $q_v(p_s, K)$ function given by (4.1), the "theoretical channel contention measure" function $q_{vs}^*(\hat{K})$ for the secondary users is given by

$$\begin{aligned} q_{vs}^*(\hat{K}) = & \frac{p_s^*(\hat{K}) - p_s^*(\lfloor\hat{K}\rfloor + 1)}{p_s^*(\lfloor\hat{K}\rfloor) - p_s^*(\lfloor\hat{K}\rfloor + 1)} q_v(p_s^*(\hat{K}), \lfloor\hat{K}\rfloor) \\ & + \frac{p_s^*(\lfloor\hat{K}\rfloor) - p_s^*(\hat{K})}{p_s^*(\lfloor\hat{K}\rfloor) - p_s^*(\lfloor\hat{K}\rfloor + 1)} q_v(p_s^*(\hat{K}), \lfloor\hat{K}\rfloor + 1). \end{aligned} \tag{4.7}$$

With the above design, the distributed MAC algorithm operates as introduced in Chapter 3, except that each primary user should replace $q_v^*(\hat{K})$ and $\boldsymbol{p}^*(\hat{K})$ functions in the algorithm by $q_{vp}^*(\hat{K})$ and $p_p^*(\hat{K})$, respectively, and each secondary user should replace $q_v^*(\hat{K})$ and $\boldsymbol{p}^*(\hat{K})$ functions in the algorithm by $q_{vs}^*(\hat{K})$ and $p_s^*(\hat{K})$, respectively. The proposed MAC algorithm supports the hierarchical user structure in the following sense.

Theorem 9. *Let K_p be the number of primary users in the system. The value of K_p is unknown to the users as well as to the receiver. With the proposed MAC algorithm, the system should possess a unique equilibrium. Let channel contention measure at the equilibrium be denoted by q_v. If $q_{vp}^*(K_p) \geq q_{vs}^*(\infty)$, on one hand, $q_v \geq q_{vs}^*(\infty)$ must hold at the equilibrium. If $q_{vp}^*(K_p) < q_{vs}^*(\infty)$, on the other hand, transmission probabilities of the secondary users should equal zero at the equilibrium.*

The proof of theorem 9 is given in AppendixB.

Note that, because the discount factor satisfies $D_s \geq 1$, transmission probability of a secondary user should be no larger than that of a primary user at the equilibrium. This consequently implies that $q_{vs}^*(\infty) \geq q_{vp}^*(\infty)$. Also note that, as shown in (4.6), if $\arg\max_x \lim_{\hat{K}\to\infty} U_s\left(\hat{K}, \frac{x}{\hat{K}}, \{C_{rj}\}\right) >$

$\frac{x_p^*}{D_s}$, we should have $x_s^* = \frac{x_p^*}{D_s}$. In this case, the value of x_s^* is set according to the channel yielding requirement. The objective of maximizing utility $U_s\left(\hat{K}, \frac{x}{\hat{K}}, \{C_{rj}\}\right)$ for the secondary users becomes irrelevant. This is a disadvantage for the secondary users due to the hierarchical structure.

4.2 Supporting Hierarchical Users with Multiple Transmission Options

Now assume that all users are equipped with M transmission options plus an idling option. Each user, for example user k, is associated with a transmission probability vector $\boldsymbol{p}_k = p_k\boldsymbol{d}_k$, where p_k is the transmission probability and $\boldsymbol{d}_k$ is the transmission direction vector. Under the assumption that all users have the same transmission direction vector $\boldsymbol{d}$, the link layer multiple access channel is modeled using two sets of parameter functions, namely the "real channel parameter function set" $\{C_{rij}(\boldsymbol{d})\}$ and the "virtual channel parameter function set" $\{C_{vj}(\boldsymbol{d})\}$ [8, Section 4.3].

Let the numbers of primary and secondary users of the system be K_p and K_s, respectively. The values of K_p and K_s are unknown to the users as well as to the receiver. We assume that each user still views the system as one with homogeneous users. A primary user, for example user k_p, intends to maximize a utility of $U_p(\hat{K}, p_p\boldsymbol{d}_p, \{C_{rij}(\boldsymbol{d}_p)\})$, under the assumption that the system contains $\hat{K}$ homogeneous primary users and all users have the same transmission probability vector $\boldsymbol{p}_p = p_p\boldsymbol{d}_p$. Based on this objective, user k_p should design its desired transmission probability vector function $\boldsymbol{p}_p^*(\hat{K})$ and calculate its theoretical channel contention measure function $q_{vp}^*(\hat{K})$ using the search-assisted approach presented in Section 3.4. The design guarantees that $q_{vp}^*(\hat{K})$ should be strictly decreasing in $\hat{K}$ for $\hat{K} \in [\hat{K}_{p\min}, \infty)$, where $\hat{K}_{p\min}$ is a design parameter explained in Section 3.4. Let us write $\boldsymbol{p}_p^*(\hat{K}) = p_p^*(\hat{K})\boldsymbol{d}_p^*(\hat{K})$. It was shown in Section 3.4 that there exists a user number upper bound $\overline{K}_p$, such that for all $\hat{K} \geq \overline{K}_p$, we have

$$\boldsymbol{d}_p^*(\hat{K}) = \boldsymbol{d}_p^*(\overline{K}_p), \qquad p_p^*(\hat{K}) = \frac{x_p^*}{\hat{K} + b_p}, \tag{4.8}$$

where b_p is a design parameter whose values should be determined by following the guideline given in Section 3.3, and x_p^* is obtained from the following asymptotic utility optimization.

$$x_p^* = \arg\max_x \lim_{\hat{K}\to\infty} U_p\left(\hat{K}, \frac{x}{\hat{K}}\boldsymbol{d}_p^*(\overline{K}_p), \{C_{rij}(\boldsymbol{d}_p^*(\overline{K}_p))\}\right). \tag{4.9}$$

In other words, for $\hat{K} \geq \overline{K}_p$, the desired transmission direction vector becomes invariant to the estimated number of users, and the desired transmission probability takes a form similar to the case when each user only has a single transmission option.

A secondary user, for example user k_s, intends to maximize a utility of $U_s(\hat{K}, p_s\boldsymbol{d}_s, \{C_{rij}(\boldsymbol{d}_s)\})$, under the assumption that the system contains $\hat{K}$ homogeneous secondary users and all users have the same transmission probability $\boldsymbol{p}_s = p_s\boldsymbol{d}_s$. While user k_s still uses the search-assisted approach to design its desired transmission probability vector function $\boldsymbol{p}_s^*(\hat{K})$ and to calculate its theoretical channel contention measure function $q_{vs}^*(\hat{K})$, we require that $\boldsymbol{p}_s^*(\hat{K})$ function should satisfy the following constraints. Recall that $\boldsymbol{p}_p^*(\hat{K}) = p_p^*(\hat{K})\boldsymbol{d}_p^*(\hat{K})$ is the desired transmission probability vector function for the primary users. Let $\boldsymbol{p}_s^*(\hat{K}) = p_s^*(\hat{K})\boldsymbol{d}_s^*(\hat{K})$. Let $D_s \geq 1$ be the discount factor. We require that secondary users should choose a user number threshold $\overline{K}_s \geq \overline{K}_p$ to satisfy the following constraint for all $\hat{K} \geq \overline{K}_s$,

$$\boldsymbol{d}_s^*(\hat{K}) = \boldsymbol{d}_p^*(\hat{K}) = \boldsymbol{d}_p^*(\overline{K}_p), \qquad p_s^*(\hat{K}) = \frac{x_s^*}{\hat{K} + b_s}, \tag{4.10}$$

where b_s is a design parameter whose values should be determined by following the guideline given in Section 3.3, and x_s^* is obtained by

$$x_s^* = \min\left\{\frac{x_p^*}{D_s}, \arg\max_x \lim_{\hat{K}\to\infty} U_s\left(\hat{K}, \frac{x}{\hat{K}}\boldsymbol{d}_s^*(\overline{K}_p), \{C_{rij}(\boldsymbol{d}_s^*(\overline{K}_p))\}\right)\right\}.$$

In other words, in addition to asymptotic utility optimization, we also make sure that x_s^* is upper-bounded by $\frac{x_p^*}{D_s}$. As shown in Section 3.4, one often needs to choose several pinpoints first for the $\boldsymbol{p}_s^*(\hat{K})$ function and then to complete the function using an interpolation approach. In this design procedure, equation (4.11) essentially sets a special pinpoint for $\boldsymbol{p}_s^*(\hat{K})$ at $\hat{K} = \overline{K}_s$ (and sets the $\boldsymbol{p}_s^*(\hat{K})$ function for $\hat{K} \geq \overline{K}_s$). The rest of the function still needs to be designed using the search-assisted approach to guarantee that the corresponding theoretical channel contention measure function $q_{vs}^*(\hat{K})$ should be strictly decreasing in $\hat{K}$ for $\hat{K} \in [\hat{K}_{s\min}, \infty)$, where $\hat{K}_{s\min}$ is a design parameter explained in Section 3.3.

With the above design approach, the distributed MAC algorithm operates as introduced in Chapter 3, except that each primary user should replace $q_v^*(\hat{K})$ and $\boldsymbol{p}^*(\hat{K})$ functions in the algorithm by $q_{vp}^*(\hat{K})$ and $\boldsymbol{p}_p^*(\hat{K})$, respectively, and each secondary user should replace $q_v^*(\hat{K})$ and $\boldsymbol{p}^*(\hat{K})$ functions in the algorithm by $q_{vs}^*(\hat{K})$ and $\boldsymbol{p}_s^*(\hat{K})$, respectively. The proposed MAC algorithm supports the hierarchical user structure in the following sense.

Theorem 10. *Let K_p be the number of primary users in the system. The value of K_p is unknown to the users as well as to the receiver. Let q_v be the channel contention measure at an equilibrium of the system. With the proposed MAC algorithm, if $q_{vp}^*(K_p) \geq q_{vs}^*(\infty)$, then $q_v \geq q_{vs}^*(\infty)$ must hold. If $q_{vp}^*(K_p) < q_{vs}^*(\infty)$, on the other hand, then transmission probabilities of the secondary users should equal zero at the equilibrium.*

Proof of the theorem is quite straightforward and is therefore skipped.

Note that there are two key differences between the MAC algorithms presented in Sections 4.1 and 4.2. First, when users are equipped with multiple transmission options, we can no longer prove that the system should possess a unique equilibrium, even though we believe this should be the case under mild conditions. The challenge comes from the fact that, due to generality of the system model, it is difficult to compare channel contention measures when users change their transmission direction vectors. Second, in the MAC algorithm presented in Section 4.1, channel yielding constraint on the $p_s^*(\hat{K})$ function is applied for all $\hat{K}$ values. In the MAC algorithm presented in Section 4.2, however, channel yielding constraint on the $\boldsymbol{p}_s^*(\hat{K})$ function is applied only for $\hat{K} \geq \overline{K}_p$. While the latter approach gives more freedom in designing the $\boldsymbol{p}_s^*(\hat{K})$ function, it generally requires the search-assisted design approach as introduced in [8, Section 4.3] even when each user is only equipped with a single transmission option. The approach presented in Section 4.1, on the other hand, provides closed form expressions for both $p_p^*(\hat{K})$ and $p_s^*(\hat{K})$ functions.

In the DSA literature, hierarchical channel sharing approaches are categorized into "overlay" and "underlay" schemes [29]. In an overlay scheme, secondary users can access the channel with significant transmission power but only when primary users are not present. In an underlay scheme, on the other hand, secondary users can access the channel under the constraint that their aggregated interference should be controlled below a pre-determined level. The distributed MAC algorithms introduced in Sections 4.1 and 4.2 of this paper can be viewed as an underlay scheme for random access networks where aggregated "interference" of the users is evaluated using the designed channel contention measure.

4.3 Simulation Results

In this section, we present simulation results to demonstrate effectiveness of the proposed MAC algorithms in supporting hierarchical user groups.

Example 1: In the first example, we consider a random multiple access system where each user is equipped with only a single transmission option. The channel is a classical collision channel that can support the successful transmission of no more than one user in a time slot if all other users idle. Assume that the virtual packet should be identical to a real packet. Virtual packet reception should be regarded as successful if and only if all users idle in a time slot. Consequently, channel contention measure, which is the success probability of the virtual packet, should equal the idling probability of the collision channel.

Assume that primary users intend to maximize the symmetric throughput if there is no secondary user in the system. In other words, under the assumption that there are K primary users, and all users have the same transmission probability p, utility function of the primary users is given by $U_p(K, p) = p(1-p)^{K-1}$. According to [8, Section 4.2], we can choose the desired transmission probability function of the primary users as $p_p^*(\hat{K}) = \frac{1}{\hat{K}+1.01}$, for $\hat{K} \geq 1$. This implies that $b_p = 1.01$ and $x_p^* = 1$. Consequently, the theoretical channel contention measure function for the primary users is given by $q^*_{vp}(\hat{K}) = \left(1 - \frac{1}{\hat{K}+1.01}\right)^{\hat{K}}$. Let the discount factor be $D_s = 1.15$. Assume that the secondary users also intend to maximize the symmetric throughput of the system. However, due to channel yielding requirement, the secondary users should choose the desired transmission probability function as $p^*_s(\hat{K}) = \frac{1/D_s}{\hat{K}+1.01}$, for $\hat{K} \geq 1$. This implies that $b_s = 1.01$ and $x^*_s = \frac{x^*_p}{D_s} = \frac{1}{1.15} = 0.87$. Consequently, the function of theoretical channel contention measure for the secondary users is given by $q^*_{vs}(\hat{K}) = \left(1 - \frac{0.87}{\hat{K}+1.01}\right)^{\hat{K}}$. Note that $q^*_{vs}(\hat{K}) \geq q^*_{vs}(\infty) = e^{-0.87}$.

In Figure 4.1, we plotted the theoretical channel contention measure functions for primary users $q^*_{vp}(\hat{K})$ and for secondary users $q^*_{vs}(\hat{K})$. It can be seen that, key idea of supporting the hierarchical

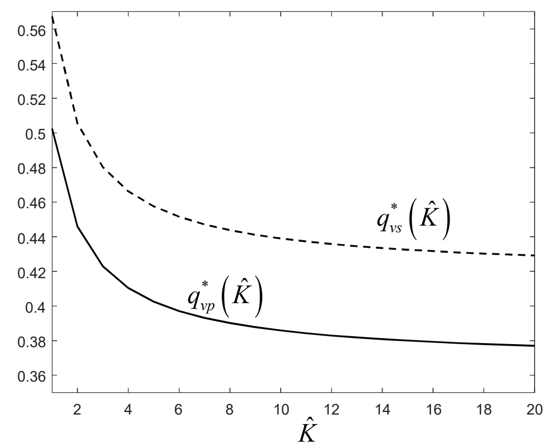

Figure 4.1: Theoretical channel contention measure functions for primary and secondary users.

user structure is to raise the tail of the $q^*_{vs}(\hat{K})$ function for the secondary users, such that aggregated impact of the secondary users on the idling probability of the channel is well controlled no matter how many secondary users want to access the channel.

In Figure 4.2, we plotted channel contention measure of the system at its unique equilibrium as functions of the number of primary users K_p and the number of secondary users K_s. The figure shows that, when the number of primary users is small $K_p \leq 3$, we have $q^*_{vp}(K_p) > q^*_{vs}(\infty) = 0.42$. In this case, secondary users can access the channel. But the system keeps the channel idling probability above $q^*_{vs}(\infty) = 0.42$ irrespective of the number of secondary users. When the number

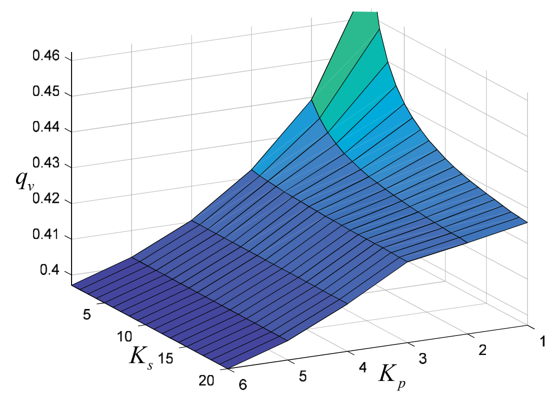

Figure 4.2: Ilding probability as a function of the numbers of primary and secondary users.

of primary users is large $K_p > 3$, on the other hand, we have $q^*_{vp}(K_p) < q^*_{vs}(\infty)$. In this case transmission probabilities of the secondary users are kept at zero, and therefore q_v is not affected by the number of secondary users.

Example 2: The second example is extended from Example 1 of Section 3.4. Consider a time-slotted multiple access network over a multi-packet reception channel. Each user is equipped with two transmission options where the first option is a high-rate option and the second option is a low-rate option, respectively. If all packets are encoded using the low-rate option, then the channel can support the parallel transmissions of no more than 12 packets. We assume that one packet from the high-rate option is equivalent to the combination of 4 low-rate packets. Therefore, the channel can support the parallel transmissions of n_1 high-rate packets plus n_2 low-rate packets if and only if $\frac{1}{3}n_1 + \frac{1}{12}n_2 \leq 1$. As explained in 1 of Section 3.4, we design the virtual packet to be equivalent to a high rate real packet. The two sets of channel parameter functions $\{C_{rij}(\boldsymbol{d})\}$ and $\{C_{vj}(\boldsymbol{d})\}$ can therefore be derived accordingly.

We assume that the primary users intend to maximize the sum system throughput. That is, if the system has K homogeneous users and all users have the same transmission probability vector $\boldsymbol{p} = [p_1, p_2]^T = p[d_1, d_2]^T$, utility function of the primary users is given by

$$U_p(K, \boldsymbol{p}, \{C_{rij}(\boldsymbol{d})\}) = K \sum_{i=1}^{2} d_i r_i$$
$$\times \sum_{j=0}^{K-1} \binom{K-1}{j} p^{j+1}(1-p)^{K-1-j} C_{rij}(\boldsymbol{d}), \tag{4.11}$$

where $r_1 = 4$, $r_2 = 1$ are the rate parameters of the two options.

According to 1 of Section 3.4, the desired transmission probability vector function $\boldsymbol{p}_p^*(\hat{K})$ for the primary users is designed as follows. For $\hat{K} \le 2$, we set $\boldsymbol{p}_p^*(\hat{K}) = \frac{2.27}{2+1.01}[1,0]^T$. For $2 < \hat{K} \le 4$, we set $\boldsymbol{p}_p^*(\hat{K}) = \frac{2.27}{\hat{K}+1.01}[1,0]^T$. For $\hat{K} \ge 10$, we set $\boldsymbol{p}_p^*(\hat{K}) = \frac{8.82}{\hat{K}+1.01}[0,1]^T$, which implies that $x_p^* = 8.82$. Note that once $\boldsymbol{p}_p^*(\hat{K})$ function is determined for $\hat{K} \le 4$ and $\hat{K} \ge 10$, the theoretical channel contention measure function $q_{vp}^*(\hat{K})$ can be calculated accordingly. For $4 \le \hat{K} \le 10$, we choose two pinpoints at $\hat{K} = 5$ and $\hat{K} = 6$. We set transmission direction vectors $\boldsymbol{d}_p^*(5)$ and $\boldsymbol{d}_p^*(6)$ for the pinpoints at the same direction vectors corresponding to the probability vectors that maximize the utility (4.11). With $\boldsymbol{d}_p^*(5)$, $\boldsymbol{d}_p^*(6)$, and $\boldsymbol{d}_p^*(4) = [1,0]^T$, $\boldsymbol{d}_p^*(10) = [0,1]^T$, we then set $\boldsymbol{d}_p^*(\hat{K})$ for $4 \le \hat{K} \le 10$ such that $\boldsymbol{d}_p^*(\hat{K})$ transit linearly in $\hat{K}$ between the neighboring pinpoints. After that, we choose the transmission probability function $p_p^*(\hat{K})$ for $4 \le \hat{K} \le 10$ such that the resulting $q_{vp}^*(\hat{K})$ function is linear in $\hat{K}$ for $4 \le \hat{K} \le 10$.

Let the discount factor be $D_s = 1.1$. For the secondary users, we assume that they also intend to maximize sum throughput of the system under the assumption that the system only contains homogeneous secondary users. With the channel yielding requirement, we design the desired transmission probability vector function $\boldsymbol{p}_s^*(\hat{K})$ for the secondary users as follows. Let $\boldsymbol{p}_s^*(\hat{K}) = \boldsymbol{p}_p^*(\hat{K})$ for $\hat{K} \le 4$. For $\hat{K} \ge 18$, we set $\boldsymbol{p}_s^*(\hat{K}) = \frac{8.82/D_s}{\hat{K}+1.01}[0,1]^T = \frac{8.02}{\hat{K}+1.01}[0,1]^T$. For $4 \le \hat{K} \le 18$, we simply set $\boldsymbol{d}_s^*(\hat{K}) = \boldsymbol{d}_p^*(\hat{K})$, and then choose $p_s^*(\hat{K})$ such that the resulting theoretical channel contention measure function $q_{vs}^*(\hat{K})$ is linear in $\hat{K}$ for $4 \le \hat{K} \le 18$.

Figure 4.3 shows the theoretical channel contention measure functions $q_{vp}^*(\hat{K})$ and $q_{vs}^*(\hat{K})$, respectively. In this example, while we raised the tail of the $q_{vs}^*(\hat{K})$ function for the secondary users, the section of $q_{vs}^*(\hat{K})$ for $\hat{K} \le 4$ remains the same as $q_{vp}^*(\hat{K})$. Therefore, secondary users and primary users are treated equally if the total number of users is no larger than 4. Such a design

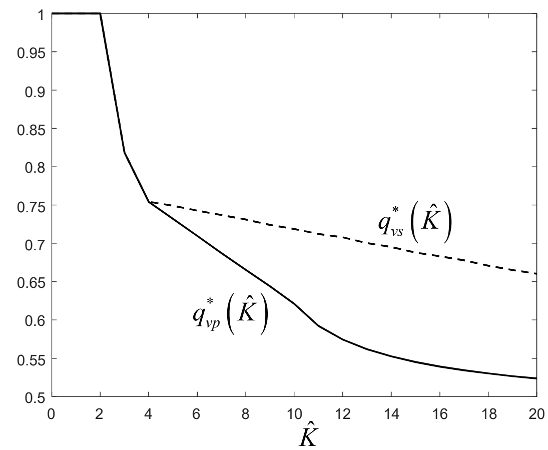

Figure 4.3: Theoretical channel contention measure functions for primary and secondary users.

is feasible because of a relatively small-valued discount factor D_s, so that $q^*_{vs}(4) > q^*_{vs}(\infty)$ can be satisfied. The design is also enabled due to the flexibility of the search-assisted design approach.

In this example, although we are not able to prove that equilibrium of the system should be unique, it is indeed the case according to numerical search. In Figure 4.4, channel contention measure q_v at the equilibrium is plotted as a function of the number of primary users K_p and the number of secondary users K_s. It can be seen that, when the number of primary users is small

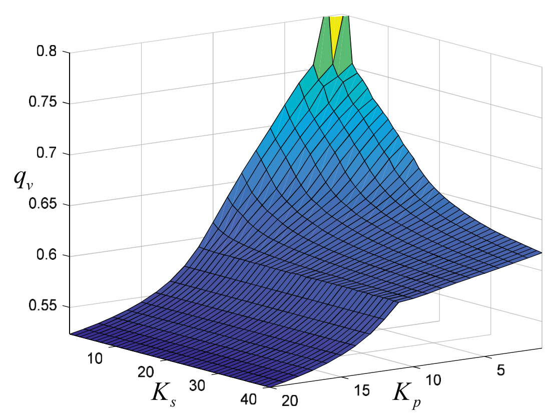

Figure 4.4: Channel contention measure at the equilibrium as a function of the number of primary users and the number of secondary users.

$K_p < 12$, we have $q^*_{vp}(K_p) > q^*_{vs}(\infty) = 0.59$. In this case, the system allows secondary users to access the channel but keeps the channel contention measure q_v above $q^*_{vs}(\infty) = 0.59$ irrespective of the number of secondary users. When the number of primary users is large $K_p \geq 12$, on the other hand, we have $q^*_{vp}(K_p) < q^*_{vs}(\infty)$. In this case transmission probabilities of the secondary users are kept at zero, and therefore q_v is not affected by the number of secondary users.

Next, we assume that the system has 6 primary users and 10 secondary users initially. Transmission probabilities of the users are initialized at $[0, 0]^T$. In each time slot, according to its own associated probability vector, a user randomly determines whether to transmit a packet or not, and if the answer is positive, which option should be used. The receiver uses an exponential moving average approach to measure q_v. More specifically, q_v is initialized at $q_v = 1$. In each time slot, an indicator variable $I_v \in \{0, 1\}$ is used to represent the success/failure status of the virtual packet reception. q_v is then updated as $q_v = (1 - \frac{1}{300})q_v + \frac{1}{300}I_v$, and is fed back to the transmitters at the end of each time slot. With the updated q_v, each user adapts its transmission probability vector according to the MAC algorithm proposed in Section 4.2 with a constant step size of $\alpha = 0.05$.

We assume that the system experiences three stages. At the beginning in Stage one, the system has 6 primary users. The system enters Stage two after the 3000th time slot, when 6 more primary users enter the system with their transmission probability vectors initialized at $[0, 0]^T$. After the 6000th time slot, the system enters Stage three when 9 primary users exit the system. Throughout the three stages, the number of secondary users is kept at 10. Convergence behavior in actual channel contention measure q_v is illustrated in Figure 4.5 together with the theoretical q_v at the corresponding equilibria of the three stages. The figure demonstrates that the system can quickly adapt to the user number changes and keep channel contention at the desired level. In Figure 4.6, we also illustrated entries of the transmission probability vector targets calculated by the primary and the secondary users. Note that values of the simulated variables presented in Figures 4.5 and 4.6 are calculated using the same exponential averaging approach explained above. It can be seen that the system is reasonably responsive to user number changes and can quickly lead transmission probability vectors of both primary users and secondary users to their corresponding theoretical

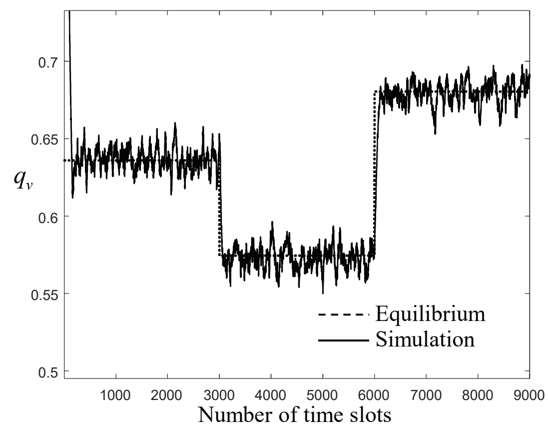

Figure 4.5: Channel contention measure of the system through three stages.

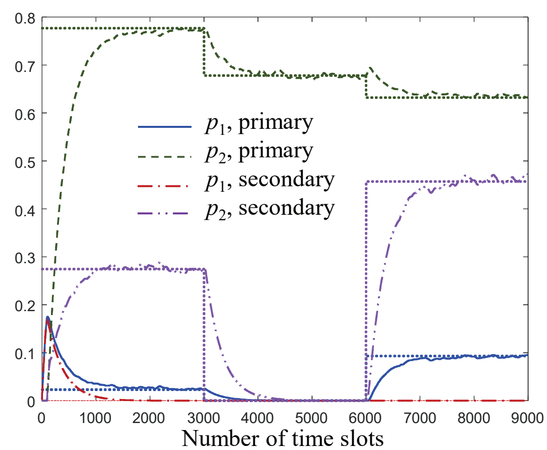

Figure 4.6: Transmission probabilities of primary and secondary users through three stages. Dashed lines represent the corresponding values at the equilibrium.

equilibrium values. The hierarchical user structure can be seen clearly in the sense that secondary users always transmit with a low rate option at a relatively low probability.

4.4 Conclusion

We proposed a distributed MAC framework to support hierarchical user groups in random multiple access systems. The MAC algorithms do not require direct message exchange among users. Users do not need to know the number of primary and secondary users in the system. Users also do not need the capability of identifying whether a transmitted packet should belong to a primary user or to a secondary user. The proposed MAC algorithm adapts the transmission scheme of each user by comparing the actual channel contention measure to a theoretical channel contention measure function. With the simple idea of raising the tail of the theoretical channel contention measure function for the secondary users, aggregated impact of the secondary users on contention level of the channel is well controlled no matter how many secondary users compete for the channel. We extended the proposed MAC algorithm to systems where each user is equipped with multiple transmission options. Simulation results showed that the proposed MAC algorithm can maintain the primary-secondary user structure and can also be reasonably responsive in a dynamic environment with users joining and existing the system.

Chapter 5

Distributed MAC Algorithm for Multiple Channels

In this chapter, we introduce an ongoing research work that is not yet completed. We propose an extension to the distributed MAC algorithm of Chapter 3 for a multiple access network with multiple non-interfering channels. A preliminary illustration of its potential convergence and performance analysis is presented. We still assume that the system contains an unknown finite number of homogeneous users. However, differs from the single channel case investigated in Chapter 3, in a multi-channel system, enforcing the assumption that homogeneous users should converge to the same transmission scheme at equilibrium becomes suboptimal. For example, let us consider a distributed multiple access network with K homogeneous users sharing two parallel collision channels. Assume that K is even-valued. Let each user be equipped with one transmission option for each channel. Assume that users intend to maximize the symmetric sum throughput. If users are forced to have the same transmission schemes at equilibrium, then the optimal solution is to let all users transmit in both channels, at a probability of $\frac{1}{K}$ for each channel. This leads to a sum throughput of $2\left(1-\frac{1}{K}\right)^{K-1}$. However, if we partition users into two groups, each with $\frac{K}{2}$ users accessing only one channel. Then each user should transmit in one channel at a probability of $\frac{2}{K}$. The sum throughput in this case equals $2\left(1-\frac{2}{K}\right)^{\frac{K}{2}-1}$, which is larger than $2\left(1-\frac{1}{K}\right)^{K-1}$. In other words, despite the fact that users are homogeneous and channels are homogeneous, a distributed MAC algorithm should still guide users to heterogeneous transmission schemes at equilibrium. In this chapter, we handle such a problem by assuming that each user should be assigned to only one channel. On one hand, for users being assigned to one channel, we adopt the distributed MAC algorithm presented in Chapter 3 to guide their transmission probability vectors toward a common target. On the other hand, we also propose a distributed algorithm to adapt channel assignments of the users toward the objective of balancing the loads of the channels. A scheme of convergence analysis is provided to illustrate the idea of a potential proof that the extended distributed

MAC algorithm should lead the system to a desired equilibrium. Simulation results are provided to demonstrate the performance of the proposed MAC algorithm.

The rest of the chapter is organized as follows. In Section 5.1, we extend the distributed MAC algorithm to the case of multiple non-interfering channels and provide an illustration scheme of convergence analysis. Simulation results are presented in Section 5.2 to illustrate the performance of the proposed MAC algorithm under various network settings.

5.1 Distributed MAC Over Multiple Channels

In this section, we extend the distributed MAC algorithm presented in Chapter 3 to a multiple access network with K homogeneous users and W non-interfering channels. Note that, while the users are homogeneous, the channels can be heterogeneous. Assume that K is unknown to the users and to the receiver. Each user is backlogged with a message queue and is equipped with M transmission options plus an idling option for each channel.

We assume that time is partitioned into frames, with detail of the partitioning approach being explained later. In each time frame, each user can assign itself to one and only one channel. Users can change their channel assignments at the beginning of each time frame, but not within a time frame. Let us first focus on one arbitrary time frame. Because channel assignments are fixed within the time frame, the multi-channel system essentially operates as multiple parallel single-channel systems. Let w_k denote the channel chosen by user k. We use K_w to denote the number of users being assigned to channel w. Therefore, $\sum_w K_w = K$. We will use a superscript $[w]$ to denote parameters corresponding to channel w.

In each time slot, the receiver assumes the existence of a virtual packet for each channel. While virtual packets assumed in different time slots for the same channel are identical, virtual packets assumed for different channels can be different. Let $q_v^{[w]}$ be the contention measure, which is the success probability of the virtual packet, for channel w. We assume that the receiver should obtain estimated contention measures for all channels and feed them back to the users. Under the assumption that all users should adopt the same transmission direction vector $\boldsymbol{d}$ for channel w, we model the channel using two sets of parameter functions, the "real channel parameter function set" $\{C_{rij}^{[w]}(\boldsymbol{d})\}$ and the "virtual channel parameter function set" $\{C_{vj}^{[w]}(\boldsymbol{d})\}$. Both $\{C_{rij}^{[w]}(\boldsymbol{d})\}$ and $\{C_{vj}^{[w]}(\boldsymbol{d})\}$ for all w are supposed to be known at the receiver.

Consider an arbitrary user, for example user k. If user k is assigned to channel $w_k = w$, then user k should maintain a transmission probability vector $\boldsymbol{p}_k^{[w]}$ for channel w. We write $\boldsymbol{p}_k^{[w]} = p_k^{[w]}\boldsymbol{d}_k^{[w]}$, where $p_k^{[w]}$ is the transmission probability and $\boldsymbol{d}_k^{[w]}$ is the transmission direction vector of user k for channel w. We assume that users intend to maximize a sum utility denoted by

$$U = \sum_{w} U^{[w]}(K_w, \boldsymbol{p}^{[w]}, \{C_{rij}^{[w]}(\boldsymbol{d}^{[w]})\}). \tag{5.1}$$

Here $U^{[w]}(K_w, \boldsymbol{p}^{[w]}, \{C_{rij}^{[w]}(\boldsymbol{d})^{[w]}\})$ is the utility function of channel w under the assumption that K_w users are assigned to channel w, and all the K_w users should adopt the same transmission probability vector $\boldsymbol{p}^{[w]} = p^{[w]}\boldsymbol{d}^{[w]}$ whose transmission direction vector is $\boldsymbol{d}^{[w]}$. Given the utility function of each channel, for example channel w, the system should design two key functions, namely the "theoretical transmission probability vector" function $\boldsymbol{p}^{*[w]}(K_w)$, and the "theoretical channel contention measure" function $q_v^{*[w]}(K_w)$. These functions should be designed according to the guideline given in Chapter 3 such that the distributed MAC algorithm presented in Chapter 3 can be adopted to help users in channel w to adapt their transmission probability vectors within one time frame toward the designed target of $\boldsymbol{p}^{*[w]}(K_w)$.

Now let us consider multiple time frames. Let T be a predetermined constant. We say that time slot t_1 belongs to the nth time frame, if the following condition is satisfied[7].

$$n \leq \frac{1}{T}\sum_{t=0}^{t_1} \alpha(t) < n+1, \tag{5.2}$$

[7]Here t_1 is the value of the time counter introduced in the distributed MAC algorithm presented in Chapter 3

where $\alpha(t)$ is the step size parameter of time slot t and T is a constant. During each time frame, channel assignment is fixed. Therefore, multi-channel system converts to a system with parallel channels during one time frame.

We continue our discussion in two parts, first we investigate homogeneous channels and then consider heterogeneous channels.

5.1.1 Homogeneous Channels

In this section, we consider the case of homogeneous channels and assume that virtual packets designed for different channels are identical. Upon receiving $q_v^{[w]}$, users in channel w calculates an estimated number of users in channel w, denoted by $\hat{K}_w$. At the beginning of a time frame, based on estimated number of users in each channel, each user say user k transmitting in channel w_k, randomly decides to switch to another channel with probability p_{sw}. The channel that user k may switch to, should have the least estimated number of users and this number should be ϵ_K-less than the estimated number of users in channel w_k, where $1 < \epsilon_K < 2$ is a positive constant. In other words, user k may switch to channel w_l if

$$\begin{aligned} &\hat{K}^{[w_l]} \leq \hat{K}^{[w]} \text{ for } w = 1, \dots, W \\ &\hat{K}^{[w_l]} + \epsilon_K < \hat{K}^{[w_k]}. \end{aligned} \tag{5.3}$$

The value of ϵ_K depends on T and convergence rate of probability vectors. Probability of switching from one channel to another one may not be the same for different channels, which means p_{sw} can be a function of the difference between the estimated number of users in the current channel and in the channel that the user may switch to. If no user can find a channel with ϵ_K-less estimated number of users than the currently assigned channel of the user, then no switching should take place.

The extended distributed MAC algorithm for multiple channels operates as follows. At the beginning of the first time frame, each user randomly assigns itself to a channel. Next, users follow the presented MAC algorithm in Chapter 3 during the time frame. Then, at the beginning of each time frame, each user say user k in channel w randomly decides to switch to channel w_l with probability p_{sw} if (5.3) is satisfied. After possible change of channel assignment at the beginning of a time frame, users then fix their channel assignments in the rest of the time frame and follow the MAC algorithm presented in Chapter 3.

To illustrate the potential convergence property of the distributed MAC algorithm, let us assume that each time frame is long enough such that transmission probability vectors of users in each channel should converge asymptotically to the designed equilibrium at the end of the time frame. Consequently, with a probability close to one, users should be able to correctly estimate the number of users assigned to each channel. Under this assumption, the process of channel assignment transitions becomes an absorbing Markov chain [31]. More specifically, on one hand, if the number of users of different channels are not balanced, then at the beginning of each time frame, users should randomly switch their channel assignments toward the under-loaded channels. In expectation, the system should move in the direction of balancing the loads of the channels. On the other hand, once the system reaches an "absorbing state" in the sense that the number of users of any channel is not larger than the number of users of another channel, then with a probability close to one, users should maintain their current channel assignment. In this case, loads of the channels should remain balanced, or in other words, the system should stay at an absorbing state.

In the non-asymptotic case when the length of each time frame is finite and the system adopts a small constant step size, users should have a high probability to correctly estimate the number of users of each channel at the end of each time frame. Therefore, with a high probability, the system should converge toward the set of absorbing states at the beginning of each time frame. While the probability of the system leaving the absorbing states is non-zero, such a probability can be reduced arbitrarily close to zero by increasing the length of the time frame and decreasing the value of the step size. Because the system should have a high probability to converge toward the set of absorbing states once it leaves the set, according to the theory of absorbing Markov chain [31], it is reasonable to expect that convergence to the absorbing states can be proven both in strong and weak senses.

5.1.2 Heterogeneous Channels

When the channels are heterogeneous, it is no longer appropriate to compare loads of the channels by comparing their estimated number of users. Instead, per user utility of each channel is a more direct measure of its load. To make a decision for channel assignment, each user, say user k, should calculate the per user utility value of every channel, denoted by $\frac{1}{\hat{K}_w}U^{[w]}(\hat{K}_w)$ for channel w. Let ϵ_U be a predetermined positive constant satisfying $1 < \epsilon_U < 2$. Assume that user k is currently assigned to channel w. At the beginning of a time frame, user k may switch to channel w_l if the following inequality holds true

$$\frac{2-\epsilon_U}{\hat{K}_{w_l}+1}U^{[w_l]}(\hat{K}_{w_l}+1) + \frac{\epsilon_U - 1}{\hat{K}_{w_l}+2}U^{[w_l]}(\hat{K}_{w_l}+2) > \frac{1}{\hat{K}_w}U^{[w]}(\hat{K}_w). \tag{5.4}$$

The left-hand-side of (5.4) is an estimated per user utility if channel w_l has ϵ_U more users. The estimation is obtained using a linear interpretation approach because ϵ_U is not integer-valued. The criterion of (5.4) basically states that, in order for user k to consider switching to channel w_l, the per user utility of channel w_l should be better than that of channel w even if channel w_l adds ϵ_U more users. Note that the criteria of (5.3) and (5.4) are similar in principle. The difference is that, while channel load is measured by the number of users in (5.3), it is measured by per user utility in (5.4). We believe that convergence properties of the extended distributed MAC algorithms under channel switching criteria of (5.3) and (5.4) should not be fundamentally different. Therefore, with heterogeneous channels and under mild conditions, we still expect that the system should converge to a state with roughly balanced channel loads.

5.2 Simulation

In this section, we use two examples to show performance of MAC algorithm for multiple channels and its convergence behavior.

Example 1: In this example, we consider a multiple access system with two non-interfering collision channels. Each user is equipped with a single transmission option. The transmission of a packet in a channel is successful if and only if no other user is transmitting in the same channel. Virtual packets for both channels are assumed to be equivalent to a real packet. Users intend to maximize the symmetric system throughput. Each user designs functions $p^{*[w]}(\hat{K}) = \frac{1}{\hat{K}+1.01}$ and $q_v^{*[w]}(\hat{K}) = (1 - \frac{1}{\hat{K}+1.01})^{\hat{K}}$ for $w = 1, 2$. We assume that there is no measurement noise and no feedback error. Users follow the proposed MAC algorithm for channel assignment and portability adaptation. Figure 5.1 depicts distribution of number of users in each channel as a function of total number of users in the system. It can be seen that the number of users in the two channels are either equal or different only by one.

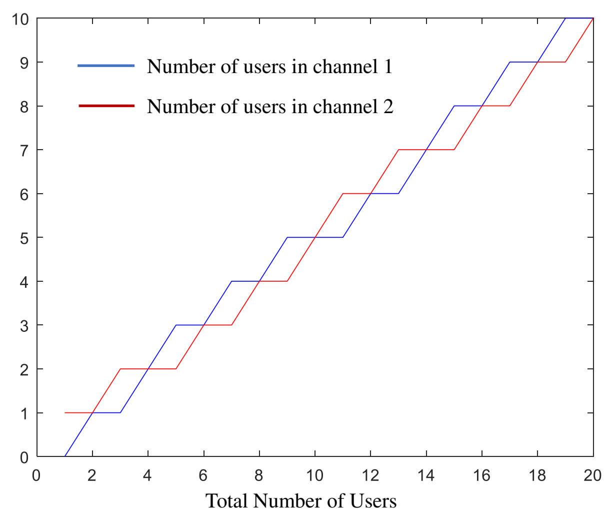

Figure 5.1: Number of users in each channel as a function of total number of users in the system

Now assume there are 8 users in the system. First, each user randomly assigns itself to one channel and initializes its probability. The receiver measures $q_v^{[w]}$ in each channel using the following exponential moving average approach. Let $I_v^{[w]}$ be an indicator that shows success/ failure status of the virtual packet in channel w. The receiver first sets $q_v^{[w]} = 1$, then updates it by $q_v^{[w]} = (1 - \frac{1}{300})q_v^{[w]} + \frac{1}{300}I_v^{[w]}$. Users adapt the proposed MAC algorithm with a constant step-size of $\alpha(t) = 0.05$ and with $p_{sw} = 0.5$. We use similar exponential moving average approaches to calculate the throughput and the average transmission probability of each channel. Figure 5.2, shows throughput in each channel. As can be seen from the figure, throughput of the two channels

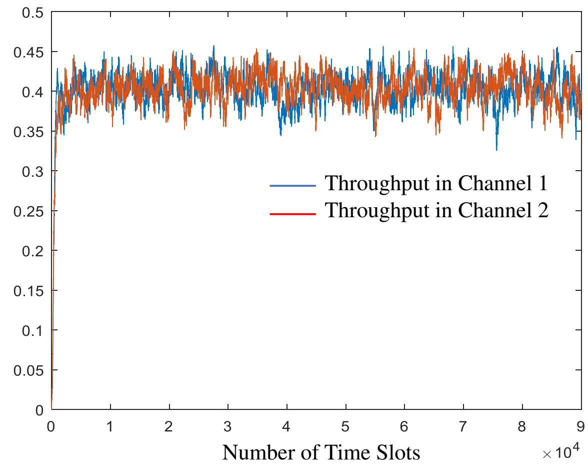

Figure 5.2: Throughput in two collision channels

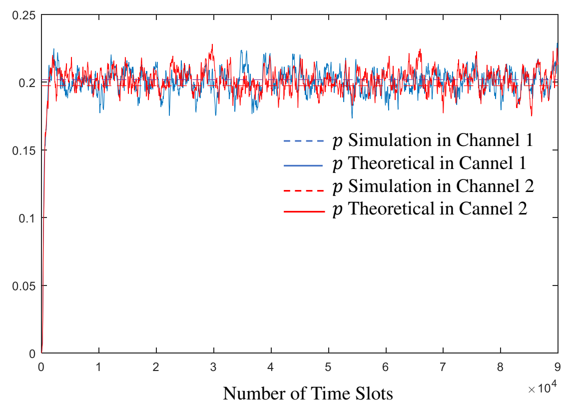

Figure 5.3: Probability convergence behavior in two collision channels

are almost identical. Figure 5.3 illustrates the convergence behavior of transmission probabilities in both channels with their theoretical values. Theoretical values for both channels are the same and the actual probabilities of the users are close to the theoretical values. Figures 5.1, 5.2 and 5.3 show the proposed MAC algorithm can lead the system to a state with balanced channel loads.

Example 2: Consider a multiple access system with one collision channel and one multi-packet reception channel. For the multi-packet reception channel, we assume that each user is equipped with two transmission options, one being labeled as the high rate option and one being labeled as the low rate option. If all packets are encoded using the low-rate option, the multi-packet reception channel can support the parallel transmissions of 12 packets. We assume that each packet encoded using the high-rate option is equivalent to the combination of 4 low-rate packets. In other words, the multi-packet reception channel can support the parallel transmissions of n_1 high-rate packets and n_2 low-rate packets if and only if $\frac{1}{3}n_1 + \frac{1}{12}n_2 \leq 1$ is satisfied. For the collision channel, we assume that each user is equipped with a single transmission option that has the same rate parameter as the high-rate option of the multi-packet reception channel. For both channels, each virtual packet is assumed to be equivalent to a high-rate packet.

Assume that users intend to maximize the symmetric sum system throughput. For the collision channel, users design $p^{*[1]}(\hat{K})$ and $q_v^{*[1]}(\hat{K})$ as explained in Example 1. For the multi-packet reception channel, each user sets $\boldsymbol{p}^{*[2]}(\hat{K}) = \frac{2.27}{2+1.01}[1,0]^T$ for $\hat{K} \leq 2$, $\boldsymbol{p}^{*[2]}(\hat{K}) = \frac{2.27}{\hat{K}+1.01}[1,0]^T$ for $2 < \hat{K} \leq 4$ and $\boldsymbol{p}^{*[2]}(\hat{K}) = \frac{8.82}{\hat{K}+1.01}[0,1]^T$ for $\hat{K} \geq 10$. $q_v^{*[2]}(\hat{K})$ then can be calculated correspondingly. To design $\boldsymbol{p}^{*[2]}(\hat{K})$ for $4 \leq \hat{K} \leq 10$, we choose two pinpoints at $\hat{K} = 5$ and $\hat{K} = 6$. We set direction vectors of the pinpoints at the direction vector corresponds to the optimal scheme. Then, we use linear interpolation to determine direction vectors for other points of $4 < \hat{K} \leq 10$. Finally, we set the transmission probability function $\boldsymbol{p}^{*}(\hat{K})$ for $4 < \hat{K} \leq 10$ such that the resulting $q_v^{*[2]}(\hat{K})$ is linear in $\hat{K}$.

We assume that the system goes through three stages. First, there are 5 users in the system. At the Stage two, 8 more users join the system. Then at Stage three, 10 users leave the system. Each user assigns itself to a channel randomly and users initialize their probability vectors at zero vectors. The receiver measures $q_v^{[w]}$ in each channel using an exponential moving average approach. Let $I_v^{[w]}$ be an indicator of the success/failure status of the virtual packet in channel w. The receiver first sets $q_v^{[w]} = 1$, then updates it by $q_v^{[w]} = (1 - \frac{1}{300})q_v^{[w]} + \frac{1}{300}I_v^{[w]}$. Users adapt the proposed MAC algorithm with a constant step-size of $\alpha(t) = 0.05$ and with $p_{sw} = 0.5$. Figure 5.4 shows the throughput achieved in two channels, and figure 5.5 shows per user throughput achieved in two channels. In figure 5.6 and 5.7 convergence behavior of the entries of the probability vectors and their theoretical values in the collision channel and the multi-packet reception channel are depicted respectively. It can be seen that the entries of the probabilities vectors are close to the corresponding theoretical values. Users can also adapt quickly to the new equilibrium as the number of users change in the system. Furthermore, figures 5.4, 5.6, 5.5 and 5.7 also show the proposed MAC algorithm achieved the objective of balanced channel loads.

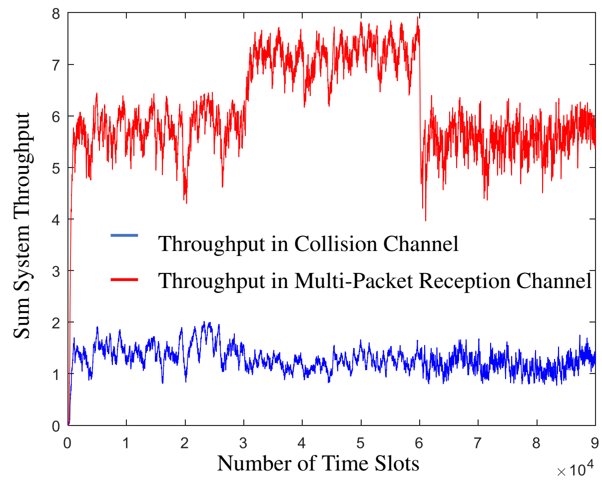

Figure 5.4: Throughput in collision channel and multi-packet reception channel

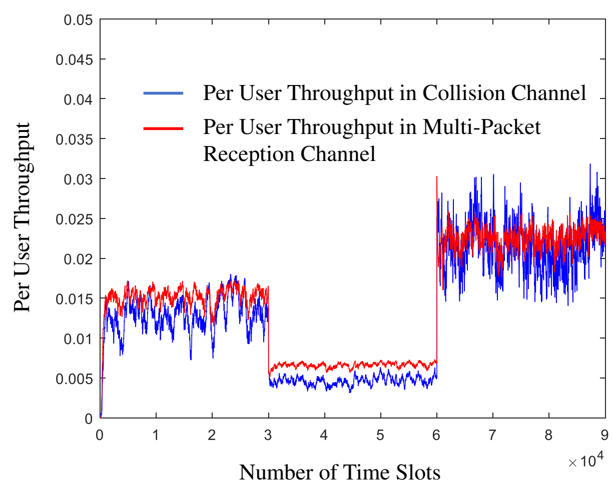

Figure 5.5: Per user throughput in collision channel and multi-packet reception channel

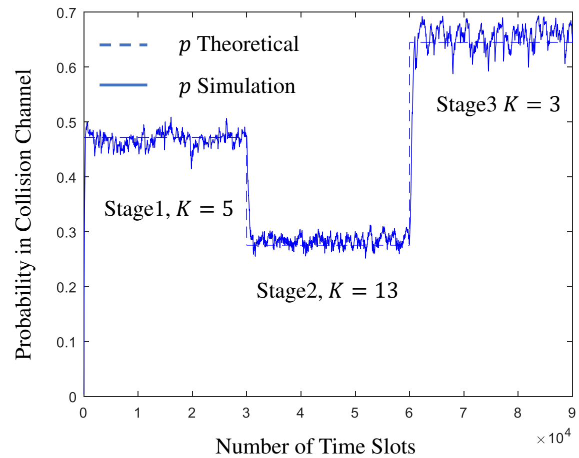

Figure 5.6: Transmission probability in collision channel

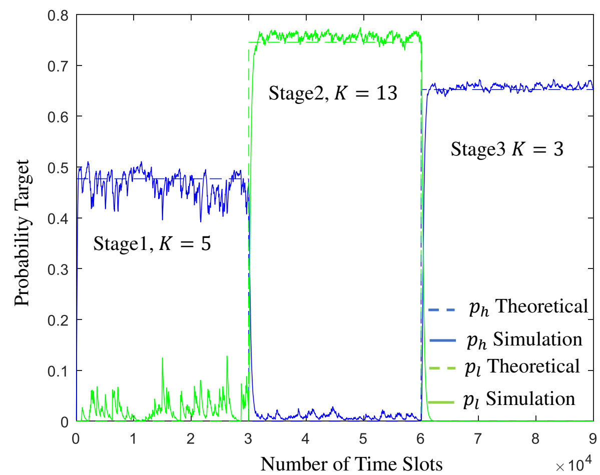

Figure 5.7: Entries of transmission probability vector in multi-packet reception channel

Chapter 6

Conclusion

In this dissertation, we introduced and investigated an enhanced physical-link layer interface. Classical physical-link layer interface only allows a link layer user either to transmit a packet or to idle. Other communication details are determined at the physical layer. However, because wireless traffic is increasingly bursty and fragmental, coordinating a large number of users in such an environment can become expensive or infeasible in terms of overhead. It is possible that full communication optimization cannot be done at the physical layer, and therefore it is necessary to support advanced communication adaptation at the data link layer. Objective of the physical-link layer interface enhancement is to enable advanced wireless capabilities such as power and rate adaptation at the data link layer.

At the physical layer, the enhanced physical-link layer interface is supported by distributed channel coding theory reviewed in Chapter 2. The channel coding theory allows each physical layer user to be equipped with an ensemble of channel codes. Each code corresponds to a transmission option at the data link layer. Each physical layer user with available message arbitrarily chooses one code to encode its message and sends it through the channel. The receiver knows the code ensembles of the users but not their particular coding choices. The receiver decodes the message of interest if a pre-determined error probability requirement can be met and reports collision otherwise. We extended the definition of "communication error" from the classical meaning of erroneous decoding to the new meaning of failure in reporting the expected outcome, and with packet collision being added as an expected outcome under certain conditions. With this new definition a "distributed channel capacity" is defined and characterized. The new capacity result coincides with Shannon information region in a sense explained in [4].

At the data link layer, with the enhanced interface, each user is equipped with multiple transmission options each of them representing a particular combination of communication parameters such as transmission power and communication rate. Motivated by questions such as how a user with multiple transmission options should respond to the feedback of success/failure status of a packet, in Chapter 3, we proposed a distributed MAC algorithm with/without enhanced interface. The proposed algorithm is applicable to a general channel model and a wide range of utility functions. We analyzed the equilibrium of the proposed MAC algorithm based on a stochastic approximation framework. As explained in Chapter 3, the key challenge is to design the theoretical channel contention measure function to be monotonic in the estimated number of users while satisfying the Lipschitz continuity condition defined in Condition 2. For the case of single transmission option, a closed-form theoretical channel contention measure function is designed. For the case of multiple transmission options, a search-assisted approach is suggested to design the monotonic channel contention measure function. With the desired monotonicity property, we proved uniqueness of the equilibrium of the proposed algorithm. Simulation results are provided to illustrate the performance and the convergence behavior of the proposed algorithm. Although our design approach does not provide optimality guarantee to the system equilibrium, simulation results demonstrated that obtaining a near optimal equilibrium is often not difficult.

In Chapter 4, we proposed a distributed MAC algorithm to support a hierarchical primary-secondary user structure in a random multiple access system. Assume that the numbers of primary and secondary users in the system are unknown to the users. Without requiring users to identify whether a transmission belongs to a primary or a secondary user, without requiring direct information exchange between any users, the proposed algorithm establishes the hierarchical structure in the following senses. First, when the number of primary users is small, the algorithm guarantees transmission success probability of primary users to stay above a pre-determined threshold despite of the number of secondary users competing for the channel. Second, when the number of primary users is large, the algorithm automatically drives transmission probabilities of the secondary users down to zero. As explained in Chapter 4, such a hierarchical structure is achieved by raising the tail of the theoretical channel contention measure function of secondary users, such that the aggregated impact of secondary users on channel contention is kept below a given threshold. We proved uniqueness of the equilibrium for systems with users being equipped with only a single transmission option. Although we could not prove equilibrium uniqueness for systems with multiple transmission options, we believe this should be the case under mild conditions, and this is supported by simulation results.

In Chapter 5, we investigated distributed MAC approach for a random multiple access system with multiple non-interfering channels. While we assume that users should be homogeneous, we assume that channels can be homogeneous or heterogeneous. We showed that enforcing homogeneous users to converge to the same transmission scheme can be suboptimal. Therefore a distributed MAC algorithm should guide users to heterogeneous transmission schemes to avoid overly crowding any channel. We assume that time is partitioned into long frames. We developed a distributed algorithm to assign each user to one channel at the beginning of each time frame. Such assignments are carried out randomly and adaptively toward the objective of achieving a balanced channel loads. Within each time frame, channel assignments of the users are fixed. Users assigned to the same channel use distributed MAC algorithm presented in Chapter 3 to adapt their transmission probability vectors toward the common unique equilibrium. We provided an argument to support the potential convergence proof of the developed algorithm, although a strict theoretical proof is not yet available. Simulation results showed that the proposed algorithm can maximize a symmetric network utility and can also achieve the objective of balancing the loads of different channels.

Bibliography

[1] J. T. T. Cover, *Elements of Information Theory*, 2nd. Wiley Interscience, 2005.

[2] H. Zimmermann, "Osi reference model-the osi model of architecture for open system interconnection", *IEEE Transaction on Communications*, vol. 28, pp. 425–432, Apr. 1980.

[3] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function", *IEEE journal on Selected Areas in Communication*, vol. 18, pp. 535–547, Mar. 2000.

[4] J. Luo, "A generalized channel coding theory for distributed communication", *IEEE Transaction on Information Theory*, vol. 63, pp. 1043–1056, Apr. 2015.

[5] J. Luo and A. Ephremides, "A new approach to random access: Reliable communication and reliable collision detection", *IEEE Transaction on Information Theory*, vol. 58, pp. 379–423, Feb. 2012.

[6] Z. Wang and J. Luo, "Error performance of channel coding in random access communication", *IEEE Transaction on Information Theory*, vol. 58, pp. 3961–3974, Jun. 2012.

[7] Y. Tang, F. Heydaryan, and J. Luo, "Distributed capacity of a multiple access channel", *IEEE WCNC*, Apr. 2019.

[8] ——, "Distributed coding in a multiple access environment", *Foundations and Trends in Networking*, vol. 12, pp. 260–412, 2018.

[9] N. Abramson, "The aloha system - another alternative for computer communication", *Proc. AFIPS Conf*, vol. 37, pp. 281–285, Nov. 1970.

[10] R. Gallager, "Conflict resolution in random access broadcast network", *AFOSR Workshop on Commun. Theory App*, pp. 74–76, Sep. 1978.

[11] J. Capetanakis, "Tree algorithms for packets broadcast channel", *IEEE Transaction on Information Theory*, vol. 25, pp. 505–515, Sep. 1979.

[12] B. Hajek and T. Loon, "Decentralized dynamic control of a multiaccess broadcast channel", *IEEE Transaction on Automatic Control*, vol. 27, pp. 559–569, Jun. 1982.

[13] B. Hajek, “Stochastic approximation methods for decentralized control of multiaccess communications”, *IEEE Transaction on Information Theory*, vol. IT-31, pp. 176–184, Mar. 1985.

[14] F. Heydaryan, Y. Tang, and J. Luo, “Distributed multiple access with multiple transmission options at the link layer”, *submitted to Ad Hoc Networks*,

[15] R. Ahlswede, “Multi-way communication channels”, *IEEE ISIT*,

[16] D. Bertsekas and R. Gallager, *Data Network*, 2nd. Prentice Hall, 1992.

[17] J. Luo and A. Ephremides, “On the throughput, capacity and stability regions of random multiple access”, *IEEE Transaction on Information Theory*, vol. 52, pp. 2593–2607, Jun. 2006.

[18] S. Ghez and S. Verd, *Stochastic Approximation Algorithms and Applications*. New York: Springer-Verlag, 1997.

[19] C. Lau and C. Leung, “Capture models for model packet radio networks”, *IEEE Transaction on Communication*, vol. 40, pp. 917–925, May 1992.

[20] S. Ghez and S. Verd, “Stability properties of slotted aloha with multipacket reception capability”, *IEEE Transaction on Automatic Control*, vol. 33, pp. 640–649, Jul. 1988.

[21] Y. Tang, F. Heydaryan, and J. Luo, “Distributed multiple access with a general link layer channel”, *Ad Hoc Networks*, vol. 85, pp. 120–130, Mar. 2019.

[22] ——, “Distributed mac over a general multi-packet reception channel”, *WiOPT*, Jun. 2019.

[23] A. Ephremides and B. Hajek, “Information theory and communication networks: An unconsummated union”, *IEEE Transaction on Information Theory*, vol. 44, pp. 2416–2434, Oct. 1998.

[24] R. Rao and A. Ephremides, “On the stability of interacting queues in a multiple-access system”, *IEEE Transaction on Information Theory*, vol. 34, pp. 918–930, Sep. 1988.

[25] S. Karlin and H. Taylor, *A First Course in Stochastic Processes*, 2ndl. San Diego, CA: Academic, 1975.

[26] V. Borkar and S. Meyn, "The o.d.e method for convergence of stochastic approximation and reinforcement learning", *SIAM Journal on Control and Optimization*, vol. 38, pp. 447–469, Jan. 2000.

[27] I. C. S. L. M. S. Committee, "Ieee standard for information technology–local and metropolitan area networks–specific requirements–part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications - amendment 8: Medium access control (mac) quality of service enhancements", *IEEE Std 802.11 - 2005*, 2005.

[28] A. Domenico, E. Strinati, and M. Benedetto, "A survey on mac strategies for cognitive radio networks", *IEEE Communication Survey Tutorials*, vol. 14, pp. 21–44, First Quarter 2012.

[29] M. Khoshkholgh, K. Navaie, and H. Yanikomeroglu, "Access strategies for spectrum sharing in fading environment: Overlay, underlay, and mixed", *IEEE Transaction on Mobile Computing*, vol. 9, pp. 1780–1793, Dec. 2010.

[30] F. Heydaryan and J. Luo, "Random multiple access with hierarchical users", *submitted to IEEE Transaction on Communications*,

[31] J. G. Kemeny and J. L. Snell, *Finite Markov Chains*, 2nd. Springer, Jul. 1976.

Appendix A

Proofs of Theorems in Chapter 3

A.1 Proof of Theorem 6

Proof. According to Step 3 of the distributed MAC algorithm, users should always have the same target transmission probability vectors. At any equilibrium, we should have transmission probability vectors of all users equal $\boldsymbol{p}^*(\hat{K})$ for some $\hat{K}$, which must satisfy $q_v(\boldsymbol{p}^*(\hat{K}), K) = q_v^*(\hat{K})$. According to Assumption 5, if $K \geq K_{\min}$, we must have $\hat{K} = K$. If $K < K_{\min}$, on the other hand, according to Assumption 3, for all $\hat{K} > K_{\min}$, we have

$$q_v^*(\hat{K}) < q_v^*(K_{\min}) = q_v(\boldsymbol{p}^*(K_{\min}), K_{\min}) \leq q_v(\boldsymbol{p}^*(K_{\min}), K). \tag{A.1}$$

Consequently, transmission probability vectors of all users at equilibrium must equal $\boldsymbol{p}^*(K_{\min})$, which equals $\boldsymbol{p}^*(K)$ according to Assumption 5. Therefore, the system should always have a unique equilibrium at $\boldsymbol{P}^* = \mathbf{1} \otimes \boldsymbol{p}^*(K)$.

Given the number of users K. Target transmission probability vector $\hat{\boldsymbol{p}}$ obtained in Step 3 of the distributed MAC algorithm can be written as a function of the transmission probability vectors of all users $\boldsymbol{P}$ as $\hat{\boldsymbol{p}}(\boldsymbol{P}) = \hat{\boldsymbol{p}}(q_v(\boldsymbol{P}, K))$. Let $\boldsymbol{P}_a$, $\boldsymbol{P}_b$ be two arbitrary transmission probability vectors of all users. According to Assumption 4 and Theorem 5, we have

$$\|\hat{\boldsymbol{p}}(\boldsymbol{P}_a) - \hat{\boldsymbol{p}}(\boldsymbol{P}_b)\| \leq K_{qp}|q_v(\boldsymbol{P}_a, K) - q_v(\boldsymbol{P}_b, K)| \leq K_{qc}K_{qp}\|\boldsymbol{P}_a - \boldsymbol{P}_b\|. \tag{A.2}$$

Therefore, the Lipschitz Continuity Condition 2 is satisfied.

Finally, when the system is noisy, the receiver can choose to measure q_v over an extended number of time slots, or equivalently, to increase the value of Q introduced in Step 2 of the proposed MAC algorithm. If users maintain their transmission probability vectors during the Q time slots, it is often the case that the potential measurement bias in the system can be reduced arbitrarily close to zero. Therefore, the Mean and Bias Condition 1 is also satisfied. $\square$

A.2 Proof of Theorem 8

Proof. First, it is easy to see that Assumption 2 is satisfied with $q_v^*(\infty)$ being equal to the limiting theoretical channel contention measure of the "Tail" regime, and $\boldsymbol{p}^*(\infty) = p^*(\infty)\boldsymbol{d}^*(\overline{K})$, where $p^*(\infty)$ is the limiting theoretical transmission probability of the "Tail" regime.

Second, in the "Head" regime when $\hat{K} \leq \underline{K}$, because $\underline{K} \geq J_{\epsilon_v}(\boldsymbol{d}^*(\underline{K}))$, $q_v^*(\hat{K})$ is strictly decreasing for $J_{\epsilon_v}(\boldsymbol{d}^*(\underline{K})) \leq \hat{K} \leq \underline{K}$, and $\boldsymbol{p}^*(\hat{K})) = p_{\max}\boldsymbol{d}^*(\underline{K})$ remains a constant vector for $\hat{K} \leq J_{\epsilon_v}(\boldsymbol{d}^*(\underline{K}))$. In other words, we should define $K_{\min} = J_{\epsilon_v}(\boldsymbol{d}^*(\underline{K}))$. Furthermore, $q_v^*(\hat{K})$ is strictly decreasing for $\underline{K} \leq \hat{K} \leq \overline{K}$ by design. Because $\overline{K} > J_{\epsilon_v}(\boldsymbol{d}^*(\overline{K}))$, $q_v^*(\hat{K})$ is also strictly decreasing for $\hat{K} \geq \overline{K}$ in the "Tail" regime. Therefore, Assumption 3 is satisfied.

Third, according to [21, Theorem 4], Assumption 4 should be satisfied in the "Head and Tail" regimes. In other words, target transmission probability vector $\hat{\boldsymbol{p}}(q_v)$ as a function of q_v is Lipschitz continuous in q_v for $q_v \geq q_v^*(\underline{K})$ and $q_v \leq q_v^*(\overline{K})$.

Next, we will prove that the theoretical transmission probability vector function $\boldsymbol{p}^*(\hat{K}) = p^*(\hat{K})\boldsymbol{d}^*(\hat{K})$ is Lipschitz continuous in $\hat{K}$ for $\underline{K} \leq \hat{K} \leq \overline{K}$. Because $\boldsymbol{d}^*(\hat{K})$ is continuous by design, the objective is to show that the search-assisted approach does not lead to any discontinuity of $p^*(\hat{K})$ in $\hat{K}$. For the sake of simple notation, we use $\frac{dp^*(\hat{K})}{d\hat{K}}$ to represent the derivative of $p^*(\hat{K})$ with respect to $\hat{K}$ if $p^*(\hat{K})$ is differentiable. If $p^*(\hat{K})$ is only continuous but not differentiable at $\hat{K}$, then $\frac{dp^*(\hat{K})}{d\hat{K}}$ represents one or an arbitrary subderivative of $p^*(\hat{K})$. If $p^*(\hat{K})$ is not continuous at $\hat{K}$, then $\frac{dp^*(\hat{K})}{d\hat{K}}$ should take the values of $\pm\infty$. Note that the adoption of such a notation does not imply a continuity assumption on $p^*(\hat{K})$. Our objective then becomes to prove that $\frac{dp^*(\hat{K})}{d\hat{K}}$ is bounded for $\underline{K} \leq \hat{K} \leq \overline{K}$.

Let $i \in \{1, \ldots, L\}$ and $0 \leq \lambda < 1$ be chosen arbitrarily. Let $\hat{K} = \hat{K}_{i\lambda}$, where $\hat{K}_{i\lambda}$ is defined in (3.26). To simplify the discussion, we assume that the neighboring two pinpoints satisfy

$\hat{K}_{i+1} = \hat{K}_i + 1$, i.e., they take neighboring integer values[8]. Write $\hat{K} = \hat{K}_{i\lambda} = (1-\lambda)\hat{K}_i + \lambda\hat{K}_{i+1}$ as a function of λ, we have $\frac{dp^*(\hat{K})}{d\hat{K}} = \frac{dp^*(\lambda)}{d\lambda}$.

To bound $\frac{dp^*(\lambda)}{d\lambda}$, we consider two different expressions of $q_v^*(\hat{K}) = q_v^*(\lambda)$. The first expression is

$$q_v^*(\lambda) = (1-\lambda)q_v^*(\hat{K}_i) + \lambda q_v^*(\hat{K}_{i+1}). \tag{A.3}$$

Take derivative with respect to λ, we get $\frac{dq_v^*(\lambda)}{d\lambda} = q_v^*(\hat{K}_{i+1}) - q_v^*(\hat{K}_i)$. Because both $q_v^*(\hat{K}_{i+1})$ and $q_v^*(\hat{K}_i)$ are bounded, there exists a positive constant $\overline{\Delta}_1 > 0$ such that

$$\left|\frac{dq_v^*(\lambda)}{d\lambda}\right| \leq \overline{\Delta}_1. \tag{A.4}$$

On the other hand, define $p_{i\lambda}^* = p^*(\hat{K}_{i\lambda})$, and consider the second expression of $q_v^*(\hat{K}) = q_v^*(\lambda)$ given below.

$$q_v^*(\lambda, p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*) = (1-\lambda)q_v(p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*, \hat{K}_i) + \lambda q_v(p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*, \hat{K}_{i+1}). \tag{A.5}$$

Taking derivative with respect to λ results in

$$\frac{dq_v^*(\lambda, p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*)}{d\lambda} = \frac{\partial q_v^*(\lambda, p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*)}{\partial\lambda} + \left[\frac{\partial q_v^*(\lambda, p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*)}{\partial\boldsymbol{d}_{i\lambda}^*}\right]^T \frac{d\boldsymbol{d}_{i\lambda}^*}{d\lambda} + \frac{\partial q_v^*(\lambda, p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*)}{\partial p_{i\lambda}^*}\frac{dp_{i\lambda}^*}{d\lambda}. \tag{A.6}$$

Now we consider the terms on the right hand side of (A.6) separately.

$$\frac{\partial q_v^*(\lambda, p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*)}{\partial\lambda} = q_v(p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*, \hat{K}_{i+1}) - q_v(p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*, \hat{K}_i). \tag{A.7}$$

Because both two terms on the right hand side of (A.7) are bounded, there exists a constant $\overline{\Delta}_2 > 0$ to satisfy

$$\left|\frac{\partial q_v^*(\lambda, p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*)}{\partial\lambda}\right| \leq \overline{\Delta}_2. \tag{A.8}$$

According to (3.13), we can write $q_v(p_{i\lambda}^*\boldsymbol{d}_{i\lambda}^*, \hat{K}_i)$ as

[8]The proof can be easily extended to the case when this assumption does not hold.

$$q_v(p^*_{i\lambda}\boldsymbol{d}^*_{i\lambda}, \hat{K}_i) = \sum_{j=0}^{\hat{K}_i} \binom{\hat{K}_i}{j} p^{*j}_{i\lambda}(1-p^*_{i\lambda})^{\hat{K}_i-j} C_{vj}(\boldsymbol{d}^*_{i\lambda}). \tag{A.9}$$

Due to Assumption 1, the right hand side of (A.9) contains no more than $K_c + 1$ terms. Because $\frac{\partial C_{vj}(\boldsymbol{d}^*_{i\lambda})}{\partial \boldsymbol{d}^*_{i\lambda}}$ is bounded for all j, $\frac{\partial q_v(p^*_{i\lambda}\boldsymbol{d}^*_{i\lambda},\hat{K}_i)}{\partial \boldsymbol{d}^*_{i\lambda}}$ must be bounded. Similarly, $\frac{\partial q_v(p^*_{i\lambda}\boldsymbol{d}^*_{i\lambda},\hat{K}_{i+1})}{\partial \boldsymbol{d}^*_{i\lambda}}$ is also bounded. Therefore, from (A.5), we can see there exists a constant $\overline{\Delta}_3 > 0$ such that

$$\left| \left[\frac{\partial q^*_v(\lambda, p_{i\lambda}\boldsymbol{d}_{i\lambda})}{\partial \boldsymbol{d}_{i\lambda}} \right]^T \frac{d\boldsymbol{d}_{i\lambda}}{d\lambda} \right| \leq \overline{\Delta}_3. \tag{A.10}$$

From (A.9), by taking partial derivative with respect to $p^*_{i\lambda}$, we get

$$\begin{aligned}
\frac{\partial q_v(p^*_{i\lambda}\boldsymbol{d}^*_{i\lambda}, \hat{K}_i)}{\partial p^*_{i\lambda}} &= \sum_{j=0}^{\hat{K}_i} \binom{\hat{K}_i}{j} j p^{*(j-1)}_{i\lambda}(1-p^*_{i\lambda})^{\hat{K}_i-j} C_{vj}(\boldsymbol{d}^*_{i\lambda}) \\
&- \sum_{j=0}^{\hat{K}_i} \binom{\hat{K}_i}{j} (\hat{K}_i - j) p^{*j}_{i\lambda}(1-p^*_{i\lambda})^{\hat{K}_i-j-1} C_{vj}(\boldsymbol{d}^*_{i\lambda}) \\
&= \sum_{j=0}^{\hat{K}_i-1} \binom{\hat{K}_i}{j} \hat{K}_i p^{*j}_{i\lambda}(1-p^*_{i\lambda})^{\hat{K}_i-j-1} (C_{v(j+1)}(\boldsymbol{d}^*_{i\lambda}) - C_{vj}(\boldsymbol{d}^*_{i\lambda})).
\end{aligned} \tag{A.11}$$

Due to Item 2 of the Pinpoints Condition 4, $\hat{K}_i > J_{\epsilon_v}(\boldsymbol{d}^*_{i\lambda})$. Therefore, $C_{vj}(\boldsymbol{d}^*_{i\lambda}) - C_{v(j+1)}(\boldsymbol{d}^*_{i\lambda}) \geq \epsilon_v$ should hold for at least one $0 \leq j \leq \hat{K}_i - 1$. Due to Item 3 of Condition 4, $\underline{p} \leq p^*_{i\lambda} \leq \overline{p}$. Hence $\left| \frac{\partial q_v(p^*_{i\lambda}\boldsymbol{d}^*_{i\lambda},\hat{K}_i)}{\partial p^*_{i\lambda}} \right|$ is bounded away from zero. The same conclusion applies to $\left| \frac{\partial q_v(p^*_{i\lambda}\boldsymbol{d}^*_{i\lambda},\hat{K}_{i+1})}{\partial p^*_{i\lambda}} \right|$. Because both $\frac{\partial q_v(p^*_{i\lambda}\boldsymbol{d}^*_{i\lambda},\hat{K}_i)}{\partial p^*_{i\lambda}}$ and $\frac{\partial q_v(p^*_{i\lambda}\boldsymbol{d}^*_{i\lambda},\hat{K}_{i+1})}{\partial p^*_{i\lambda}}$ are negative-valued, from (A.5), we can see there exists a positive constant $\underline{\Delta}_1 > 0$ such that

$$\left| \frac{\partial q^*_v(\lambda, p_{i\lambda}\boldsymbol{d}_{i\lambda})}{\partial p_{i\lambda}} \right| \geq \underline{\Delta}_1. \tag{A.12}$$

Because the two expressions of $q^*_v(\hat{K})$ given in (A.3) and (A.5) must equal each other, by combining (A.4), (A.6), (A.8), (A.10), and (A.12), we conclude that there exists a positive constant $K_g > 0$, such that $\left\| \frac{d\boldsymbol{p}^*(\hat{K})}{d\hat{K}} \right\| \leq K_g$. With the extended definition of $\frac{d\boldsymbol{p}^*(\hat{K})}{d\hat{K}}$, as explained at the beginning of the proof, $\left\| \frac{d\boldsymbol{p}^*(\hat{K})}{d\hat{K}} \right\| \leq K_g$ means that $\boldsymbol{p}^*(\hat{K})$ is Lipschitz continuous in $\hat{K}$.

According to Item 1 of the Pinpoints Condition 4, for any $\hat{K}_a, \hat{K}_b \in [\underline{K}, \overline{K}]$, we have

$$|q_v^*(\hat{K}_a) - q_v^*(\hat{K}_b)| \geq \frac{\epsilon_q}{\overline{K} - \underline{K}} |\hat{K}_a - \hat{K}_b|. \tag{A.13}$$

This means that, for $\hat{K}$ being obtained according to Step 3 of the distributed MAC algorithm, $\hat{K}(q_v)$ as a function of q_v is Lipschitz continuous in q_v for $q_v^*(\overline{K}) \leq q_v \leq q_v^*(\underline{K})$. Because we just proved that $\boldsymbol{p}^*(\hat{K})$ is Lipschitz continuous in $\hat{K}$, we conclude that the target transmission probability vector $\hat{\boldsymbol{p}}(q_v)$ obtained according to Step 3 of the distributed MAC algorithm is Lipschitz continuous in q_v for $q_v^*(\overline{K}) \leq q_v \leq q_v^*(\underline{K})$. Combined with Lipschitz continuity of $\hat{\boldsymbol{p}}(q_v)$ in the "Head and Tail" regimes, we can see that Assumption 4 is also satisfied.

Fourth, because $q_v^*(\hat{K})$ is strictly decreasing in $\hat{K}$ for $\hat{K} \geq J_{\epsilon_v}(\boldsymbol{d}^*(\underline{K}))$, it is easy to prove that, if $K \leq K_{\min} = J_{\epsilon_v}(\boldsymbol{d}^*(\underline{K}))$, then $q_v^*(\hat{K}) = q_v(\boldsymbol{p}^*(\hat{K}), K)$ should hold for all $\hat{K} \leq K_{\min}$, and for $K_{\min} \leq K \leq \underline{K}$ and $K \geq \overline{K}$, $q_v^*(\hat{K}) = q_v(\boldsymbol{p}^*(\hat{K}), K)$ should have a unique solution at $\hat{K} = K$.

Now consider the case when $\underline{K} \leq K \leq \overline{K}$. With users setting their transmission probability vectors at $\boldsymbol{p}^*(\hat{K})$, because $\hat{K} > J_{\epsilon_v}(\boldsymbol{d}^*(\hat{K}))$ and $\underline{p} \leq p^*(\hat{K}) \leq \overline{p}$, if $K > \hat{K}$ and $\hat{K}$ is an integer, we must have

$$q_v(\boldsymbol{p}^*(\hat{K}), K) < q_v(\boldsymbol{p}^*(\hat{K}), \hat{K}) = q_v^*(\hat{K}). \tag{A.14}$$

If $K > \hat{K}$ and $\hat{K}$ is not an integer, we have

$$\begin{aligned} &q_v(\boldsymbol{p}^*(\hat{K}), K) < q_v(\boldsymbol{p}^*(\hat{K}), \lfloor \hat{K} \rfloor), \\ &q_v(\boldsymbol{p}^*(\hat{K}), K) \leq q_v(\boldsymbol{p}^*(\hat{K}), \lfloor \hat{K} \rfloor + 1), \end{aligned} \tag{A.15}$$

which implies that

$$q_v(\boldsymbol{p}^*(\hat{K}), K) < q_v^*(\hat{K}). \tag{A.16}$$

On the other hand, if $K < \hat{K}$ and $\hat{K}$ is an integer, we must have

$$q_v(\boldsymbol{p}^*(\hat{K}), K) > q_v(\boldsymbol{p}^*(\hat{K}), \hat{K}) = q_v^*(\hat{K}). \tag{A.17}$$

If $K < \hat{K}$ and $\hat{K}$ is not an integer, we have

$$q_v(\boldsymbol{p}^*(\hat{K}), K) > q_v(\boldsymbol{p}^*(\hat{K}), \lfloor \hat{K} \rfloor + 1),$$
$$q_v(\boldsymbol{p}^*(\hat{K}), K) \geq q_v(\boldsymbol{p}^*(\hat{K}), \lfloor \hat{K} \rfloor), \tag{A.18}$$

which also implies that

$$q_v(\boldsymbol{p}^*(\hat{K}), K) > q_v^*(\hat{K}). \tag{A.19}$$

Consequently, $q_v^*(\hat{K}) = q_v(\boldsymbol{p}^*(\hat{K}), K)$ must have a unique solution at $\hat{K} = K$. Therefore, Assumption 5 should hold true.

$\square$

Appendix B

Proofs of Theorems in Chapter 4

B.1 Proof of Theorem 9

Proof. According to the stochastic approximation framework presented in [8, Section 4.1], the system should have at least one equilibrium. Assume that the system contains two equilibria, whose corresponding channel contention measures equal q_v and $\tilde{q}_v$, respectively. Without loss of generality, we assume that $q_v < \tilde{q}_v$.

Assume that, at the first equilibrium corresponding to channel contention measure q_v, the number of users estimated by the primary users and by the secondary users equal respectively $\hat{K}_p$ and $\hat{K}_s$. At the other equilibrium corresponding to channel contention measure $\tilde{q}_v$, the estimates equal $\tilde{K}_p$ and $\tilde{K}_s$, respectively. Consequently, we have

$$\begin{aligned} q_v &= q_{vp}^*(\hat{K}_p) = q_{vs}^*(\hat{K}_s), \\ \tilde{q}_v &= q_{vp}^*(\tilde{K}_p) = q_{vs}^*(\tilde{K}_s). \end{aligned} \tag{B.1}$$

Because $q_v < \tilde{q}_v$, due to the fact that $q_{vp}^*(\hat{K})$ and $q_{vs}^*(\hat{K})$ functions are strictly decreasing in $\hat{K}$ [8, Theorem 4.4], (B.1) implies that $\hat{K}_p \geq \tilde{\hat{K}}_p$ and $\hat{K}_s \geq \tilde{\hat{K}}_s$. This consequently implies that $p_p^*(\hat{K}_p) \leq p_p^*(\tilde{K}_p)$ and $p_s^*(\hat{K}_p) \leq p_s^*(\tilde{K}_p)$. However, if each user at the first equilibrium should transmit at a probability no higher than the corresponding probability at the other equilibrium, we must have $q_v \geq \tilde{q}_v$, which contradicts the assumption that $q_v < \tilde{q}_v$. Therefore, equilibrium of the system must be unique.

Let q_v be the channel contention measure at the unique equilibrium. We will prove the following equivalent statement. That is, if $q_v < q_{vs}^*(\infty)$, we must have $q_{vp}^*(K_p) = q_v < q_{vs}^*(\infty)$. Otherwise if $q_v \geq q_{vs}^*(\infty)$, we must have $q_{vp}^*(K_p) \geq q_{vs}^*(\infty)$.

According to the proposed MAC algorithm, if $q_v < q_{vs}^*(\infty)$, we should have $\hat{K}_s = \infty$ and all secondary users should have zero transmission probability at the equilibrium. Consequently, the system becomes equivalent to one with homogeneous (primary) users, as analyzed in [8]. According to [8, Theorem 4.5], we should have

$$q_v = q_v(p_p^*(K_p), K_p) = q_{vp}^*(K_p). \tag{B.2}$$

In other words, primary users should obtain correct user number estimate. This implies that $q_{vp}^*(K_p) = q_v < q_{vs}^*(\infty)$.

If $q_v \geq q_{vs}^*(\infty)$, on the other hand, we have $q_v = q_{vp}^*(\hat{K}_p)$. In this case, $\hat{K}_s < \infty$, meaning that secondary users should transmit with a positive probability. Because if all secondary users exit the system, channel contention measure of the system should converge to $q_v(p_p^*(K_p), K_p) = q_{vp}^*(K_p)$, due to monotonicity of the $q_v(p, K_p)$ function, we must have

$$\begin{aligned} & q_{vs}^*(\infty) \leq q_v = q_v(p_p^*(\hat{K}_p), K_p) \\ & \leq q_v(p_p^*(K_p), K_p) = q_{vp}^*(K_p). \end{aligned} \tag{B.3}$$

□