On Random Access Control for Multipacket Reception S-ALOHA in Wireless Cellular Networks

by

Jun-Bae Seo

B.Sc., Korea University, 2001
M.Sc., Korea University, 2003

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

in

The Faculty of Graduate Studies

(Electrical and Computer Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA
(Vancouver)

June 2012

© Jun-Bae Seo, 2012
Abstract

Studies on slotted ALOHA (S-ALOHA) systems have a long research history and been quite mature in the literature. However, this thesis revisits S-ALOHA systems by considering multipacket reception (MPR) channels, because previous studies are no longer applicable to MPR capable S-ALOHA systems due to the new random access channels (RACHs) of contemporary wireless cellular network standards. Particularly, it is dubious to apply the contention resolution algorithms (CRAs) developed so far for S-ALOHA systems to MPR S-ALOHA systems without any modification. Accordingly, this thesis proposes cross-layer CRAs and investigates the performance of some existing CRAs in MPR S-ALOHA systems in order to maximize the system throughput in tandem with stabilizing the system.

Cross-layer CRAs proposed are based on estimating the system backlog information, which is obtained from multiple access interference (MAI) in the physical layer of Code Division Multiple Access (CDMA) MPR channel. Then, the proposed algorithms broadcast a retransmission probability in order to maximize the system throughput. The performance and the stability of the proposed algorithms are examined under various radio channel conditions.

Compared to the proposed algorithms based on the system backlog information, the existing uniform backoff (UB) and exponential backoff (EB) algorithms are blind to such information. In order to understand the behaviour of these algorithms in MPR S-ALOHA, they are investigated with respect to various performance metrics and stability conditions. Additionally, the power ramping (PR) scheme is examined together with UB algorithm,
Abstract

in order to see whether it shows power capture effect in MPR S-ALOHA systems during a random access. Furthermore, the queueing performance of the MPR S-ALOHA system is investigated with an uplink traffic of the Markov Modulated Bernoulli Process (MMBP), when UB algorithm with retry limit is employed. It is also examined whether MPR S-ALOHA would be feasible for reserving a channel for a delay sensitive traffic. As a practical example, Semi-Persistent Scheduling (SPS) with initial random access in Long Term Evolution (LTE) is examined over MPR S-ALOHA, in which a traffic channel reservation for Voice-over IP (VoIP) is made at the onset of the ON period by a random access.
Preface

This thesis consists of both submitted and published works. I am the first author and principal contributor of all manuscript chapters. All chapters are co-authored with Dr. Victor C. M. Leung, who supervises the present thesis research.

Journal Papers Accepted/Published


- Jun-Bae Seo and Victor C. M. Leung, “Queuing Performance of Multichannel S-ALOHA Systems with Correlated Arrivals,” *IEEE Transactions on Vehicular Technology*, 60(9): 4574-4586, Nov. 2011: This paper can be found in chapter 5.
Journal Papers Submitted

- Jun-Bae Seo and Victor C. M. Leung, “Performance Modeling and Stability of Semi-Persistent Scheduling with Initial Random Access in UMTS-LTE,” under revision: This paper can be found in chapter 6.


Conference Papers Published


# Table of Contents

Abstract ................................................................. ii

Preface ................................................................. iv

Table of Contents ...................................................... vii

List of Tables .......................................................... xi

List of Figures .......................................................... xii

List of Abbreviations .................................................... xiv

Acknowledgments ......................................................... xvii

Dedication .............................................................. xviii

1 Introduction .......................................................... 1

1.1 MPR S-ALOHA ..................................................... 3

1.2 Random Access in S-ALOHA Systems ......................... 4

1.2.1 Access Control Algorithms ................................. 4

1.2.2 Stability of S-ALOHA ....................................... 5

1.2.3 Fairness ......................................................... 12

1.3 Contributions ...................................................... 12

1.4 Thesis Organization ............................................... 14
# Table of Contents

## 2 Cross Layer Contention Resolution Algorithms

2.1 System Model ................................. 18
2.2 Algorithm Design ............................. 21
2.3 Algorithm Evaluation ........................ 27
   2.3.1 Performance Measures .................. 27
   2.3.2 System Stability ....................... 28
2.4 Numerical Results and Discussions ....... 29
2.5 Summary ...................................... 34

## 3 Existing Backoff Algorithms for Single-Buffered Terminals

3.1 Backoff Algorithms .......................... 39
   3.1.1 System Model ............................ 39
   3.1.2 Uniform Backoff Algorithm in LTE .... 39
   3.1.3 Binary Exponential Backoff Algorithm in IEEE 802.16 .... 40
   3.1.4 Access Prioritization by Persistence Value .......... 41
3.2 Performance Evaluation ..................... 41
   3.2.1 Channel Model ............................ 42
   3.2.2 Approximation on Correlated Backoff Processes .... 44
   3.2.3 Uniform Backoff Algorithm ............... 44
   3.2.4 Binary Exponential Backoff Algorithm .......... 49
   3.2.5 Stability .................................. 52
   3.2.6 Access Prioritization Scheme ............... 55
   3.2.7 Dynamic Window Assignment Algorithm .......... 58
3.3 Numerical Studies ........................... 60
3.4 Summary ...................................... 68
# Table of Contents

4 Backoff Algorithm with Power Ramping in LTE ....................................... 70
   4.1 System Model : Retransmission Backoff with Power Ramping ............... 73
   4.2 Performance Evaluation ........................................................................ 75
      4.2.1 Performance Metrics ................................................................... 75
      4.2.2 Correlation Coefficient on Backoff Process ................................. 79
      4.2.3 Terminal-Assisted UB Algorithm (TAUBA) .................................. 80
   4.3 Numerical Studies ............................................................................... 83
   4.4 Summary ............................................................................................. 88

5 Queueing Performance of Multichannel ALOHA Systems ......................... 89
   5.1 System Model ..................................................................................... 93
      5.1.1 Uplink Traffic Source Model ....................................................... 93
      5.1.2 Access Protocol and Backoff Algorithm ...................................... 96
   5.2 Queueing Performance Evaluation ...................................................... 97
      5.2.1 Approximation on Correlated Multi-queue .................................. 97
      5.2.2 Multi-buffered Terminals ............................................................. 98
      5.2.3 Single-buffered Terminals ........................................................... 107
   5.3 Numerical Studies .............................................................................. 109
   5.4 Summary ............................................................................................. 117

6 Semi-Persistent Scheduling in LTE ......................................................... 118
   6.1 Models and Assumptions .................................................................... 123
      6.1.1 Random Access for SPS Algorithm ........................................... 123
      6.1.2 Uplink Traffic Model ................................................................... 126
      6.1.3 Semi-Persistent Scheduling in LTE ............................................. 127
   6.2 Analysis ............................................................................................... 130
      6.2.1 Equilibrium Point Analysis ......................................................... 130
# Table of Contents

6.2.2 Packet Dropping Probability ............................................. 137  
6.3 Numerical Results and Discussions ........................................ 140  
6.4 Summary ........................................................................... 146  

7 Conclusions and Future Work .................................................... 148  
7.1 Research Contributions ....................................................... 148  
7.2 Suggestions for Future Work ................................................. 152  

Bibliography .............................................................................. 154
# List of Tables

3.1 Nomenclature for analysis .................................................. 42
3.2 Nomenclature for DWA algorithm ........................................ 56
4.1 Throughput gain $\tau_p$ with $\beta$ ........................................ 79
5.1 Performance metrics with various backoff window sizes. ............. 116
6.1 Symbols used in SPS algorithm with initial random access ............ 127
6.2 Stabilized system performance with various parameters ............... 142
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Instability of S-ALOHA with Bernoulli arrival.</td>
</tr>
<tr>
<td>1.2</td>
<td>Instability of S-ALOHA with Poisson arrival.</td>
</tr>
<tr>
<td>1.3</td>
<td>Stability region of S-ALOHA with multi-buffered terminals.</td>
</tr>
<tr>
<td>1.4</td>
<td>Organization of this thesis.</td>
</tr>
<tr>
<td>2.1</td>
<td>System performance under perfect power control.</td>
</tr>
<tr>
<td>2.2</td>
<td>System throughput under perfect power control.</td>
</tr>
<tr>
<td>2.3</td>
<td>System stability by negative drift.</td>
</tr>
<tr>
<td>2.4</td>
<td>Comparison of uncoded and BCH-coded system under imperfect power control.</td>
</tr>
<tr>
<td>2.5</td>
<td>Performance of BCH-coded system under imperfect power control, $\sigma_X^2 = 1.5$ dB.</td>
</tr>
<tr>
<td>3.1</td>
<td>UB algorithm with different $M$ and $P$.</td>
</tr>
<tr>
<td>3.2</td>
<td>Comparison of UB and BEB algorithms with different $U$ and $L$.</td>
</tr>
<tr>
<td>3.3</td>
<td>Performance of BEB algorithm with different $K$, $L$ and $M$.</td>
</tr>
<tr>
<td>3.4</td>
<td>Access prioritization scheme of UB algorithm with differential $p$ and $U$.</td>
</tr>
<tr>
<td>3.5</td>
<td>Access prioritization scheme of UB algorithm with differential $p$.</td>
</tr>
<tr>
<td>3.6</td>
<td>Access prioritization scheme of BEB algorithm with differential $p$ and $K$.</td>
</tr>
<tr>
<td>3.7</td>
<td>Performance comparison of UB algorithm with/without DWA algorithm.</td>
</tr>
<tr>
<td>4.1</td>
<td>The effect of power ramping and $U$ on the system performance.</td>
</tr>
</tbody>
</table>
List of Figures

4.2 The effect of the initial transmission power selection on the system performance. ................................................. 84
4.3 The correlation coefficient of $M$ terminals’ backoff process. ................................................................. 85
4.4 Performance of TAUBA against some static UB algorithms. ................................................................. 86
5.1 An uplink traffic model of Web-browsing. ........................................................................................................ 91
5.2 Effect of multichannel on multi-buffered system, $p_{00} = 0.85, p_{11} = 0.65,$ $U = 25, J = 3, K = 10.$ ................................................................................. 106
5.3 Performance comparisons of multi-buffered systems with various source correlation, $p_{11} = 0.6, U = 20, K = 10.$ ................................................................................. 108
5.4 Performance comparisons of single- (bufferless) and multi-buffered systems, $p_{00} = 0.75, p_{11} = 0.6, U = 20, J = 6, K = 1.$ ................................................................................. 111
5.5 Performance comparisons of multi-buffered systems with different retry limit $L,$ $p_{00} = 0.85, p_{11} = 0.65, U = 20, K = 10.$ ................................................................................. 113
5.6 Performance comparisons of multi-buffered systems with different window size $U,$ $p_{00} = 0.85, p_{11} = 0.65, L = 2, K = 10.$ ................................................................................. 115
6.1 Signal flow of SPS with SRI or with random access. ......................................................................................... 119
6.2 State transition diagram of terminals with SPS algorithm with random access. ........................................ 129
6.3 Bistability of the system: Equilibrium function $F(c_0)$ and the state probability for $c_0, \pi_{c_0}$ .................................................. 141
6.4 Comparison of PRMA and SPS with random access ......................................................................................... 143
6.5 The effect of $K$ on the system performance. ................................................................................................. 145
6.6 The effect of $U$ on the system performance. ................................................................................................. 146
# List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACK</td>
<td>Acknowledgement</td>
</tr>
<tr>
<td>AMR</td>
<td>Adaptive Multi-rate</td>
</tr>
<tr>
<td>BCH</td>
<td>Bose and Ray-Chaudhuri</td>
</tr>
<tr>
<td>BEB</td>
<td>Binary Exponential Backoff</td>
</tr>
<tr>
<td>BS</td>
<td>Base Station</td>
</tr>
<tr>
<td>CDMA</td>
<td>Code Division Multiple Access</td>
</tr>
<tr>
<td>CRA</td>
<td>Contention Resolution Algorithm</td>
</tr>
<tr>
<td>CSMA/CA</td>
<td>Carrier Sense Multiple Access - Collision Avoidance</td>
</tr>
<tr>
<td>DFT</td>
<td>Delayed First Transmission</td>
</tr>
<tr>
<td>DWA</td>
<td>Dynamic Window Assignment</td>
</tr>
<tr>
<td>eNodeB</td>
<td>evolved NodeB</td>
</tr>
<tr>
<td>EPA</td>
<td>Equilibrium Point Analysis</td>
</tr>
<tr>
<td>H-ARQ</td>
<td>Hybrid Automatic Repeat Request</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hyper-Text Transfer Protocol</td>
</tr>
<tr>
<td>IFT</td>
<td>Immediate First Transmission</td>
</tr>
<tr>
<td>i.i.d.</td>
<td>(statistically) independent, identically distributed</td>
</tr>
<tr>
<td>FDD</td>
<td>Frequency Division Duplexing</td>
</tr>
<tr>
<td>LTE</td>
<td>Long Term Evolution</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>MAI</td>
<td>Multiple Access Interference</td>
</tr>
<tr>
<td>MCS</td>
<td>Modulation and Coding Scheme</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multi-input Multi-output</td>
</tr>
<tr>
<td>MMBP</td>
<td>Markov Modulated Bernoulli Process</td>
</tr>
<tr>
<td>MMPP</td>
<td>Markov Modulated Poisson Process</td>
</tr>
<tr>
<td>MPR</td>
<td>Multi-packet Reception</td>
</tr>
<tr>
<td>OFDMA</td>
<td>Orthogonal Frequency Division Multiple Access</td>
</tr>
<tr>
<td>PMF</td>
<td>Probability Mass Function</td>
</tr>
<tr>
<td>PN</td>
<td>Pseudo-Noise</td>
</tr>
<tr>
<td>POMDP</td>
<td>Partially Observable Markov Decision Process</td>
</tr>
<tr>
<td>PR</td>
<td>Power Ramping</td>
</tr>
<tr>
<td>PRACH</td>
<td>Physical Random Access Channel</td>
</tr>
<tr>
<td>PUSCH</td>
<td>Physical Uplink Shared Channel</td>
</tr>
<tr>
<td>PUCCH</td>
<td>Physical Uplink Common Control Channel</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>RACH</td>
<td>Random Access Channel</td>
</tr>
<tr>
<td>RAP</td>
<td>Random Access Preamble</td>
</tr>
<tr>
<td>RAR</td>
<td>Random Access Response</td>
</tr>
<tr>
<td>SDMA</td>
<td>Space Division Multiple Access</td>
</tr>
<tr>
<td>SINR</td>
<td>Signal-to-Interference plus Noise Ratio</td>
</tr>
<tr>
<td>SPS</td>
<td>Semi-Persistent Scheduling</td>
</tr>
<tr>
<td>SR</td>
<td>Scheduling Request</td>
</tr>
<tr>
<td>SRI</td>
<td>Scheduling Request Indicator</td>
</tr>
<tr>
<td>TAUBA</td>
<td>Terminal-Assisted Uniform Backoff Algorithm</td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission Control Protocol</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>TDD</td>
<td>Time Division Duplexing</td>
</tr>
<tr>
<td>UB</td>
<td>Uniform Backoff</td>
</tr>
<tr>
<td>UMTS</td>
<td>Universal Mobile Telecommunication Systems</td>
</tr>
<tr>
<td>UTRA</td>
<td>UMTS Terrestrial Radio Access</td>
</tr>
<tr>
<td>VoIP</td>
<td>Voice-over IP</td>
</tr>
<tr>
<td>WiMAX</td>
<td>Worldwide Interoperability for Microwave Access</td>
</tr>
<tr>
<td>WLAN</td>
<td>Wireless Local Area Network</td>
</tr>
</tbody>
</table>
Acknowledgments

First and foremost, I would like to thank my advisor, Professor Victor Leung, for his invaluable advice, warm encouragement and guidance, and enormous patience. He has improved the quality of my research and presentation by his constructive comments from the early stages of my research until the very end.

I would like to thank Professor Vincent Wong for his advice and also the members of my doctoral committee, Dr. Lutz Lampe, Dr. Vikram Krishnamurthy, Dr. Sathish Gopalakrishnan, Dr. Norman Hutchinson, Dr. Jane Wang, Dr. Julian Cheng and Dr. Xianbin Wang for their advice and time. I would like to thank my undergraduate advisor Professor Hyong-Woo Lee for his guidance to graduate studies. I also want to thank my fellow labmates and friends at the Communications lab, the Department of Electrical and Computer Engineering at the University of British Columbia who helped me along the way, especially Mr. Haoming Li, Mr. Seyedali Hosseininezhad, Dr. Vahid Shah-Mansouri, Dr. Hu Jin, Dr. Jun-Su Kim and Dr. Seok Woo. Further, I would like to thank Dr. Ki-Dong Lee at LGE for his advice. I thank my friends at the University of British Columbia, Mr. Mohit Law, Mr. Dan-Sik Yoo, Mr. Tae-Sik Chae, Mr. Sang-Hoon Yeo, and Mrs. Jacqueline Stewart. Most importantly, I would like to thank my parents for their love and support. I thank God for providing me with everything along this way. This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the University of British Columbia Graduate Fellowship.
Dedication

With love, to my parents.
Chapter 1

Introduction

In the past few years, wireless cellular network standards have evolved into the Third Generation Partnership Project (3GPP) Long-Term Evolution (LTE) and IEEE 802.16 Worldwide Interoperability for Microwave Access (WiMAX) systems. This is done by replacing code division multiple access (CDMA) with orthogonal frequency division multiple access (OFDMA). Furthermore, both systems have integrated various technologies into the physical layer of such OFDMA systems, such as adaptive modulation and coding scheme (MCS), and multi-input/multi-output (MIMO) antennas. These technological paradigm shifts not only increase wireless capacity, but also enable multipacket reception (MPR) capable random access channels (RACHs). Thus, it comes under question whether previous work on a single channel slotted ALOHA (S-ALOHA) is still valid for MPR capable S-ALOHA (simply MPR S-ALOHA). While the stability region of MPR S-ALOHA systems has been discussed in [1][2], the question still remains open how we can achieve the maximum stable throughput over an MPR S-ALOHA channel by providing system stability, or how efficiently some existing contention resolution algorithms (CRAs) used in a single channel S-ALOHA can work in MPR S-ALOHA. Thus, this thesis examines MPR S-ALOHA systems by investigating the following research questions in each chapter.

1. Given an MPR S-ALOHA system, can we construct a CRA with objectives of maximizing throughput (approaching optimal) in tandem with stabilizing the system? (Chapter 2)
Chapter 1. Introduction

2. In contrast to some CRAs proposed in the literature, window-based uniform backoff (UB) or binary exponential backoff (BEB) algorithms have been employed in contemporary communication standards where an MPR S-ALOHA is embedded. Then, how well do these algorithms work compared to the optimum performance? What are the stability conditions of these existing algorithms? (Chapter 3)

3. In LTE, a power ramping (PR) scheme is incorporated into UB algorithm. In Chapters 3-5, we ignore PR scheme by assuming that it helps to adjust the transmission power as an open loop power control that does not affect the system performance. This chapter will examine how effective the PR scheme can be, by showing a throughput gain per additional transmission power consumption by the PR scheme in comparison with throughput by non-PR scheme. We also propose a terminal-assisted uniform backoff algorithm (TAUBA) which controls the window size adaptively in the presence of PR scheme. (Chapter 4)

4. Queueing performance of S-ALOHA has been studied over past decades with respect to information-theoretic and queueing-theoretic perspectives. How does a terminal’s queueing process behave in an MPR S-ALOHA system? (Chapter 5)

5. MPR S-ALOHA is used for establishing (or reserving) a traffic or control channel in LTE. It is examined whether it would be feasible to employ MPR S-ALOHA to establish a Voice-over IP (VoIP) traffic channel by means of Semi-Persistent Scheduling (SPS) in LTE without violating delay constraint of VoIP. (Chapter 6)

The next subsections introduce MPR S-ALOHA and the CRAs developed for a single channel S-ALOHA systems, respectively. Survey on the CRAs for MPR S-ALOHA will be given in Chapter 2.
Chapter 1. Introduction

1.1 MPR S-ALOHA

MPR channels have been characterized in [1] as the probability that \( k \) among \( n \) packets simultaneously transmitted are successfully received, which is denoted by \( C_{k,n} \) for \( k \leq n \) and \( \sum_{k=0}^{n} C_{k,n} = 1 \). MPR channels can be found in the following systems such as, the conventional CDMA systems, space division multiple access (SDMA) systems [3], and systems employing successive (or parallel) interference cancellation [4], iterative decoding or Hybrid Automatic Repeat reQuest (H-ARQ) [5], wherever multiuser detection is available in a RACH.

With the current standards, such as LTE and IEEE 802.16 systems, an MPR channel is found in the RACH of each system. Each base station (BS) in the systems has a finite number of orthogonal random access preambles (RAPs) like pseudo-noise (PN) codes of CDMA. When terminals perform random access based on S-ALOHA, they randomly select one of the RAPs and transmit it to one of the OFDMA channels which can also be randomly selected. Upon receiving a unique RAP, the BS allocates an uplink channel to the terminal that selected it. Here, the RACH acts as a CDMA-OFDMA channel. Furthermore, if multiple terminals choose the same RAP, for the terminals that are successfully detected, the same uplink channel is allocated. This is called a code-collision in this thesis. However, LTE allows those terminals to transmit their messages (called \( \text{Msg3} \) [6]) on the same uplink channel, by expecting that the collision of these messages would be resolved by H-ARQ. In LTE, random access is said to be successfully completed when terminals receive a contention resolution message after successfully transmitting \( \text{Msg3} \). Therefore, another MPR channel is found in LTE, in which collisions are resolved by H-ARQ.
1.2 Random Access in S-ALOHA Systems

1.2.1 Access Control Algorithms

This section briefly introduces some well-known CRAs (i.e., access control algorithm, or backoff algorithm) for S-ALOHA system in the literature. In general, CRAs can be classified by whether the algorithm utilizes the system backlog information (by estimation) or not. Many researchers have developed CRA algorithms. The most well known and highly documented are Lam and Kleinrock [7], Hajek and van Loon [8], Kelly [9], and Rivest’s Bayesian broadcasting algorithm [10]. The maximum achievable throughput of these algorithms is \( e^{-1} = 0.368 \). A summary on these algorithms [7]-[9] can be found in [10]. One drawback of these algorithms is that they operate in a centralized way. It is noticeable that the centralized algorithms can be regarded as a Partially Observable Markov Decision Process (POMDP), in the sense that they make a control policy over a Markovian backlog process after observing the channel outcome.

As a distributed algorithm, tree algorithms have been proposed by Capentanakis [11], Tsybakov and Mikhailov [12], and Hayes [13]. These algorithms and their variants are well documented in [14][15][16]. Especially, capacity bound and maximum achievable throughput of various tree algorithms in the literature are chronologically well summarized in [16]. One drawback of these algorithms is that newly arriving packets should be aware of the end of an ongoing contention resolution interval, i.e., synchronization issue. Maximum achievable throughput of a feasible tree algorithm so far known is 0.487 in [17][18]. Finally, there are window-based UB or BEB algorithms which will be described in Chapter 3. These algorithms are most practical in the sense that they work in a distributed manner and are easy to implement for various well-known systems, such as IEEE 802.3 Ethernet, IEEE 802.11 Wireless Local Area Network (WLAN), Universal Mobile Telecommunications Sys-
tem (UMTS), LTE, IEEE 802.16 and IEEE 802.22. The maximum achievable throughput of these window-based (stabilizing) backoff algorithms is bounded by 0.368. In the next subsection, we briefly introduce the stability of S-ALOHA systems.

1.2.2 Stability of S-ALOHA

This section provides a simple introduction on the stability of S-ALOHA systems. More details are found in the references given at the end of each subsection. Single-buffered terminals indicate that terminals can store only one packet in (re)transmission, while multi-buffered terminals have finite queue to store incoming packets awaiting channel access. Note that the stability of BEB algorithm has been well summarized in [19].

Stability of the System with Single-Buffered Terminals

In general, for the system with single-buffered terminals, bistability is found under the finite population model, while instability may occur under an infinite population model. In what follows, we assume that a newly generated packet is immediately transmitted, while backlogged packets are transmitted with probability $p$.

Finite Population (Bernoulli Arrival): We illustrate the bistability of S-ALOHA systems with Bernoulli arrivals of newly generated packets [7][20]. Consider an S-ALOHA system, in which $M$ terminals generate a packet with probability $\sigma$ and transmit it with probability one in a slot. If the newly generated packet transmission is not successful, then (backlogged) terminals retransmit the packet with probability $p$. Then, the throughput of this system given $n$ backlogged terminals is expressed by

$$S_\sigma(n) = (M - n)\sigma(1 - \sigma)^{M-1}(1 - p)^n + (1 - \sigma)^{M-n}np(1 - p)^{n-1}, \quad (1.1)$$
Chapter 1. Introduction

Figure 1.1: Instability of S-ALOHA with Bernoulli arrival.

in which the first term shows the event that only one newly generated packet is transmitted, while \( n \) backlogged terminals do not retransmit their packet, and vice versa in the second term. The input rate of newly generated packets to this system given \( n \) backlogged terminals can be expressed as

\[
S_i(n) = (M - n)\sigma. \tag{1.2}
\]

Define an equilibrium point of this system as a point \( n^* \) which satisfies

\[
S_i(n^*) = S_o(n^*). \tag{1.3}
\]

Fig. 1.1 illustrates \( S_i(n) \) and \( S_o(n) \) of the system with \( \sigma = 0.003, p = 0.06, \) and \( M = 100, \) in which three points of \( n^* \) are found. The system showing this behaviour is often called \textit{bistable}, in which the throughput-delay performance at locally stable equilibrium point can be achieved only for some finite time period. If the system has a unique \( n^* \), it is called a stable system. Given \( M \) and \( \sigma \), it is interesting to drive this system to have only one
equilibrium point by controlling $p$.

**Infinite Population (Poisson Arrival):** Now suppose an S-ALOHA system in which new packet arrival rate follows a Poisson distribution with rate $\lambda$. Let $\pi_n$ denote the stationary probability that the S-ALOHA system has $n$ backlogged terminals. As before, newly generated packets will be transmitted with probability one. Thus, when at least two packets are transmitted, a collision occurs. Then $\pi_n$ satisfies the following balance equation:

$$
\pi_n = \pi_n (1 - p)^n a_0 + \pi_n \left( 1 - \sum_{k=0}^{1} \binom{n}{k} p^j (1 - p)^{n-k} \right) a_0 + \pi_n (1 - p)^n a_0 + \pi_{n+1} (1 - p)^n a_0 + \pi_{n-1} (1 - (1 - p)^{n-1}) a_1 + \sum_{k=2}^{n} \pi_{n-k} a_k,
$$

in which we have $a_k = \lambda^k \exp^{-\lambda} / k!$ and $\pi_n = 0$ for $n < 0$. Finally, the normalization condition $\sum_{n=0}^{\infty} \pi_n = 1$ holds. As shown in [14][21], the ratio, $\pi_{n+1}/\pi_n$, in (1.4) grows without bound as $n \to \infty$. This implies that S-ALOHA system with infinite population model is always unstable. This can be qualitatively explained as follows [14][15]. Denote

![Figure 1.2: Instability of S-ALOHA with Poisson arrival.](image)

by $G$ the expected number of new ($\lambda$) and retransmitted packets (per second). Then, the
throughput of this S-ALOHA system is predicted by $S_o = Ge^{-G}$. As we have seen in Fig. 1.1, we also depict $S_o$ and $S_i = \lambda$ in Fig. 1.2. By small perturbation on the actual traffic rate around $G_1$, the system throughput moves around $\lambda = G_1 e^{-G_1}$. If a large perturbation causes the actual traffic to exceed $G_2$ (which will occur with probability one), then $S_o$ decreases below $\lambda$. Once it happens, the system never returns to the point $(\lambda, G_1)$, which means instability.

**Pakes’ Lemma (Foster’s theorem)**

We have seen previously bistability and instability of S-ALOHA system with finite and infinite population model, respectively. We now connect the previous two examples into Foster’s theorem (extended by Pakes’ lemma [22]) for an ergodic Markov chain [22][23]. If the following two conditions hold, it is sufficient for an irreducible, aperiodic homogeneous Markov chain (whose state space is the set of nonnegative integers) to be ergodic:

1. $|E[B_{t+1} - B_t | B_t = m]| < \infty$ for $\forall m$,
2. $\lim_{m \to \infty} \sup_{B_t = m} E[B_{t+1} - B_t | B_t = m] < 0$, \hspace{1cm} (1.5)

in which $B_t$ denotes the number of backlogged terminals at time $t$ in S-ALOHA systems. We denote the second condition by $d(m)$. In order for an S-ALOHA system with finite population model not to have bistability, there exists only one $m^*$, which shows $d(m) < 0$ for $m^* < m \leq M$. If this would be found for a stable system with infinite population model, i.e., $M \to \infty$, then the system backlog is bounded. More specifically, we can write
Chapter 1. Introduction

for infinite population model (given $B_t = m$)

$$B_{t+1} = \begin{cases} 
  m - 1 & \text{with probability } mp(1-p)^{m-1}a_0 \\
  m & \text{with probability } (1-p)^m a_1 + (1-p(1-p)^{m-1})a_0 \\
  m + 1 & \text{with probability } (1-(1-p)^m) a_1 \\
  m + k & \text{with probability } a_k \text{ for } k \geq 2.
\end{cases}$$

According to (1.5), we have

$$E[B_{t+1} - B_t | B_t = m] = E[B_{t+1} | B_t = m] - E[B_t | B_t = m]$$

$$= \left\{ (m-1)mp(1-p)^{m-1}a_0 + m(1-p)^m a_1 + (1-p(1-p)^{m-1})a_0 \\
+ (m+1)(1-(1-p)^m)a_1 + \sum_{k=2}^{\infty}(m+k)a_k \right\} - m$$

$$= \lambda - mp(1-p)^{m-1}a_0 - (1-p)^m a_1.$$

Then, we should have $m^\ast$ for $\lambda < m^\ast p(1-p)^{m^\ast-1}a_0 + (1-p)^{m^\ast}a_1$ in order to stabilize the system.

**Stability of the System with Multi-Buffered Terminals**

In contrast with S-ALOHA systems with single-buffered terminals, we assume $M$ terminals, each of which has a buffer of infinite size [2][24]. Denote by $Q^t_i$ the queue length process of the $i$-th terminal at time $t$. Then, a multidimensional stochastic process $Q^t = [Q^t_1, \ldots, Q^t_M]$ is defined as stable [24], if we have

$$\lim_{t \to \infty} \Pr[Q^t < x] = F(x) \quad \text{and} \quad \lim_{x \to \infty} F(x) = 1,$$  \hfill (1.6)
in which \( \mathbf{x} \) is a vector of length \( M \). So far, bistability in S-ALOHA system with multi-buffered terminals has not been discussed in the literature. We shall discuss this later in Chapter 5. Exact queueing analysis on this multidimensional queueing process in S-ALOHA system\(^1\) is found only in [2][26] where two terminals are considered in S-ALOHA and MPR S-ALOHA systems, respectively. All other works on queueing study for S-ALOHA systems and CSMA systems including WLAN in the literature are an approximation on this queueing process. In addition, it has been shown that this queueing process, \( Q_t' \), is positively correlated [27]. This implies that terminals’ queue length could keep increasing, once they are congested, or vice versa. In order to achieve (1.6), each terminal’s queueing process in an S-ALOHA system should satisfy Loynes’ theorem [28], which means that the packet arrival rate to each terminal should be less than each terminal’s throughput, i.e., service rate. In [29] Rao and Ephremeides consider the maximum acceptable packet arrival rate in S-ALOHA systems, which maintains the stability, by introducing a dominant system where terminals with empty queue keep transmitting a dummy packet in order to interfere with other terminals with non-empty queue. As a result, if \( M \) terminals have identical new packet arrival rate (Poisson arrival with mean rate \( \lambda \)) and retransmission probability \( (p) \), it is shown that the maximum achievable throughput of this S-ALOHA system can be expressed as \( \lambda < p(1-p)^{M-1} \). As an example, we illustrate this in Fig. 1.3 for \( M = 20 \) and \( \lambda = 0.012 \). Note that this system can be stabilized for \( p_1 < p < p_2 \). More details are found in [2][29].

**Stability in MPR S-ALOHA Systems in This Thesis**

Throughout this thesis we assume that new packets are generated based on Bernoulli trials in the system with single-buffered and multi-buffered terminals under a finite population

\(^1\)For CSMA with collision detection, one can find queueing analysis in [20][25]. However, no queueing analysis is found for carrier-sense multiple access (CSMA) even with two terminals.
model. Thus, the stability condition in an MPR S-ALOHA system with single-buffered terminals in Chapters 2, 3, and 4 is considered as a single equilibrium point, as we have introduced. In Chapter 5, we consider a multi-channel S-ALOHA system, in which \( M \) terminals have a buffer of finite size. Due to the assumption on finite queue length, (1.6) always holds. Furthermore, we consider retry limit, i.e., limiting the number of retransmissions of a packet in MPR S-ALOHA systems as in the standards such as LTE [6], WiMAX [30], UMTS [31], and WLAN [32]. Since packets are dropped due to such a retry limit, the maximum stable throughput of our interest can be defined as the maximum number of acceptable terminals in conjunction with mean packet arrival rate, while a predefined packet dropping probability is not violated. In addition, for the positive correlation of terminals’ backoff processes or terminals’ queueing processes, our theoretical evaluations on MPR S-ALOHA systems in Chapters 3-5 are based on statistically independent, identically distributed (i.i.d.) assumption on terminals’ backoff processes or queueing processes. We shall introduce the details on this approximation in Section 3.2.2, Section 4.2.2, and Section 5.2.1.

Figure 1.3: Stability region of S-ALOHA with multi-buffered terminals.
1.2.3 Fairness

Fairness provision in random access control algorithms has been discussed mostly in mobile ad-hoc networks [33][34][35] with respect to proportional fairness of transmission rate based on utility optimization. In this thesis the fairness of an access algorithm is evaluated by the inverse of the variance of access delay. Suppose an access control algorithm, with which terminals experience finite mean and zero variance of access delay. Then, terminals using such an access algorithm experience constant access delay, once a packet is generated. Accordingly, in Chapter 3 we shall examine how UB and BEB algorithms behave in the second-order moment of access delay.

1.3 Contributions

This thesis proposes novel CRAs and investigates the performance of some existing CRAs in MPR capable ALOHA. The contributions of each chapter is summarized as follows.

1. Chapter 2 considers novel centralized CRAs in CDMA MPR channels. The system employs a BS that constructs system backlog information from multiple access interference (MAI). Then, the BS broadcasts the retransmission probability for each slot to all mobile terminals within its coverage in order to maximize the expectation of the system throughput conditioned on the number of retransmitting terminals. Under perfect/imperfect power control, the performance of our algorithms and the system stability are evaluated by theoretical analysis and compared to simulations.

2. In contrast with various CRAs [7][8][10][36] proposed in the literature which are based on system backlog estimation, recent standards such as LTE and IEEE 802.16 systems have already adopted window-based backoff algorithms, UB and BEB algorithms with retry limit, respectively as CRA. Compared to the algorithms proposed
Chapter 1. Introduction

in Chapter 2, UB and BEB algorithms do not take the backlog size into account. Thus, Chapter 3 examines how well these algorithms work in CDMA-OFDMA channels and what stability conditions these algorithms have. In addition, the access prioritization schemes are considered to provide different access classes differential performance. To this end, the system allows different access classes to use UB or BEB algorithm with different parameters. In order to optimize UB algorithm, this chapter considers a dynamic window assignment algorithm that is based on Bayesian broadcasting, in which the base station adaptively controls the window size of the UB algorithm for unsaturated traffic condition.

3. In Chapter 3, we ignore the impact of the PR scheme on the system throughput, by assuming that it works as an open loop power control in UMTS and LTE. Since it has been known that higher throughput can be achieved by power capture, Chapter 4 examines whether the PR scheme in LTE affects the system performance in conjunction with UB algorithm. This is done by constructing a metric called throughput gain per average transmission power additionally consumed by the PR scheme.

4. Chapter 5 considers the queueing performance of multichannel S-ALOHA, i.e. CDMA-OFDMA channels, with UB algorithm. This work is an extension of a classical problem of queueing study on S-ALOHA system, which has a long research history. In contrast with previous work that considers only the independent arrival process, this study assumes a Markov modulated Bernoulli process (MMBP) as correlated arrivals over an uplink traffic. Our analysis and simulations show the system performance in relation to the source correlation, number of channels, window size and retry limit. The results from our study allow the parameters of UB algorithm to be properly chosen to meet the access-level quality of service requirements.
5. In SPS of LTE [6], a terminal obtains a traffic channel at the beginning of the ON state of VoIP traffic by either random access or a dedicated control channel. Once a traffic channel is reserved, a terminal can transmit its VoIP packet periodically in the reserved traffic channel without additional control signalling during an entire ON period. Chapter 6 examines the feasibility of S-ALOHA in SPS of LTE as a way of establishing a traffic channel for VoIP. Since large random access delay in establishing a traffic channel may deteriorate the quality of VoIP traffic, we examine the maximum number of acceptable VoIP terminals without exceeding some delay constraint.

1.4 Thesis Organization

As shown in Fig. 1.4, the rest of this thesis is organized as follows. Chapter 2 proposes novel centralized cross-layer CRAs in CDMA MPR channel and examine their performance and stability, under perfect/imperfect radio channel conditions. Chapter 3 examines the performance of the existing decentralized backoff algorithms such as UB and BEB algorithms,
used in LTE and IEEE 802.16 systems, which can be compared to the CRAs proposed in
Chapter 2. Furthermore, an optimal window size of UB algorithm based on backlog esti-
mation is considered. In Chapter 4, we investigate the effectiveness of the PR scheme with
UB algorithm. Chapter 5 considers the queueing performance of multi-buffered terminals
in multichannel S-ALOHA with UB algorithm, which can be found in LTE random access
channel. In particular, a correlated arrival over uplink by MMBP is assumed. Chapter 6
examines the performance of SPS in LTE, i.e., a reservation random access protocol, in
which reservations requested by MPR S-ALOHA. In Chapters 3-6, the system model and
its random access protocol are mostly based on LTE. The random access protocol of LTE
is introduced in detail in Chapters 3 and 6. In addition, throughout Chapters 3-6 we only
consider an ideal wireless channel, i.e., no bit error, while we take into account that power
control errors and MAI are dominant in the wireless channel in Chapter 2. Chapter 7
summarizes the overall results and suggests some directions for future research.
Chapter 2

Cross Layer Contention Resolution Algorithms

There exist extensive information theoretic studies on MPR S-ALOHA systems ([2] and references therein), e.g., in terms of maximum achievable stable throughput and capacity regions provided that the limiting queue distribution exists for each terminal. However, CRAs for MPR channels are less studied. Ghez et al. [36] examined a decentralized stable CRA, in which terminals estimate the current backlog size based on the binary channel feedback, i.e., empty or non-empty slot. However, estimation on new packet arrivals is not incorporated. In [4], Yu and Giannakis proposed a tree algorithm with successive interference cancellation. Although it achieves a 69% throughput efficiency without estimating backlog size, all collided packets need to be stored for successive interference cancellation. Sakakibara et al. [37] examined the bistability of ALOHA systems with various capture and MPR channels, but they did not consider how to control the retransmissions in order to avoid the bistability. Ngo and Krishnamurthy [38] investigated MPR S-ALOHA system in a non-cooperative game-theoretic perspective, in which terminals retransmit only when their channel states are above a certain threshold. In such a game-theoretic formulation, it should be assumed that each player, i.e., terminal has \textit{a priori} knowledge of the number of players, which may not be practical in many wireless networks.

In contrast to [4][36][37][38], we propose centralized cross-layer CRAs for immediate first transmission (IFT) and deferred first transmission (DFT) protocols in an MPR channel.
realized in a CDMA S-ALOHA system. Compared to the previous work reviewed above, a distinctive feature of our algorithms is the use of cross-layer information to estimate the system backlog size. More specifically, the BS measures the power of MAI in the physical layer, and estimates the backlog size and new packet arrivals in the medium access control (MAC) layer based on the measurements. The backlog estimation algorithms for S-ALOHA systems with and without MPR channels ([36] and reference therein) so far are mostly event-based distributed algorithms, in which terminals update backlog size, according to the events, e.g., the transmission success, failure or emptiness at each slot. To this end, terminals accessing an MPR channel should be aware of the channel state, so that they could update the current backlog size according to various events, e.g., the number of packets successfully transmitted with and without other interfering packets. However, our proposed algorithms relieve terminals’ processing burden and, as we shall see, they are not confined to conventional CDMA systems, but would be applicable to systems like [6][30] which use random access codes for terminal identification in random access control channels. Also, in [4][36], all terminals with or without a packet should monitor the feedback channel in order to keep track of the backlog size. In contrast, in the proposed algorithms terminals only need to monitor the feedback channel when they have packets to send. To demonstrate the robustness of the backlog size estimation, we further consider systems with imperfect power control. Based on such estimations, BS determines the retransmission probability to maximize the system throughput for every slot and broadcasts this information to the terminals. The performance and stability of the proposed algorithms are analyzed and compared to simulations.

The organization of this chapter is as follows. Sections 2.1 and 2.2 introduce the system model and proposed algorithms, respectively. System performance and stability are analyzed in Section 2.3. Numerical results are presented in Section 2.4. Concluding
2.1 System Model

For this investigation we consider an MPR channel derived from a cellular CDMA system with a finite number of terminals in each cell. Time is divided into slots of a fixed length that is equal to one packet transmission time, and terminals transmit to the BS using S-ALOHA access over the uplink while the BS broadcasts to all terminals over the downlink. We assume that terminals have distinct PN signatures known by the BS [39] and that the BS receiver uses a bank of correlators for multiuser detection. Further, we assume that uplink and downlink channels have approximately the same path loss because, e.g., time division duplexing (TDD) is employed, or the shadowing of both links is strongly correlated in frequency division duplexing (FDD).

At the end of each slot in which some terminals have sent packets, the BS determines which PN signatures (terminals) are successfully received, and broadcasts the following information over a downlink signalling channel to all terminals:

1. The IDs of terminals that successfully transmitted in the previous slot. The IDs of PN signatures could be also transmitted, instead of terminals' IDs.
2. The retransmission probability, $r_t$, for the next slot $t$, as determined by the algorithms proposed in the next section.
3. The signal power at the BS receiver required from a terminal for successful packet reception, and the transmission power of the current broadcast message.

For the sake of simplicity, we assume that this feedback is immediate after an uplink slot ends and before the next slot begins. In practice there is a non-negligible feedback delay,
which can be accommodated by considering the system as a time-multiplexed parallel slotted system [30][40]. We further assume that broadcasts from BS are always received by the terminals without errors. An active terminal that does not receive its ID in the BS broadcast at the end of its transmission slot regards itself as backlogged, and it will then retransmit in the next slot $t$ with probability $r_t$. Furthermore, each terminal in the cell measures the received power of the broadcast message and compares it with the transmission power specified in the message to estimate the path loss between the terminal and the BS, and uses this estimate for open loop uplink access power control.

We denote the received signal from $k$ terminals at the BS at time $t$ by $\psi(t)$. Further, let $S$, $a_j(t)$ and $b_j(t)$, respectively, denote the signal power required from each terminal at the BS receiver, the PN-signature signal of a data bit from the $j$-th terminal and the data bit limited to $[0, T_b]$. We assume that $a_j(t)$ has unit energy and is generated at the rate of $N$ chips (spreading gain) for each data bit. The received signal corresponding to data bits from $k$ terminals is then expressed as

$$\psi(t) = \sum_{j=1}^{k} \sqrt{2\gamma_j} S a_j(t - \tau_j) b_j(t - \tau_j) \cos(2\pi f_c(t - \tau_j) - \varphi_j) + \mathcal{N}(t), \quad (2.1)$$

where $f_c$, $\tau_j$ and $\varphi_j$ denote the carrier frequency, time delay of the $j$-th terminal relative to the beginning slot boundary, and phase of the $j$-th terminal, respectively. We assume that the power control error of the $j$-th terminal is given by a lognormal random variable, $\gamma_j$, whose corresponding normal random variable has zero mean and standard deviation $\sigma_{X_j} (= \sigma_X)$ [41]. We also assume i.i.d. power control errors over all the terminals. Finally, $\mathcal{N}(t)$ denotes additive white Gaussian noise with zero mean and standard deviation $\sigma_g$.

To obtain the data bit error probability in (2.1), we consider the output of a correlation
receiver matched to the $i$-th terminal, which can be expressed as

$$Z_i = \int_0^{T_b} \psi(t) a_i(t) \cos \omega_c t \, dt$$

$$= \sqrt{\frac{S}{2}} T_b \left( \sqrt{\gamma_i} b_i + \sum_{j=1, j \neq i}^{k} \sqrt{\gamma_j} I_{j,i} \right) + \int_0^{T_b} \mathcal{N}(t) a_i(t) \cos \omega_c t \, dt \tag{2.2}$$

where $b_i \in [-1, +1]$ and $I_{j,i}$ is assumed to be a Gaussian random variable with zero mean and variance of $1/3N$ [42]. It should be noted that each individual interference component may not be Gaussian distributed, but given a sufficiently large number of interference components the sum of these components can be assumed to be Gaussian distributed. Based on (2.2), we can express the conditional bit error probability given $\hat{\gamma}_{k-1}$, denoted by $P_{b|\hat{\gamma}_{k-1}}$, as

$$P_{b|\hat{\gamma}_{k-1}} = Q\left( \sqrt{\frac{3N}{\hat{\gamma}_{k-1}}} \right), \tag{2.3}$$

where $Q(y) = (1/\sqrt{2\pi}) \int_y^\infty \exp^{-x^2/2} \, dx$. Additionally, $\hat{\gamma}_{k-1}$ is the sum of $k-1$ i.i.d. lognormal random variables, i.e., $\hat{\gamma}_{k-1} = \sum_{j=1, j \neq i}^{k} \hat{\gamma}_j$, in which $\hat{\gamma}_j = \gamma_j / \gamma_i$ is a lognormal random variable whose corresponding normal random variable has zero mean and variance of $\sigma^2 = 2\hat{\sigma}_X^2$. Note that with perfect power control, $\hat{\gamma}_j$ is equal to one, in which case the bit error probability is given by $Q(\sqrt{3N/(k-1)})$ [42]. In (2.3), we assume that the white Gaussian noise component is negligible. The bit error probability given $k - 1$ terminals’ transmissions, $P_{b|k-1}$, can be obtained by

$$P_{b|k-1} = \int_0^{\infty} P_{b|\hat{\gamma}_{k-1}} f_{k-1}(y) \, dy \tag{2.4}$$

where $f_{k-1}(y)$ is the probability distribution of the sum of $k - 1$ i.i.d. lognormal distributions with zero mean and variance $\hat{\sigma}_{k-1}^2$. By Wilkinson’s method [45], we write $f_{k-1}(y)$
approximately as

\[
f_{k-1}(y) = \frac{1}{\sqrt{2\pi y \hat{\sigma}_{k-1}^2}} e^{-\frac{(\ln y - \hat{\mu}_{k-1})^2}{2\hat{\sigma}_{k-1}^2}}
\]

in which \(\hat{\mu}_{k-1}\) and \(\hat{\sigma}_{k-1}^2\) are respectively expressed as

\[
\hat{\mu}_{k-1} = \ln(k-1) + \frac{1}{2} \sigma^2 - \frac{1}{2} \hat{\sigma}_{k-1}^2 \quad \text{and} \quad \hat{\sigma}_{k-1}^2 = \ln \left( \frac{1}{k-1} e^{\sigma^2} + \frac{k-2}{k-1} \right).
\]

Assume that a packet consists of \(L_b\) bits. Then, the packet success probability for an uncoded system with \(k\) terminals transmitting is given by \(P_s(k) = (1 - P_{b|k-1})^{L_b}\). Throughout this chapter, we use the notation of a binomial distribution as \(B(i, n, p) = \binom{n}{i} p^i (1-p)^{n-i}\).

Then, we obtain the packet success probability for a coded system with a \(v\)-error-correcting code as \(P_s(k) = \sum_{i=0}^{v} B(i, L_b, P_{b|k-1})\).

### 2.2 Algorithm Design

We design CRAs based on centralized estimation of the system backlog and selection of the retransmission probability, \(r_t\), at the BS for the MPR channel modelled above. We consider two possible variations of the S-ALOHA protocol. In the IFT protocol, newly generated packets are transmitted with probability one, whereas in the DFT protocol they join the backlog without immediate initial transmissions.

We present two CRAs, **Algorithm 2.1** and **Algorithm 2.2** for the IFT and DFT protocols, respectively. In the algorithms, \(r_t^{(I)}\) or \(r_t^{(D)}\) denote the retransmission probabilities broadcasted at time \(t\) for IFT or DFT protocol, respectively. The number of packets successfully transmitted at time \(t\) is denoted by \(\xi_t\). The estimated backlog size, estimated and predicted number of newly transmitted packets at time \(t\) are respectively denoted by \(\beta_t\).
Chapter 2. Cross Layer Contention Resolution Algorithms

\( \lambda_t \) and \( \hat{\lambda}_t \). Finally, \( u_t \) denotes the estimated number of packets simultaneously transmitted at time \( t \).

For both Algorithm 2.1 and 2.2, the retransmission probability \( r_t \) is determined as follows: We denote \( C_{j,k} \) as the probability that \( j \) packets are successfully received among a total of \( k \) packets transmitted. Assuming that successful packet reception with probability \( P_s(k) \) is independent for each terminal, we can express \( C_{j,k} \) for an uncoded system as

\[
C_{j,k} = B(j, k, P_s(k)).
\]

(2.7)

The conditional system throughput given \( k \) packets for an uncoded system is given by

\[
\overline{C}_k = \sum_{j=0}^{k} jC_{j,k}.
\]

(2.8)

Let \( c^* \) denote the number of terminals that should be transmitting simultaneously to maximize \( \overline{C}_k \), which is obtained by

\[
c^* = \text{arg}_k \max \overline{C}_k.
\]

(2.9)

Now, suppose that \( \beta_t \) backlogged terminals transmit their packets based on a Bernoulli process with probability \( r_t \). The mean number of transmissions from the \( \beta_t \) backlogged terminals is \( \beta_t r_t = \sum_{i=0}^{\beta_t} iB(i, \beta_t, r_t) \), which is expected to be equal to \( c^* \). Accordingly, \( r_t \) can be obtained as

\[
r_t = \min\left(\frac{c^*}{\beta_t}, 1\right),
\]

(2.10)

which sets \( r_t \) to one when \( \beta_t \) is less then \( c^* \). For an S-ALOHA system without MPR channel, i.e., \( c^* = 1 \), one can readily find \( r_t = 1/\beta_t \). Additionally, \( \overline{C}_{c^*} \) represents the
Chapter 2. Cross Layer Contention Resolution Algorithms

Algorithm 2.1 - CRA for IFT protocol

1: Initialization: $\beta_0 = \hat{\lambda}_1 = 0$ and $r_1^{(I)} = 0.5$.
   Do the following at $t = 1, 2, 3, \ldots$:
2: Compute $u_t$ based on (2.13).
3: Estimate $\lambda_t = \max \left( u_t - r_t^{(I)} \cdot \beta_{t-1}, 0 \right)$.
4: Update $\beta_t = \max (\beta_{t-1} + \lambda_t - \xi_t, 0)$.
5: Predict new arrival by a first-order AR model:
   $\hat{\lambda}_{t+1} = \theta \hat{\lambda}_t + (1 - \theta) \lambda_t$.
6: Broadcast $r_{t+1}^{(I)}$:
   $r_{t+1}^{(I)} = \min \left( \max \left( (c^* - \hat{\lambda}_{t+1})/\beta_t, 0 \right), 1 \right)$.

maximum throughput or capacity of the MPR channel. For a coded system, $c^*$ is expressed as

$$c^* = \arg_k \max \max \left( L(v) C_k \right) \quad (2.11)$$

where $L(v)$ denotes the code rate of a $v$-error-correcting code. Note that both $c^*$ and $C_{c^*}$ are functions of power control error $\sigma_X^2$, which requires the BS to know $c^*$ under power control errors. Additionally, $c^*$ also depends on the MPR channel. For practical implementation, we assume that the BS can estimate power control errors from the random access channel by comparing the measured power and the required power broadcasted. Then, BS uses one of the predetermined values of $c^*$'s (a table) according to the estimated power control errors.

In order to estimate $\beta_t$ in (2.10), we first estimate the number of packets simultaneously transmitted at time $t$ (denoted by $u_t$) and determine $\beta_t$ based on the IFT and DFT protocols. To this end, the BS measures the total received energy (before decorrelation)
per bit. Given \( k \) packets transmitted, we can write this as

\[
\int_0^{T_b} \psi(t)^2 dt = T_b \left( S \sum_{j=1}^{k} \gamma_j + 2S \sum_{j=1}^{k-1} \sqrt{\gamma_j} \sum_{\ell=j+1}^{k} \sqrt{\gamma_\ell I_{j,\ell}} \right) + \mathcal{N}_e
\]  

in which all the terms related to \( \mathcal{N}(t) \) are collectively denoted by \( \mathcal{N}_e \). Taking expectation on (2.12), we have

\[
E \left[ \frac{1}{T_b} \int_0^{T_b} \psi(t)^2 dt \right] = S \cdot E \left[ \sum_{j=1}^{k} \gamma_j \right],
\]  

(2.13)

The second term in (2.12) vanishes in (2.13) due to the Gaussian assumption made in (2.2). Dividing (2.13) by \( S \) and rounding it, we can have an estimator \( u_t \) for \( k \) and apply it in Step 2 of each algorithm. As in (2.5) and (2.6), the probability distribution of \( \sum_{j=1}^{k} \gamma_j \) can also be approximated as a lognormal distribution by Wilkinson’s method.

Based on (2.9) and (2.13), each step of the algorithms can be explained as follows:

In order to distinguish new and retransmission packets in Step 3, the number of packet retransmissions at time \( t \) is estimated as the product of \( r_t \) and the previously estimated backlog \( \beta_{t-1} \), i.e., the expected number of packet retransmissions from the previous backlog. By subtracting this from \( u_t \), we can estimate \( \lambda_t \), the number of new packets at time \( t \). Note that Step 3 is identical for both the IFT and DFT protocols\(^2\). In Steps 4-5, the BS updates the current backlog by subtracting successfully transmitted packets and adding new arrivals, and then predicts new packet arrivals. By reserving some capacity for new packets in the IFT protocol, the BS determines \( r_t \) in Step 6. However, in the DFT protocol, the backlog estimation is once more smoothed out in Step 4, and we add the predicted new arrivals to this backlog in Step 6. Capacity reservation for new arrivals is not needed.

Regarding the algorithms, we have the following remarks:

\(^2\)In Algorithm 2.2, this can be seen when both sides are multiplied by \( r_t^{(D)} \) in Step 3.
Chapter 2. Cross Layer Contention Resolution Algorithms

Algorithm 2.2 - CRA for DFT protocol

1: Initialization: $\beta_0 = \hat{\lambda}_0 = 0$ and $r_1^{(D)} = 0.5$.
   Do the following at $t = 1, 2, 3, \ldots$:
2: Compute $u_t$ based on (2.13).
3: Estimate $\lambda_{t-1} = \max \left( \frac{u_t}{r_t^{(D)}} - \beta_{t-1}, 0 \right)$.
4: Update $\beta_{t-1} = \rho \beta_{t-1} + (1 - \rho) \left( \frac{u_t}{r_t^{(D)}} \right)$ and
   $\beta_t = \max (\beta_{t-1} - \xi_t, 0)$.
5: Predict new arrival by $\hat{\lambda}_t = \theta \hat{\lambda}_{t-1} + (1 - \theta) \lambda_{t-1}$.
6: Broadcast $r_{t+1}^{(D)}$ as $r_{t+1}^{(D)} = \min \left( c^*/(\beta_t + \hat{\lambda}_t), 1 \right)$.

1. For simplicity, we ignore mobility of terminals or assume that they are either static or
   show very slow mobility compared to the processes for packet arrivals and successful
   transmissions. Then, the algorithms do not depend on time, but on the numbers of
   previous backlog and new packets, and we can assume that $\beta_t$ is a stationary process.

2. For new packets, $\lambda_t$ can in practice depend on the total number of terminals residing
   in a cell, their mobility, population size and traffic characteristics and the current
   backlog. In Step 5 of each algorithm, the parameter $\theta \in (0, 1)$ is a smoothing factor
   of a first-order AR model. In Section 2.4, we show that the desired performance can
   be achieved by choosing $\theta > 0.95$.

3. By using the equations of Steps 3-4 in the IFT protocol, we can express the evolution
   of the backlog over time as

   $$\beta_{t+k} = \beta_{t-1} \prod_{j=0}^{k} \left( 1 - r_{t+j}^{(I)} \right) + \sum_{i=0}^{k} \vartheta_{t+i} \prod_{j=1}^{k-i} \left( 1 - r_{t+j}^{(I)} \right)$$

   with $\vartheta_{t+i} = \max \left( r_{t+i}^{(I)} \beta_{t-1+i} + \lambda_{t+i} - \xi_{t+i}, 0 \right)$. The term $r_{t+i}^{(I)} \beta_{t-1+i} + \lambda_{t+i}$ indicates
   the sum of the expected number of retransmissions and new packet arrivals. Therefore,
   $\vartheta_{t+i}$ denotes the remaining packets due to transmission failure. The first term on
the right-hand side indicates the remaining backlogged terminals which would not retransmit at each time. The second term denotes the terminals which have failed to transmit and stayed in the backlog without further retransmissions. Note that \( \xi_{t+i} \) is a function of \( r_t^{(I)} \beta_{t-1+i} + \lambda_{t+i} \). The algorithm chooses \( r_t^{(I)} \) in order to minimize \( \vartheta_{t+i} \). As \( k \to \infty \), the evolution of the backlog over time \( \beta_{t+k} \) becomes smaller when \( r_t^{(I)} \leq r_{t+j+1}^{(I)} \) and \( \vartheta_{t+i} < r_{t+i}^{(I)} \beta_{t-1+i} \). By using Step 6, we can rewrite \( r_t^{(I)} \leq r_{t+j+1}^{(I)} \) as \( \lambda_{t+1} - \lambda_t - (\beta_{t-1} - \beta_t) \leq 0 \). This means that the backlog’s decreasing rate should be greater than or equal to the new packet arrival rate. This corresponds to the negative drift condition in [46], which we shall see in Section 2.3.2. We also rewrite \( \vartheta_{t+i} < r_{t+i}^{(I)} \beta_{t-1+i} \) as \( \lambda_{t+i} < \xi_{t+i} \). For \( \lambda_{t+i} = \xi_{t+i} \) the backlog does not change at all.

4. Since new packet arrivals become backlogged immediately in the DFT protocol, Step 3 is based on \( u_t = r_t^{(D)} (\beta_t^{*} + \lambda_t^{*}) \) in Algorithm 2.2 where \( \beta_t^{*} \) and \( \lambda_t^{*} \) are true values of backlog and new packet arrivals at time \( t-1 \). In Step 4, based on a first-order AR model we simply smooth out \( \beta_{t-1} \) by \( \rho \) and \( u_t / r_t^{(D)} \) which contains noisy versions of \( \beta_t^{*} \) and \( \lambda_t^{*} \). In other words, the previous estimation \( \beta_t^{*} \) is adjusted by noisy observation \( u_t / r_t^{(D)} \).

5. Since terminals have spatially orthogonal signature vectors in SDMA [3], (2.12) can be still valid for backlog estimation in SDMA.
2.3 Algorithm Evaluation

2.3.1 Performance Measures

The retransmission probability is said to be stationary if \( r \) depends only on the state of the backlog process. For the IFT protocol, we can write

\[
\lim_{t \to \infty} r_t^{(I)} = \lim_{t \to \infty} \min \left( \max \left( \frac{c^* - \hat{\lambda}_{t+1}(\epsilon_a)}{\beta_t(\epsilon_b)}, 0 \right), 1 \right),
\]

in which as \( t \to \infty \), we have \( \hat{\lambda}_{t+1}(\epsilon_a) \approx (M - m)p + \epsilon_a \) and \( \beta_t(\epsilon_b) \approx m + \epsilon_b \) when there are in total \( M \) terminals and \( m \) backlogged terminals. The variables \( \epsilon_a \) and \( \epsilon_b \) indicate the estimation errors for new arrivals and backlog size, respectively. When no estimation errors are assumed, we can write \( r_m^{(I)} \) and \( r_m^{(D)} \) for \( m \) backlogged terminals as

\[
r_m^{(I)} = \min \left( \frac{c^* - (M - m)p}{m}, 1 \right) \quad \text{and} \quad r_m^{(D)} = \min \left( \frac{c^*}{m}, 1 \right). \tag{2.14}
\]

Let \( q_{m,n} \) denote the one-step state transition probability of the backlogged terminals, i.e., \( q_{m,n} = \Pr [\beta_{t+1} = n | \beta_t = m] \). Suppose that \( M \) terminals exist in a cell and that \( M - m \) terminals (except \( m \) backlogged terminals) generate new packets every slot each with probability \( p \). For the IFT protocol, \( q_{m,n} \) for \( 0 \leq m, n \leq M \) can be expressed as

\[
q_{m,n} = \sum_{v=0}^{M} \sum_{k=0}^{v} C_{k,v} B(l, M - m, p) B(v - l, m, r_m^{(I)}).
\tag{2.15}
\]

For the DFT protocol, we can obtain \( q_{m,n} \) by

\[
q_{m,n} = \sum_{v=0}^{m} \sum_{k=0}^{v} C_{k,v} B(l, M - m, p) B(v, m, r_m^{(D)}). \tag{2.16}
\]
where \( l = n - m + k \). We then write the state transition matrix for the system backlog as 
\[
Q = [q_{m,n}],
\]
where \( q_{m,n} \) is the element at the \( m \)-th row and \( n \)-th column. Further, we denote the stationary probability row vector by \( \vec{\pi} = [\pi_m] \) for \( 0 \leq m \leq M \), where \( \pi_m \) denotes the probability that \( m \) backlogged terminals are in the system. We can obtain \( \vec{\pi} \) by 
\[
\vec{\pi} = \vec{\pi} \cdot Q
\]
and \( \vec{\pi} \cdot \vec{e} = 1 \), where \( \vec{e} \) denotes a unit column vector whose length corresponds to \( \vec{\pi} \). We denote the throughput given \( m \) backlogs in the system by \( S_m \). We can obtain \( S_m \) for the IFT protocol by 
\[
S_m = \sum_{n=0}^{M} \sum_{v=1}^{M} \sum_{k=1}^{v} kC_{k,v} \mathcal{B}(l, M - m, p) \mathcal{B}(v - l, m, r_m^{(I)}).
\]
For the DFT protocol, one can get 
\[
S_m = \sum_{n=0}^{M} \sum_{v=1}^{m} \sum_{k=1}^{v} kC_{k,v} \mathcal{B}(l, M - m, p) \mathcal{B}(v, m, r_m^{(D)}).
\]
For both the IFT and DFT protocols, the overall system throughput and the backlog delay are respectively obtained by 
\[
\bar{S} = \sum_{m=0}^{M} \pi_m S_m \quad \text{and} \quad \bar{D} = \frac{\bar{N}}{\bar{S}},
\]
where \( \bar{N} = \sum_{m=0}^{M} m \pi_m \) is the average number of backlogged terminals in the system.

### 2.3.2 System Stability

To evaluate the stability of the algorithm, we examine the negative drift condition \( d(m) \) [46] (introduced in Section 1.2.2), which is obtained by 
\[
d(m) = E[\beta_{t+1} - \beta_t | \beta_t = m] = \sum_{n=0}^{M} (n - m) q_{m,n}.
\] (2.17)

Clearly, we have \( d(0) > 0 \) and \( d(M) < 0 \). The system is said to be mono-stable if there exists a single state \( m \) which makes \( d(m) \) transit from positive to negative, and from such
Chapter 2. Cross Layer Contention Resolution Algorithms

2.4 Numerical Results and Discussions

We use the spreading gain $N = 16$ and the packet length $L_b = 255$ in the following examples. Unless otherwise stated, population size $M = 50$, $\rho = 0.95$ in the DFT protocol, and the smoothing factor of estimating new packet arrivals is $\theta = 0.95$ for both the IFT and DFT protocols. In all the figures, the curves with circles and triangles indicates IFT and DFT protocols, respectively. For verification, we build a simulation program with MATLAB. With perfect power control, we assume that each packet has unit power, while the power of each packet under imperfect power control follows lognormal random variables. At initialization, we set randomly some of $M$ terminals as backlogged. Here, we do not

Figure 2.1: System performance under perfect power control

(a) Throughput vs. $p$. (b) Backlog delay vs. $p$ under perfect power control.

a state $m$, $d(m)$ would not bounce back to positive again. If there are two states to make $d(m)$ transit from positive to negative, the system is said to be bistable, and the backlog size oscillates between these two states.
include mobility of terminals and the radio link simulation for bit error probability. Therefore, packet success is determined by (2.4) in our simulation program. In particular, we averaged simulations of 30000 slots over 5 runs in Figs. 2.3(a) and 2.3(b). In Figs. 2.1(a)-2.3(b), we characterize the performance of the proposed algorithms with perfect power control by varying $p$, $\theta$ and $\rho$, while in Figs. 2.4(a)-2.5(b) we examine the performance with imperfect power control.

In Figs. 2.1(a) and 2.1(b), system throughput $\overline{S}$ and backlog delay $\overline{D}$ of the uncoded system with perfect power control are respectively depicted. The two figures show good agreement between analysis and simulations for $p < 0.055$ and $p > 0.1$, which means that performance of the algorithms with perfect power control is close to the upper bounds of the analytical models for these cases. In particular, Fig. 2.1(b) shows that the delay of DFT is slightly larger than that of IFT especially for $p \leq 0.07$. It results from the fact that in the DFT protocol, new packets should join the backlog and wait, whereas the IFT protocol allows new packets to be transmitted immediately. For $p > 0.07$, the delays of
Chapter 2. Cross Layer Contention Resolution Algorithms

IFT and DFT protocols converge, while $\overline{S}$ is saturated.

In Figs. 2.2(a) and 2.2(b), we examine the effect of $\theta$ in IFT/DFT protocols and of $\rho$ in DFT on $\overline{S}$. Fig. 2.2(a) shows that $\overline{S}$ decreases for high $p$, when small $\theta$ is applied. This implies that the backlog size varies slowly. Particularly in Fig. 2.2(b) we examine $\overline{S}$ by varying $M$ at $p = 0.075$. Note that the mean of traffic intensity in the system can be expressed by $Mp$. Accordingly, we can see almost identical $\overline{S}$ at $p = 0.09$ in Fig. 2.1(a) and at $M = 60$ in Fig. 2.2(b).

For the effect of $\rho$ in the DFT protocol, we can improve $\overline{S}$ at $M = 50$ by using $\rho = 0.75$. With smaller $\rho$, the enhancement of $\overline{S}$ is not significantly found. Moreover, $\overline{S}$ is slightly degraded with small $\rho$ for large $M$. Note that in the analytical model, the evolution of the system backlog over time can be governed by the one step transition probability matrix $Q^n$ for $n = 1, 2, \ldots$ where $n$ is the discrete time index. By spectral decomposition of $Q$, we can conjecture that the second largest eigenvalue$^3$ of $Q$ plays a critical role in explaining

\footnotesize{$^3$The first largest eigenvalue of $Q$ is always one due to stationarity.}
the transient behaviour and in representing the correlation of a Markov process [47], which models the backlog size here. Since \( \rho \) represents how much \( \beta_t \) correlates to \( \beta_{t-1} \) in the first order AR model, we would choose \( \rho \) in proportion to the second largest eigenvalue of \( Q \).

In Figs. 2.3(a) and 2.3(b), we present the negative drift, \( d(m) \), of the analytical models with perfect knowledge on backlog size and simulations with perfect power control of IFT/DFT protocols, respectively. We choose \( p = 0.07 \) and \( p = 0.1 \), since the system is with the onset of saturation and in saturation at these values in Figs. 2.1(a) and 2.2(a). In Fig. 2.3(a), \( d(m) \) of analytical model in IFT protocol agrees quite well with that of simulation, and a single stable operating point is respectively found for \( p = 0.07 \) and 0.1. Note that negative drift for \( 0.07 < p < 0.1 \) lies between the lines depicted in Fig. 2.3(a). Fig. 2.3(b) shows \( d(m) \) in DFT protocol. Some differences between analytical and simulation results are found. In particular, \( d(m) \) goes to slightly positive at around \( m = 10 \) and becomes negative a few states away. Accordingly, two operating points are found, i.e., bistability. Even though such a bistability is found in DFT protocol, the system performance are not severely degraded as shown in Figs. 2.1(a)-2.2(b). This results from the fact that such two operating points are closely located and that the small positive increase of \( d(m) \) might be negligible.

In Figs. 2.4(a)-2.4(d), solid, dashed and dotted lines denote \( \sigma_X^2 = 0.5 \text{ dB}, 1 \text{ dB} \) and \( 1.5 \text{ dB} \), respectively. In Figs. 2.4(a) and 2.4(b), imperfect power control is considered for an uncoded system. It is noticeable that \( c^* \) and \( \overline{C_c^*} \) become smaller due to high bit error probability, compared to the case with perfect power control. Accordingly, the larger the power control error, the lower \( \overline{S} \) is observed. Additionally, the system becomes saturated when \( p > 0.05 \), because \( \overline{C_c^*} \) is decreased. To measure throughput efficiency for the system with imperfect power control, we obtain the throughput efficiency by dividing the maximum of \( \overline{S} \) in Fig. 2.4(a) by \( \overline{C_c^*} \). Then, the corresponding throughput efficiencies of an uncoded
Chapter 2. Cross Layer Contention Resolution Algorithms

(a) Throughput of uncoded system.

(b) Backlog delay of uncoded system.

(c) Throughput of BCH-coded system.

(d) Backlog delay of BCH-coded system.

Figure 2.4: Comparison of uncoded and BCH-coded system under imperfect power control
system with \( \sigma_X^2 = 0.25, 0.5 \) and 1 (in dB) are 79\%, 69\% and 54\%, respectively. Note that power control errors affect packet error probabilities as well as backlog estimation errors.

In Figs. 2.4(c) and 2.4(d), \( \overline{S} \) and \( \overline{D} \) of Bose and Ray-Chaudhuri (BCH)-coded system \((v = 8)\) are depicted. We obtain the throughput of a coded system by multiplying \( \overline{S} \) by the coding rate \( L(v) \) in (2.11). Compared to \( \overline{S} \) and \( \overline{D} \) of an uncoded system, a coded system shows higher \( \overline{S} \) and lower \( \overline{D} \) for the same power control errors, e.g., 47\% throughput enhancement for \( \sigma_X^2 = 1 \) dB. We obtain the throughput efficiency for \( \sigma_X^2 = 0.5, 1 \) and 1.5 (dB) as 50.62\%, 50.09\% and 52.07\%, respectively. For \( \sigma_X^2 = 2 \) dB (not depicted), the algorithms show a throughput efficiency of 54\%. Note that the throughput in Fig. 2.4(c) and Fig. 2.5(a) is normalized by multiplying coding rate, \( L(v) \). Additionally, we observe the role of \( \rho \) in the DFT protocol for a coded system. As previously mentioned, the performance of the algorithm for the DFT protocol becomes close to that for IFT protocol, when \( \rho \) is chosen as a value close to the second largest eigenvalue of \( Q \) in (2.16), particularly for small \( p \) as in Fig. 2.2(b).

In Figs. 2.5(a)-2.5(b), the effect of error correcting capability, \( v \), is examined for coding rate \( L(3) = 0.9, L(8) = 0.75 \) and \( L(14) = 0.57 \). We can obtain \( v \) by finding the argument \( v \) to maximize (2.11). In Fig. 2.5(b), the backlog delay, which means the time to be taken for sending 255 information bits in a coded system, is normalized by dividing \( \overline{D} \) by \( L(v) \). While the normalized backlog delays for \( v = 8 \) and 14 are not distinguishable in Fig. 2.5(b), Fig. 2.5(a) shows that the system with \( v = 8 \) shows better throughput performance than others.

### 2.5 Summary

In this chapter we have proposed throughput efficient cross-layer CRAs for IFT and DFT protocols in CDMA-based MPR S-ALOHA systems, in which MAI is exploited to estimate
the system backlog in the presence of power control errors. In the proposed algorithms, retransmission probability for backlogged terminals is chosen slot-by-slot to maximize the conditional expected system throughput. For the IFT and DFT protocols, the performance of the algorithms under perfect power control is close to that predicted by the analytical performance model without estimation errors. By examining negative drift, the algorithms are shown to be stable. For uncoded/coded systems under imperfect power control, system throughput decreases due to reduction of channel capacity. Especially, the throughput efficiency in coded systems is maintained over 50% up to power control error of 2 dB.
Chapter 3

Existing Backoff Algorithms for
Single-Buffered Terminals

The CRAs standardized in LTE and IEEE 802.16 systems employ a backoff algorithm with retry limit: UB in the case of LTE [6] and BEB in the case of IEEE 802.16 system [30]. Besides retry limits, each standard considers some access level priority in order to provide quality of service (QoS) at the MAC level among the services such as security services, public utilities, emergency service, e.g., rescue in a congested area [31]. In addition to prioritization in the access control, access priority can be further enhanced by differentiating some parameters of those backoff algorithms. In this chapter, we provide a comparative study on these backoff algorithms with retry limit and access priority differentiation.

Previous work on backoff algorithms with retry limit can be categorized based on the design of the backoff algorithm, i.e., window-based or probability distribution-based, and the modeling assumptions, i.e., finite or infinite population, and saturated or unsaturated traffic condition. Kim [48] considered a frequency-hopped CDMA system under an infinite population model, in which uniform retransmission intervals are assumed. It has been shown that system stability can be achieved by controlling retry limit and the channel coding rate that affects the transmission success rate of a packet. Sakakibara et al. examined EB algorithm with retry limit in S-ALOHA systems [37][49]. Additionally, Liu [50] examined an EB algorithm in frequency-hopped S-ALOHA systems under a finite population with unsaturated traffic (FP/UST) model. The EB algorithm in [37][49][50],
which retransmits a packet with probability \( r_i (\leq r_{i-1}) \) after it has been unsuccessfully transmitted \( i - 1 \) times, is an approximate model for the widely-used window-based BEB backoff algorithm. While the system bistability has been extensively examined according to the retry limit in [37][49][50], packet delay and packet dropping probability have not been evaluated. In [19], Kwak et al. considered retry limit of window-based BEB in S-ALOHA under a finite population with saturated traffic (FP/ST) model. Similarly, Falla et al. [51], Vu et al. [52] and Chuck et al. [53] examined window-based BEB with retry limit for an IEEE 802.16 system under FP/ST. It should be noted that S-ALOHA system under FP/ST model always operates in a stable but inefficient region [37][49]. In contrast to [19][51][52][53] with FP/ST model, in this chapter we examine the performance of UB and BEB algorithms with retry limit under FP/UST model, which is closer to practical situations, and discuss their stability.

For some access prioritization scheme in random access, Xiao [32] examined not only the effect of the retry limit of BEB algorithm in IEEE 802.11 system\(^4\), but also considered prioritized access schemes by differentiating contention window size of BEB algorithm under FP/ST. Additionally, Angeles et al. [54] considered a prioritization scheme in S-ALOHA system by differentiating retransmission probability of high- and low-priority terminals. In comparison with [32][54], we consider the performance of UB and BEB algorithms with access priority and their stability under FP/UST model. Our comparative study of UB and BEB algorithms could provide a comprehensive understanding on the random access in LTE and IEEE 802.16 systems.

The contributions of this chapter are three-fold. First, we examine UB and BEB algorithms with retry limit used in the RACHs of LTE and IEEE 802.16 system under FP/UST model for non-prioritized access. Secondly, we consider some prioritized access

\(^4\)The BEB algorithm in IEEE 802.11 systems decrements its backoff counter during idle channel, while the UB and BEB algorithms in wireless cellular networks [6][31] decrement its backoff counter regardless of channel status.
Chapter 3. Existing Backoff Algorithms for Single-Buffered Terminals

schemes by differentiating the persistence value [31] and other parameters of the UB and BEB algorithms. Thirdly, we extend an existing Bayesian broadcasting algorithm [55] into a dynamic window assignment (DWA) scheme in the UB algorithm, which is suitable for application in LTE. In this chapter, we ignore the physical characteristics of the OFDM-CDMA channel in order to focus on the backoff algorithms in the MAC layer. The main results can be summarized as follows:

1. Under an FP/UST model, we obtain the stability conditions of the system with UB and BEB backoff algorithms with/without retry limit. In contrast to the previous work [37][49][50] showing that some approximate backoff algorithms exhibit bistability, our analysis shows that UB and BEB algorithms without retry limit do not have such a bistability characteristic. Instead, these backoff algorithms operate in either a stable or unstable manner. Thus, instead of controlling the retry limit to maintain system stability as in [37][49][50], we shall see that the retry limit provides a tradeoff between throughput and delay performance at the expense of the packet dropping probability.

2. In contrast to the extensive studies on BEB algorithms in the previous work [19][32][51][52][53], we compare the performance of the UB algorithm standardized in LTE [6] against that of the BEB algorithm standardized in WiMAX [30], in terms of throughput, mean and variance of delay, and packet dropping probability given the retry limit. We show that UB can be designed to have a similar performance as BEB. It is also shown that UB achieves much smaller variances of total access delay than BEB. Note that the variance of the access delay is an important indicator of fair use of the channel by all terminals over time; i.e., a backoff algorithm showing a small variance in retransmission delay will support more even channel access by the terminals.

3. We examine some access prioritization schemes for the UB and BEB algorithms by
differentiating the persistence probability as specified in [6] and window size. It is shown that the persistence value effectively provides access prioritization, especially when the system is overloaded.

4. In [6], LTE allows the window size of UB algorithm to be dynamically changed by the eNodeB (a base station of LTE) according to network congestion. Our DWA scheme for the UB algorithm outperforms the existing mechanism in the RACH of LTE for static and time-varying traffic conditions, provided that orthogonality of RAPs are preserved.

### 3.1 Backoff Algorithms

#### 3.1.1 System Model

The RACH of LTE and IEEE 802.16 systems have been introduced in Section 1.1. In both systems, if the number of retransmissions exceeds the maximum retry limit, the RAP is dropped in the MAC layer. This is reported to a higher layer. Once the RAP is successfully transmitted, the terminal can transmit a bandwidth request message for bandwidth reservation on the uplink channel that the BS has granted to it via the random access response (RAR) message in LTE or the uplink-map in IEEE 802.16 system. Upon each collision, terminals in each system conduct the following backoff algorithm.

#### 3.1.2 Uniform Backoff Algorithm in LTE

Each backlogged terminal draws randomly a backoff counter between 0 and $U - 1$. The backoff counter is then decremented by one at every slot, until it reaches zero. When the window size of UB algorithm to be dynamically changed by the eNodeB (a base station of LTE) according to network congestion. Our DWA scheme for the UB algorithm outperforms the existing mechanism in the RACH of LTE for static and time-varying traffic conditions, provided that orthogonality of RAPs are preserved.

---

5In the UB algorithm of LTE [6], thirteen different window sizes from 0 and 960 msec are specified, which can be dynamically selected.
backoff counter is zero, the terminal selects a random real number between 0 and 1. As in [31], if this number is less than the persistence probability (or persistence value) $p$, the terminal transmits its packet; otherwise it postpones the packet transmission by repeating the above procedure in the next slot, until the packet is transmitted. When the packet is not successfully transmitted, i.e., the transmission encounters a collision, the terminal delays its retransmission again by a random time newly chosen between 0 and $U - 1$. This trial is repeated up to $L$ times more, until the retransmission succeeds, or the terminal drops the packet and goes back to idle if the $L + 1$-th retransmission remains unsuccessful. Accordingly, fast resolution of collisions in RAP transmissions is expected to enable fast bandwidth reservations. Additionally in LTE, the window size of the UB algorithm, which is denoted by $U$ hereafter and is assigned by the BS in the first RAR message, can be dynamically chosen according to network congestion. Thus, it is expected that controlling $U$ appropriately can resolve collisions efficiently under some time-varying traffic condition.

### 3.1.3 Binary Exponential Backoff Algorithm in IEEE 802.16

Upon an unsuccessful packet transmission, the BEB algorithm increases the backoff window exponentially by doubling its size. Let $W_i$ denote the backoff window size of the $i$-th backoff stage, which is determined by

$$W_i = 2^i W_0 \quad \text{for} \quad 0 \leq i \leq K$$

(3.1)

where $W_0$ is the minimum backoff window size. At the $i$-th backoff stage, the backlogged terminal randomly chooses a backoff counter in the range $[0, W_i - 1]$ for $0 \leq i \leq K$. Once chosen, the backoff counter is decremented every slot as in UB algorithm. When it reaches zero, the terminal transmits its packet. Upon transmission failure, the terminal increments the backoff stage by one and chooses the next backoff counter again. Although the retry
limit, \( L \), is independent of \( K \) in [30], we assume here that the BEB algorithm retries \( L \) times additionally at the \( K \)-th stage. Upon the retransmission failure after total of \( L + 1 \) retransmissions at the \( K \)-th stage, it drops the packet. Thus, the maximum of total number of retransmissions is \( K + L + 1 \).

### 3.1.4 Access Prioritization by Persistence Value

In order to provide access priority, UMTS includes eight access priority classes [31], each of which has a different persistence value \( p = f_i(N) = s_i \cdot 2^{(N-1)} \) for \( 0 \leq i \leq 7 \). Note that the value of \( i \) can indicate a certain partition of the RACHs. The persistence scaling factor \( s_i \) is assumed to be a value between 0.2 and 0.9 by step of 0.1, and the dynamic persistence level \( N \) from 1 to 8 is broadcast regularly. If the persistence scaling factors are not broadcast, a default value of one is assumed. Once the backoff counter reaches zero, the terminal draws a real random number between 0 and 1. If this number is less than \( f_i(N) \), the terminal transmits a packet. Otherwise, it repeats this test until transmission. Note that two successive tests are separated by a certain time period called the \( T_2 \) timer.

We apply this access prioritization scheme to both UB and BEB algorithms to facilitate comparisons.

### 3.2 Performance Evaluation

Before proceeding with our analysis, we summarize in Table 3.1 the notations defined previously, which will be used in this section.

In the system model, we assume a finite population of \( M \) terminals. Each terminal can hold only one packet for transmission or retransmission until it is successfully received. A
Table 3.1: Nomenclature for analysis

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Number of orthogonal random access codes (preambles)</td>
</tr>
<tr>
<td>M</td>
<td>Total number of terminals</td>
</tr>
<tr>
<td>L</td>
<td>Retry limit of retransmissions</td>
</tr>
<tr>
<td>U</td>
<td>Window size of uniform backoff algorithm</td>
</tr>
<tr>
<td>K</td>
<td>Maximum stage of BEB algorithm, maximum window $W_K = 2^K - 1$</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>Probability that a packet is generated at an idle terminal every slot, i.e. Bernoulli trial with probability $\epsilon$</td>
</tr>
<tr>
<td>$p_s$</td>
<td>Transmission success probability of a packet</td>
</tr>
<tr>
<td>$p_r$</td>
<td>Retransmission probability of a packet</td>
</tr>
<tr>
<td>$p_c$</td>
<td>Code-collision probability of a packet, $p_c = 1 - p_s$</td>
</tr>
<tr>
<td>$p$</td>
<td>Persistence probability (value)</td>
</tr>
</tbody>
</table>

terminal that has no packet to transmit is said to be idle. An idle terminal generates a packet in every slot based on a Bernoulli trial with probability $\epsilon$, i.e., unsaturated traffic condition. When a terminal has a packet to send, it immediately becomes backlogged by performing the backoff algorithm, i.e., it employs a DFT protocol. We finally assume that terminals can get feedback information for the RAP transmitted, e.g., success or failure, from the BS right before the next slot without any error, i.e., no feedback delay or error is taken into account.

3.2.1 Channel Model

Suppose that a terminal among $M$ terminals (re)transmits a packet with probability one. Then, its transmission success probability $p_s$ can be written as

$$p_s = \sum_{n=0}^{M-1} \frac{1}{n+1} \left[ \sum_{k=0}^{n} (k+1)\Lambda(k+1|n+1,P) \right] \Omega(n,M-1,p_r),$$

(3.2)

in which the term in the bracket indicates the conditional throughput given $n+1$ packets transmitted (including the packet of our interest). By dividing this term by $n+1$, we get
the conditional transmission success probability given \( n + 1 \) packets. For the code-collision under perfect orthogonality condition among RAPs, we have \( C_{k+1,n+1} = \Lambda(k + 1|n + 1, P) \). Other channel models can be analyzed by replacing \( C_{k+1,n+1} \) accordingly. Throughout this chapter, we denote the binomial probability distribution function with parameters \( b, L \) and \( p \) by \( \Omega(b, L, p) = \binom{L}{b} p^b (1 - p)^{L-b} \). To see the probability that a terminal chooses a unique RAP, we define as \( \Lambda(k|v, N_c) \) the probability that \( k \) among \( v \) RAPs simultaneously transmitted are uniquely chosen among total of \( N_c \) RAPs, i.e., successful transmission probability of \( k \) out of \( v \) packets. Recall that backlogged terminals select one of \( P \) RAPs and transmit it. Then, we can obtain recursively \( \Lambda(k|v, P) \) by \[ \Lambda(k|v, P) = \sum_{m=0}^{v} \Omega(m, v, 1/P) \Lambda(k - I(m-1)|v-m, P-1) \] in which the indicator function \( I(x) \) equals one at \( x = 0 \) or zero elsewhere. Eq. (3.3) means that, by examining each of \( v \) packets transmitted in order, if a unique RAP is transmitted with probability \( \Omega(1, v, 1/P) \), it is counted as one success and we decrement \( k, v \) and \( P \) by one. Otherwise, we subtract from \( v \) a group of \( m \) duplicated RAPs, which occurs with probability \( \Omega(m, v, 1/P) \) for \( m \neq 1 \). Eq. (3.3) has the following initial conditions:

\[
\begin{align*}
\Lambda(0|1, 0) &= 1, & \Lambda(0|1, P) &= 0 \text{ for } P \geq 1, & \Lambda(1|1, 0) &= 0, \\
\Lambda(1|1, P) &= 1 \text{ for } P \geq 1, & \Lambda(1|v, 1) &= 0 \text{ for } v \neq 1, \\
\Lambda(k|v, P) &= 0 \text{ for } \min(v, P) < k, \\
\Lambda(-1|v, P) &= 0 \text{ for any } v \text{ and } P.
\end{align*}
\]

If \( P = 1 \), the channel is reduced to an S-ALOHA channel. In fact, \( \Lambda(k|v, P) \) is related to the classical occupancy problem in [57], in which more general cases are considered.
3.2.2 Approximation on Correlated Backoff Processes

In order to model $M$ terminals’ backoff algorithm (either UB or BEB) we should consider a multidimensional Markov process, $b^t = [b^t_1, b^t_2, \ldots, b^t_M]$, in which $b^t_i$ denotes the backoff process of the $i$-th terminal at time $t$. In BEB algorithm we can have $-1 \leq b^t_i \leq K$ according to (3.1). Note that $-1$ indicates the state of a terminal having no packet to send. The stationary behaviour of $b^t$ can be written by

$$\lim_{t \to \infty} \Pr[b^t_1 = i_1, \ldots, b^t_M = i_M] \quad \text{for} \quad -1 \leq i_j \leq K, \ 1 \leq j \leq M. \quad (3.4)$$

With slight abuse of notation, we denote by $b_i$ the stationary behaviour of the $i$-th terminal’s backoff process. If we are interested in the probability that $n$ terminals have no packet while $M - n$ terminals are in the backoff process, based on i.i.d. assumption on terminals’ backoff processes, we can write this as

$$M!/(n!(M-n)!)\Pr[b_j = -1]^n (1 - \Pr[b_j = -1])^{M-n},$$

in which any terminal among $M$ terminals can be the $j$-th terminal. This can be extended by a multinomial distribution for the event that $n_i$ terminals are in the $i$-th backoff stage for $-1 \leq i \leq K$.

3.2.3 Uniform Backoff Algorithm

Define $\pi_E$ and $\pi_{i,j}$ for $0 \leq i \leq L$ and $0 \leq j \leq U - 1$ as the stationary state probability that a terminal is idle and that a terminal is at the $j$-th backoff counter value of the $i$-th backoff stage, respectively. Then, we can write the state balance equation for $\pi_E$ as

$$\pi_E = (1 - \epsilon)\pi_E + p \cdot p_s \sum_{i=0}^{L-1} \pi_{i,0} + p\pi_{L,0}. \quad (3.5)$$
We can write \( \pi_{0,j} \) for \( 0 \leq j \leq U - 1 \) as

\[
\begin{align*}
\pi_{0,0} &= (1 - p) \pi_{0,0} + \epsilon \frac{p}{U} \pi_E + \pi_{0,1}, \\
\pi_{0,j} &= \frac{\epsilon}{U} \pi_E + \pi_{0,j+1} \quad \text{for} \quad 1 \leq j \leq U - 2, \quad \text{and} \quad \pi_{0,U-1} = \frac{\epsilon}{U} \pi_E. \quad (3.6)
\end{align*}
\]

We can also write \( \pi_{i,j} \) for \( 1 \leq i \leq L \) and \( 0 \leq j \leq U - 1 \) as

\[
\begin{align*}
\pi_{i,0} &= (1 - p) \pi_{i,0} + \frac{p p_c}{U} \pi_{i-1,0} + \pi_{i,j+1}, \\
\pi_{i,j} &= \frac{p p_c}{U} \pi_{i-1,0} + \pi_{i,j+1} \quad \text{for} \quad 1 \leq j \leq U - 2 \quad \text{and} \quad \pi_{i,U-1} = \frac{p p_c}{U} \pi_{i-1,0}. \quad (3.7)
\end{align*}
\]

Using (3.6) and (3.7), we rewrite \( \pi_{i,j} \) for \( 0 \leq i \leq L \) as

\[
\begin{align*}
\pi_{i,0} &= p_i \frac{\epsilon}{p} \pi_E \quad \text{and} \quad \pi_{i,j} = p_i \frac{U - j}{U} \pi_E \quad \text{for} \quad 1 \leq j \leq U - 1. \quad (3.8)
\end{align*}
\]

Using the normalization condition \( \pi_E + \sum_{i=0}^{L} \sum_{j=0}^{U-1} \pi_{i,j} = 1 \), we can get \( \pi_E \) as

\[
\pi_E = \left[ 1 + \epsilon \left( \frac{1}{p} + \frac{U - 1}{2} \right) \frac{1 - p_c^{L+1}}{1 - p_c} \right]^{-1}. \quad (3.9)
\]

Then, the retransmission probability \( p_r \) can be expressed as

\[
p_r = p \sum_{i=0}^{L} \pi_{i,0} = \frac{1 - p_c^{L+1}}{p_s} \epsilon \pi_E. \quad (3.10)
\]

When \( L \to \infty \) in (3.9) and (3.10), we have the case without retry limit.

To find the solution of \( p_r \) satisfying (3.2) and (3.10) simultaneously, we construct a
function \( f_U(p_r) \) with (3.2) and (3.10) as

\[
f_U(p_r) = p_r - \frac{1 - (1 - p_s)^{L+1}}{p_s} \epsilon \pi_E. \tag{3.11}
\]

Note that (3.11) is a function of \( p_r \), when we substitute (3.2) into (3.11) for \( p_s \). The solution of \( p_r \) satisfying \( f_U(p_r) = 0 \) is numerically found for \( 0 < p_r < 1 \). If a unique \( p_r \) is found, we have a stable system. Otherwise, the system is said to be either bistable or unstable. We shall discuss this in Section 3.2.5.

As a performance metric, the system throughput \( \tau \) in packets/slot can be obtained as

\[
\tau = \sum_{n=0}^{M} \sum_{k=0}^{n} k \Lambda(k|n, P) \Omega(n, M, p_r). \tag{3.12}
\]

We then obtain the throughput for this channel, normalized with respect to the total number of codes, by \( \tau_N = \tau/P \) (packets/slot/code). The packet dropping probability \( P_d \), due to reaching the retry limit, can be expressed as

\[
P_d = p_c p \pi_{L,0} = p_c^{L+1} \epsilon \pi_E, \tag{3.13}
\]

where \( p_c \) is the code-collision probability. Note that \( P_d \) goes to zero as \( L \to \infty \).

In order to obtain the mean packet transmission delay, i.e., access delay, some previous work [19][51][52] used the mean-value analysis. Instead, we use an absorbing Markov chain technique. First, we define the packet retransmission delay \( D_{i,j} \) as the time for an absorbing process starting from the initial backoff counter value \( j \) at the \( i \)-th backoff stage to reach (i.e., be absorbed into) the idle state. As usual, we denote the expectation of a random variable by an overline, thus \( \overline{D_{i,j}} = E[D_{i,j}] \). Note that, in the \( i \)-th backoff stage for \( 0 \leq i \leq L \), the initial backoff counter value \( j \) is chosen uniformly from \( 0 \leq j \leq U - 1 \).
Define the mean access delay $\overline{d}$ as the mean time taken for a terminal to transmit a packet successfully, after it is generated. We can write $\overline{d}$ as

$$\overline{d} = \frac{1}{U} \sum_{j=0}^{U-1} \overline{D}_{0,j} = \overline{D}_{0,0} + \frac{U - 1}{2}, \quad (3.14)$$

in which we have used the following relation

$$\overline{D}_{i,j} = E[1 + \overline{D}_{i,j-1}] = j + \overline{D}_{i,0} \quad (3.15)$$

for $0 \leq i \leq L$ and $1 \leq j \leq U - 1$. The first equality in (3.15) indicates that it always takes one slot for the absorbing process to go down by one to the next backoff counter value with probability one. Since the higher layer of a terminal will be informed upon packet dropping, we include the delay of the packets successfully transmitted and dropped in (3.14). We can write $\overline{D}_{i,0}$ for $0 \leq i \leq L - 1$ as

$$\overline{D}_{i,0} = pp_s + (1 - p) E[1 + D_{i,0}] + \frac{pp_c}{U} E\left[\sum_{j=0}^{U-1} (1 + D_{i+1,j})\right] = \frac{1}{p} + p_c \left( \overline{D}_{i+1,0} + \frac{U - 1}{2} \right) = \left( \frac{1}{p} + p_c \frac{U - 1}{2} \right) \frac{1 - p_c^{L-i}}{1 - p_c} + p_c^{L-i}, \quad (3.16)$$

which indicates that the retransmission delay takes one slot with probability $p \times p_s$, when the backoff counter is zero. Further, when the persistence test is not passed with probability $1 - p$, it takes one more slot before the next persistence test. When a packet is retransmitted, but unsuccessful, the retransmission delay takes one slot and additional delay depending on the next backoff counter. We have

$$\overline{D}_{L,0} = 1 \cdot p + (1 - p) \left( 1 + \overline{D}_{L,0} \right), \quad (3.17)$$
which is simplified as $\overline{D}_{L,0} = 1/p$. Using (3.15) and (3.17), one can get $\overline{D}_{i,0}$ for $0 \leq i \leq L - 1$ by

$$
\overline{D}_{i,0} = \frac{1}{1-p_c} \left( \frac{1}{p} (1 - p_c^{L-i+1}) + p_c \frac{U-1}{2} (1 - p_c^{L-i}) \right).
$$

(3.18)

In order to obtain the access delay variance, let $\overline{d}^2$ denote the second moment of the access delay, which is expressed by

$$
\overline{d}^2 = \frac{1}{U} E \left[ \sum_{j=0}^{U-1} D_{0,j}^2 \right] = \frac{1}{U} \sum_{j=0}^{U-1} \overline{D}_{0,j}^2 = \overline{D}_{0,0}^2 + (U - 1) \overline{D}_{0,0} + \frac{2U^2 + 3U - 5}{6} \tag{3.19}
$$

where we have used

$$
\overline{D}_{i,j}^2 = E[(1 + D_{i,j-1})^2] = j + 2 \sum_{k=0}^{j-1} \overline{D}_{i,k} + \overline{D}_{i,0}^2 = 2j \overline{D}_{i,0} + j^2 + \overline{D}_{i,0}^2,
$$

(3.20)

for $0 \leq i \leq L$ and $1 \leq j \leq U - 1$. In the above, we can write $\overline{D}_{i,0}^2$ for $0 \leq i \leq L - 1$ as

$$
\overline{D}_{i,0}^2 = pp_c + (1-p)E \left[ (1 + D_{i,0})^2 \right] + \frac{pp_c}{U} E \left[ \sum_{j=0}^{U-1} (1 + D_{i+1,j})^2 \right]
$$

$$
= pp_c + (1-p) \left( 1 + 2\overline{D}_{i,0} + \overline{D}_{i,0}^2 \right) + \frac{pp_c}{U} \left( U + 2 \sum_{j=0}^{U-1} \overline{D}_{i+1,j} + \sum_{j=0}^{U-1} \overline{D}_{i+1,j}^2 \right),
$$

(3.21)

which is the second moment of (3.16). By substituting (3.15) and (3.20) into (3.21), we can rewrite

$$
\overline{D}_{i,0}^2 = \frac{1}{p} \left[ \left( 1 + pp_c \frac{2U^2 + 3U - 5}{6} \right) + 2(1-p)\overline{D}_{i,0} + pp_c \left( (U+1)\overline{D}_{i+1,0} + \overline{D}_{i+1,0}^2 \right) \right].
$$

(3.22)
Finally we have

\[ \overline{D^2_{L,0}} = 1^2 \cdot p + (1 - p)E \left[ (1 + D_{L,0})^2 \right], \]  

(3.23)

which is simplified as \( \overline{D^2_{L,0}} = 1/p \left( 1 + 2(1 - p)\overline{D_{L,0}} \right) \). The variance of the packet access delay is obtained by \( \overline{d^2} = \overline{D^2} - \overline{D}^2 \).

### 3.2.4 Binary Exponential Backoff Algorithm

Let \( \hat{\pi}_E \) and \( \hat{\pi}_{i,j} \) denote the stationary state probability that a terminal has no packet and that a terminal is at the \( j \)-th backoff time of the \( i \)-th backoff stage, respectively, in the BEB algorithm. We can write a balance equation for \( \hat{\pi}_E \) as

\[ \hat{\pi}_E = (1 - \epsilon)\hat{\pi}_E + pp_s \sum_{i=0}^{K-1} \hat{\pi}_{0,i} + \hat{\pi}_{0,K}. \]  

(3.24)

Substituting \( W_0 \) for \( U \) in (3.6), we can get the balance equations for \( i = 0 \). We express \( \hat{\pi}_{i,j} \) for \( 0 \leq i \leq K - 1 \) and \( 0 \leq j \leq W_i - 1 \) as

\[ \hat{\pi}_{i,0} = (1 - p)\hat{\pi}_{i,0} + \frac{pp_c}{W_i} \hat{\pi}_{i-1,0} + \hat{\pi}_{i,1} \]

\[ \hat{\pi}_{i,j} = \frac{pp_c}{W_i} \hat{\pi}_{i-1,0} + \hat{\pi}_{i,j+1} \quad \text{for } 1 \leq j \leq W_i - 2 \quad \text{and} \quad \hat{\pi}_{i,W_i-1} = \frac{pp_c}{W_i} \hat{\pi}_{i-1,0}. \]

Instead of \( W_i \) in (3.25), we use \( W_K \) for \( \hat{\pi}_{i,j} \) for \( K \leq i \leq K + L \). For \( L = 0 \), (3.25) is valid for \( i = K \). We can simplify (3.25) for \( 0 \leq i \leq K - 1 \) as

\[ \hat{\pi}_{i,0} = p_c \frac{\epsilon}{p} \hat{\pi}_E \quad \text{and} \quad \hat{\pi}_{i,j} = p_c \frac{W_i - j}{W_i} \epsilon \hat{\pi}_E \quad \text{for } 1 \leq j \leq W_i - 1. \]  

(3.25)
We also have \( \hat{\pi}_{i,j} \) for \( K \leq i \leq K + L 

\[
\hat{\pi}_{i,0} = p_i^e \epsilon \hat{\pi}_E \quad \text{and} \quad \hat{\pi}_{i,j} = \frac{p_i^e (W_K - j)}{W_K} \epsilon \hat{\pi}_E \quad \text{for} \quad 1 \leq j \leq W_K - 1.
\] (3.26)

By using the following normalization condition

\[
\hat{\pi}_E + \sum_{i=0}^{K-1} \sum_{j=0}^{W_i-1} \hat{\pi}_{i,j} + \sum_{i=K}^{K+L} \sum_{j=0}^{W_K-1} \hat{\pi}_{i,j} = 1,
\] (3.27)

we can get

\[
\hat{\pi}_E = \left[ 1 + \epsilon \left\{ 1 - \frac{p_c^{K+L+1}}{p(1-p_c)} + \frac{p_c}{2} \left( \frac{2W_0(1-(2p_c)^{K-1})}{1-2p_c} \right) - \frac{1 - p_c^{K-1}}{1-p_c} \right) + \frac{p_c^K (1-p_c^{L+1})}{1-p_c} \left( \frac{W_K-1}{2} \right) \right\}^{-1}.
\] (3.28)

Denote by \( \hat{p}_r \) the retransmission probability for BEB algorithm, which can be expressed as

\[
p_r = p \sum_{i=0}^{K+L} \hat{\pi}_{i,0} = \frac{1 - p_c^{K+L+1}}{p_s} \epsilon \hat{\pi}_E.
\] (3.29)

Since (3.2) is also valid for the BEB algorithm, as in (3.11), we can construct \( f_B(p_r) \) with (3.2) and (3.29) as

\[
f_B(p_r) = p_r - \frac{1 - p_c^{K+L+1}}{p_s} \epsilon \hat{\pi}_E
\] (3.30)

and then solve for \( f_B(p_r) = 0 \). The system throughput is readily obtained by (3.12). The packet dropping probability can be obtained as

\[
P_d = p_d \hat{\pi}_{K+L,0} = p_c^{K+L+1} \epsilon \hat{\pi}_E.
\] (3.31)
Chapter 3. Existing Backoff Algorithms for Single-Buffered Terminals

As we have seen before in (3.13), either large $K$ or $L$ makes $P_d$ vanish.

To analyze the delay performance of BEB algorithm, let $D_{i,j}$ denote the random variable of the packet retransmission delay starting from the $j$-th backoff counter of the $i$-th backoff window of BEB algorithm. Then, similarly to (3.14), the mean access delay, $\bar{d}$, is expressed as

$$
\bar{d} = D_{0,0} + \frac{W_0 - 1}{2}.
$$

(3.32)

As in (3.16), we can express $\overline{D_{i,0}}$ for $0 \leq i \leq K + L - 1$ by

$$
\overline{D_{i,0}} = pp_s + (1 - p)E[1 + D_{i,0}] + \frac{pp_c}{W_{\min(i+1,K)}}E\left[\sum_{j=0}^{W_{\min(i+1,K)}-1} (1 + D_{i+1,j})\right]
$$

$$
= \frac{1}{p} + pc \left(\overline{D_{i+1,0}} + \frac{W_{\min(i+1,K)} - 1}{2}\right),
$$

(3.33)

in which $\min(a, b)$ denotes the minimum of $a$ and $b$. In (3.33) we have used $\overline{D_{t,j}} = j + \overline{D_{t,0}}$ and (3.1). Then, we have

$$
\overline{D_{i,0}} = \left(\frac{1}{p} - \frac{pc}{2}\right) \frac{1 - p_c^{K-1-i}}{1 - p_c} + pc W_0 \frac{1 - (2p_c)^{K-1-i}}{1 - 2p_c} + \overline{D_{K-1,0}} p_c^{K-1-i}.
$$

(3.34)

for $0 \leq i \leq K - 2$. We can write $\overline{D_{i,0}}$ for $K - 1 \leq i \leq K + L - 1$ as

$$
\overline{D_{i,0}} = \frac{1}{1 - p_c} \left(\frac{1}{p} \left(1 - p_c^{K+L-i+1}\right) + pc \frac{W_K - 1}{2} \left(1 - p_c^{K+L-i}\right)\right)
$$

(3.35)

where we have used $\overline{D_{K+L,0}} = 1/p$. For both UB and BEB algorithms, as $L \to \infty$, the respective derivations correspond to the algorithm without retry limit.
Now, the second moment of the access delay of the BEB algorithm can be expressed as

$$d^2 = \frac{1}{W_0} \sum_{j=0}^{W_0-1} D_{0,j}^2 = D_{0,0}^2 + \bar{D}_{0,0} + 0.5.$$  \hspace{1cm} (3.36)

As in (3.21), we can write $\bar{D}_{i,0}$ for $0 \leq i \leq K + L - 1$ as

$$\bar{D}_{i,0} = pp_s + (1 - p)E[(1 + D_{i,0})^2] + \frac{pp_c}{W_{\text{min}(i+1,K)}} E \left[ \sum_{j=0}^{W_{\text{min}(i+1,K)} - 1} (1 + D_{i+1,j})^2 \right].$$  \hspace{1cm} (3.37)

After some lengthy manipulation with $\bar{D}_{i,j} = 2j\bar{D}_{i,0} + j^2 + \bar{D}_{i,0}$ for $0 \leq i \leq K + L$ and $1 \leq j \leq W_{\text{min}(i+1,K)} - 1$, we can rearrange (3.37) as

$$\bar{D}_{i,0} = \frac{1}{p} \left[ \left( 1 + pp_c \frac{2W_{\text{min}(i+1,K)}^2 + 3W_{\text{min}(i+1,K)} - 5}{6} \right) + 2(1 - p)\bar{D}_{i,0} \right] + pp_c \left( (W_{\text{min}(i+1,K)} + 1)\bar{D}_{i+1,0} + \bar{D}_{i+1,0} \right).$$  \hspace{1cm} (3.38)

Finally we have $\bar{D}_{K+L,0} = 1/p \left( 1 + 2(1 - p)\bar{D}_{K+L,0} \right)$.

### 3.2.5 Stability

We discuss now the stability of these backoff algorithms especially for S-ALOHA channel, i.e., $P = 1$. Accordingly in what follows we use $p_s = (1 - p_r)^{M-1}$. For the system with $P > 1$, the following argument can be similarly applied. However, it might not seem obvious due to $C_{k,n}$ in (3.2). The stability of either UB or BEB algorithm can be guaranteed, when a unique $p_r$ exists that solves (3.11) or (3.30). If multiple solutions are found, the system is said to be bistable, i.e., multiple equilibrium points. Finally, when no solution is found, the system is said to be unstable, i.e., backlog grows without bound. It is not difficult to prove the existence of the solution by showing $f_U(0) < 0$ and $f_U(p_r) > 0$ for some $0 < p_r \leq 1$ in
UB algorithm. At $p_r = 0$, we get

$$f_U(0) = -1/(1/\epsilon + (1/p + 0.5(U - 1))). \quad (3.39)$$

As $p_r \to 1$, $p_s$ vanishes exponentially ($p_c \to 1$) due to $M$ in (3.2). Then, we approximately write

$$\lim_{p_r \to 1} f_U(p_r) = p_r - \frac{\delta_U}{1/\epsilon + \delta_U(1/p + 0.5(U - 1))} \quad (3.40)$$

with $\delta_U \equiv L + 1$. If (3.40) becomes positive for $0 < p_r \leq 1$, it implies an odd number of solutions of $f_U(p_r)$. Note that whether (3.40) is positive is mainly determined by $U$. Even for $L \to \infty$ (no retry limit) the existence of the solution can be guaranteed, when a large $U$ relative to $M$ is chosen in (3.40).

By the same token, for the BEB algorithm we have at $p_r = 0$

$$f_B(0) = -\epsilon p/(p + \epsilon) < 0. \quad (3.41)$$

As $p_r \to 1$, we can write

$$\lim_{p_r \to 1} f_B(p_r) = p_r - \frac{2\delta_B}{2/\epsilon + W_0 2^{K(L + 1)} + 2\delta_B/p - (K + L + 2W_0)} \quad (3.42)$$

with $\delta_B \equiv K + L + 1$. As before, we can have odd number of solutions of $f_B(p_r)$, if $f_B(p_r) > 0$ for $0 < p_r \leq 1$. Similar to $U$ in (3.40), whether (3.42) is positive depends on $K$. Even some small $K$ can guarantee that (3.42) is positive due to the exponential term in the denominator of (3.42). Letting $\epsilon \to 1$ and $p \to 1$ and ignoring some negligible terms
in (3.42), we have approximately

\[ f_B(p_r) \approx p_r - \frac{\delta_B}{W_0 2^{K-1}(L+1)}. \]  

(3.43)

For the BEB algorithm \((W_0 = 2)\) under saturated traffic condition \((\epsilon \to 1)\), the solution always exists for some \(K\).

Until now, we have shown the existence of the solutions of \(f_U(p_r)\) and \(f_B(p_r)\). It is not easy to prove their uniqueness due to nonlinearity of \(f_U(p_r)\) and \(f_B(p_r)\). It is shown in [37][48][49][50] that applying a retry limit, i.e., limiting feedback to the channel input, can stabilize the system with a probability distribution-based backoff algorithm. If we show the stability of the system with window-based backoff algorithm without retry limit, i.e., \(L \to \infty\), it is reasonable to expect that the system with retry limit is also stable. From (3.9) and (3.11) we can rewrite the solution of \(f_U(p_r)\) as

\[ p_r = \frac{1 - (1 - p_s)^{L+1}}{p_s (1 + \epsilon \left( \frac{1}{p} + \frac{U-1}{2} \right) \frac{1 - p_c^{L+1}}{p_s}) \epsilon \left( 1 - p_r - p_r \right)}, \]  

(3.44)

in which the terms with \(L\) vanishes, as \(L \to \infty\). We can simplify (3.44) as

\[ p_r = \frac{\epsilon}{p_s \epsilon (1/p + 0.5(U - 1))}. \]  

(3.45)

Since \(p_s = (1 - p_r)^{M-1}\) is negligible for \(\epsilon \to 1\) (i.e., saturated traffic condition) and large \(M\), a unique \(p_r\) can be determined, i.e. \(p_r = 2/(U + 1)\) for \(p = 1\). Although \(p_r\) can be affected by \((1 - p_r)^{M-1}\) for unsaturated traffic condition, the uniqueness of \(p_r\) can be found for large \(M\). Similarly, letting \(L \to \infty\), we can approximately write the solution of \(f_B(p_r)\)
for the BEB algorithm in (3.30) as

$$p_r = \frac{\epsilon}{p_s + \epsilon (1/p + p_r p_s W_0 (2^{K-1} - 1) + 0.5p_c^K (W_K - 1))}. \tag{3.46}$$

From (3.46) we get a unique $p_r = 2/(W_K - 1)$, since we have $p_s \to 0 \ (p_c \to 1)$ for saturated traffic and large $M$. Thus, window-based backoff algorithms, such as UB and BEB algorithms (even with large $L$, i.e., many retransmissions) show only either stable or unstable behaviour in accordance with $U$ (for the UB algorithm) or $K$ (for the BEB algorithm) and $\epsilon$. Note that the bistability observed in [37][48][49][50] results from an approximate model of some window-based backoff algorithms, i.e., based on a (truncated) geometric distribution, not from window-based backoff algorithms.

### 3.2.6 Access Prioritization Scheme

Consider $C$ access priority classes, each of which consists of $M^{[\nu]}$ terminals for $1 \leq \nu \leq C$. Hereafter, the superscript of a parameter indicates the access priority class. We can achieve access prioritization by differentiating $p, U, K, L$, or a combination of them. Let $p^{[\nu]}_s$ and $p^{[\nu]}_r$ denote the transmission success and retransmission probability of a terminal belonging to the $\nu$-th access priority class. Then, we can obtain $p^{[\nu]}_s$ of both UB and BEB algorithms by

$$p^{[\nu]}_s = \sum_{n^{[1]}=0}^{M^{[1]}-1} \cdots \sum_{n^{[\nu]}=0}^{M^{[\nu]}-1} \cdots \sum_{n^{[C]}=0}^{M^{[C]}-1} \frac{1}{\tilde{n} + 1} \sum_{k^{[1]}=0}^{n^{[1]}} \cdots \sum_{k^{[C]}=0}^{n^{[C]}} (\tilde{k} + 1) C_{\tilde{k}+1, \tilde{n}+1} \Omega(n^{[\nu]}, M^{[\nu]} - 1, p^{[\nu]}_r)$$

$$\times \prod_{i=1, i \neq \nu}^C \Omega(n^{[i]}, M^{[i]}, p^{[i]}_r) \tag{3.47}$$
Table 3.2: Nomenclature for DWA algorithm

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_E(t)$</td>
<td>Number of RAPs not transmitted at time $t$</td>
</tr>
<tr>
<td>$C_S(t)$</td>
<td>Number of RAPs successfully received at time $t$</td>
</tr>
<tr>
<td>$B(t)$</td>
<td>The estimated number of backlogged terminals at time $t$</td>
</tr>
<tr>
<td>$\lambda_t$</td>
<td>The estimated number of new packet arrivals at time $t$</td>
</tr>
<tr>
<td>$U_t$</td>
<td>The window size is assigned at time $t$</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Smoothing factor</td>
</tr>
</tbody>
</table>

where $\tilde{k} = \sum_{\nu=1}^{C} k^{[\nu]}$ and $\tilde{n} = \sum_{\nu=1}^{C} n^{[\nu]}$. For S-ALOHA channel with $P = 1$, we readily get

$$p_s^{[\nu]} = (1 - p_r^{[\nu]})^{M^{[\nu]}-1} \prod_{i=1, i \neq \nu}^{C} (1 - p_r^{[i]})^{M^{[i]}}. \quad (3.48)$$

We can obtain $p_r^{[\nu]}$ of both backoff algorithms for $1 \leq \nu \leq C$ by substituting (3.48) into either (3.10) or (3.29), respectively. Then, the system throughput can be obtained as

$$\tau = \sum_{n^{[1]}=0}^{M^{[1]}} \cdots \sum_{n^{[C]}=0}^{M^{[C]}} \sum_{k^{[1]}=0}^{n^{[1]}} \cdots \sum_{k^{[C]}=0}^{n^{[C]}} \tilde{k} \prod_{\nu=1}^{C} \Lambda(k^{[\nu]}, n^{[\nu]}, P) \Omega(n^{[\nu]}, M^{[\nu]}, p_r^{[\nu]})$$

where the first summation holds for all possible combinations of $n^{[\nu]}$ for $1 \leq \nu \leq C$. For $P = 1$, we have $\tau = \sum_{\nu=1}^{C} \tau^{[\nu]}$, in which $\tau^{[\nu]}$ is expressed as

$$\tau^{[\nu]} = \Omega(1, M^{[\nu]}, p_r^{[\nu]}) \prod_{i=1, i \neq \nu}^{C} \Omega(0, M^{[i]}, p_r^{[i]}) \quad (3.49)$$

We can also obtain the mean and variance of a packet retransmission delay by using (3.14), (3.19), (3.32) and (3.36) for both backoff algorithms, respectively.
Figure 3.1: UB algorithm with different $M$ and $P$. 

(a) Normalized throughput

(b) Mean of retransmission delay.

(c) Variance of retransmission delay.

(d) Packet dropping probability.
3.2.7 Dynamic Window Assignment Algorithm

The proposed dynamic window assignment algorithm is given as Algorithm 3.1, in which the variables are defined in Table 3.2. We assume that a BS uses a bank of matched filters, each of which tracks a specific RAP, so that in each time slot it can determine respectively the number of RAPs not transmitted and the number of RAPs received, $C_E(t)$ and $C_S(t)$. 

In the 2nd line of Algorithm 3.1, $P - C_E(t) - C_S(t)$ is the number of RAPs unsuccessfully transmitted. The underlying idea of this dynamic window assignment algorithm is a simple extension of the existing Bayesian algorithm in multidimensional packet reservation multiple access [55], in which each terminal estimates $B(t)$ first and then determines the retransmission probability by $p_r = \min(1, R/B(t))$. Here, $R$ is the number of orthogonal resources, e.g., time slots in [55]. In the UB algorithm, we expect $1/U_t$ to approximate $p_r$ in the mean value. If there are $P$ RAPs and $B(t)$ backlogged terminals, we can determine $U_t = \max(1, B(t)/P)$ under the assumption of perfectly orthogonal RAPs. Note that $\max(a, b)$ denotes the maximum of $a$ and $b$. To this end, we estimate $B(t)$ in the 7-th line as the same way in [55]. For $P = 1$, the backlog estimation is identical to Rivest’s algorithm [10]. In addition to backlog size estimation, we assign averaged $U_t$ as shown in the 8-th line, in which $\lceil x \rceil$ denotes the smallest integer not less than $x$. In practical implementation, $U_t$ can be broadcast to terminals right before the next slot, or it is given via the first RAR message in LTE. Note that, if $P_d$ is not sufficiently small, the estimation on the number of backlogged terminals becomes inexact due to packet dropping.
Chapter 3. Existing Backoff Algorithms for Single-Buffered Terminals

Figure 3.2: Comparison of UB and BEB algorithms with different $U$ and $L$.
**Chapter 3. Existing Backoff Algorithms for Single-Buffered Terminals**

**Algorithm 3.1 - Dynamic Window Assignment Algorithm**

1: Initialize $\lambda_0 = 0$, $U_0 = 20$, $B(0) = 0$ and do the following at $t = 1, 2, \ldots$

2: if $P - C_E(t) - C_S(t) = P$ then
3: \[ \lambda_t = Pe^{-1}. \]
4: else
5: \[ \lambda_t = 0.9\lambda_t + 0.1C_S(t). \]
6: end if

7: $B(t) = \max(0, B(t-1) + (e-2)^{-1}(P - C_E(t) - C_S(t)) - (C_E(t) + C_S(t))) + \lambda_t.$
8: $U_t = \lceil \beta U_{t-1} + (1 - \beta)B(t)/P \rceil.$

### 3.3 Numerical Studies

In order to validate our analysis, we built a simulation program with MATLAB. Each simulation is carried out for 20000 slots. In the following, we use $W_0 = 2$ in the EB algorithm. In Figs. 3.1(a) to 3.7(d), the curves are obtained numerically from the equations given in the above analysis, while the symbols indicated the corresponding simulation results.

1. **The effect of multichannel code-collisions**: We first examine the effect of code-collision in the UB algorithm by varying $P$ in Figs. 3.1(a)-3.1(d) where we have $p = 1$, $U = 15$ and $L = 5$. As expected, we can see that large $P$ provides higher $\tau_N$ for large $\epsilon$ given $M$ in Fig. 3.1(a) and better performances, e.g. $d$ in Fig. 3.1(b). However, we check that the performances of $M = 160$ and $P = 4$ are identical to those of $M = 40$ and $P = 1$. Therefore, it can be concluded that, although a large $P$ can increase the RACH capacity, intrinsic properties of the system performances do not change under perfect orthogonality assumption.

2. **The effect of $U$, $K$ and $L$ in UB and EB algorithms**: We examine UB and EB algorithms without priority classes, i.e., $p = 1, C = 1$ in S-ALOHA channel ($P = 1$) in order to see their own characteristics in accordance with the window size $U$, $K$ and with retry limit $L$ in Figs. 3.2(a)-3.3(d). The horizontal axis indicates the traffic load ($M\epsilon$) with $M = 70$. In general, our analysis agrees well with simulations. However, we have found
that analytical and simulation results show some difference in the delay variance of BEB with small $L$ values ($\leq 3$), while other performance metrics show good agreements between analysis and simulation given any set of parameters. When a larger $L$ is used ($L \geq 4$) in the BEB algorithm, we can find good agreement between analytical and simulation results in the delay variance. Our conjecture is that, when $L$ becomes large, the process of our interest becomes memoryless, i.e., less dependent on lower stages with different window size.

Comparing the UB and EB algorithms, we observe that UB with $U = 20$ ($L = 5$) and BEB with $K = 5$ ($L = 0$) show almost identical throughput ($\tau$), the mean access delay ($\overline{d}$) and packet dropping probability ($P_d$) in Figs. 3.2(a)-3.2(b) and 3.2(d), respectively. Accordingly, it can be concluded that by tuning the parameters, $U$, $K$ and $L$ properly, the UB algorithm can give similar $\tau$, $\overline{d}$ and $P_d$ as the BEB algorithm. However, we can find two distinct differences between the UB and BEB algorithms. Firstly, $\overline{d}$ of BEB in the under-loaded region ($M_\epsilon < 0.45$) is much smaller than that of UB, because lower backoff stages are more frequently utilized in BEB. It is noticeable that in the under-loaded region, $\tau$ and $P_d$ of the two backoff algorithms do not show large differences. Secondly, the delay variance of BEB ($\overline{d}_v$) is much worse than that of UB in Fig. 3.2(c), which might be caused by the exponentially increasing nature of backoff intervals in the BEB algorithm. In both algorithms, we can observe that the peak of $\overline{d}_v$ occurs at the traffic load that gives the maximum $\tau$. We can also find that, when the system becomes overloaded, i.e., $p_c \to 1$, $\overline{d}$ and $\overline{d}_v$ of UB are respectively approximated as $\overline{d} = 0.5(L+1)U$ and $\overline{d}_v = (L+1)(U^2-1)/12$. For BEB, we have $\overline{d} = 0.5\sum_{i=0}^{K} W_i + 0.5LW_K$ and $\overline{d}_v = (1/12) \sum_{i=0}^{K} (W_i^2 - 1) + L(W_K^2 - 1)/12$. For the effect of window size, i.e., $U$ in the UB algorithm and $K$ in the BEB algorithm, one can find an important tradeoff between the throughput-packet dropping probability pair and the mean and variance of delay particularly in the overloaded region.
Figure 3.3: Performance of BEB algorithm with different $K$, $L$ and $M$. 
Chapter 3. Existing Backoff Algorithms for Single-Buffered Terminals

Figure 3.4: Access prioritization scheme of UB algorithm with differential $p$ and $U$. 

(a) Throughput. 

(b) Mean of retransmission delay. 

(c) Variance of retransmission delay. 

(d) Packet dropping probability.
(\(M \epsilon > 0.45\)). In this region, larger \(U\) or \(K\) increases \(\tau\) and reduces \(P_d\) at the expense of increasing the delay. Note that large \(U\) or \(K\) can stabilize the system for large \(M\). Therefore, it might be desirable that \(U\) and \(K\) would be adjusted according to traffic load in order to minimize delay performance.

For the effect of retry limit, larger \(L\) reduces \(P_d\) in UB and BEB algorithms, while other performance metrics get worse in the UB algorithm. In other words, UB with a smaller retry limit can provide better throughput and delay performance at the expense of increased packet dropping probability, in essence removing collisions by packet dropping.

In the UB algorithm, \(L\) represents the number of retransmissions as well as the number of backoff stages. In contrast, we can allow more retransmissions in BEB without increasing the backoff stage, i.e., \(K + L\). Thus, we additionally examine the effect of \(L\) in BEB algorithm in Figs. 3.3(a)-3.3(d). In these figures the line with black squares indicates the case with \(M = 100\), whereas other lines and marks indicate the cases with \(M = 70\). It is shown that increasing \(L\) given \(K\) in BEB improves \(\tau\) and reduces \(P_d\), while it increases the mean and variance of delay. The case with \(K = 8\) and \(L = 0\) improves \(\tau\) much more in the overloaded region than the case with \(K = 5\) and \(L = 5\), while the former shows huge variance of delay in Fig.3.3(c). Note that at the same traffic load, \(\tau\) and \(\bar{d}\) with larger population size \(M = 100\), depicted by the dash-dotted line with black squares, are worse than those with \(M = 70\).

3. Access prioritization in the UB algorithm: In Figs. 3.4(a)-3.4(d), we present the performance of the prioritized access scheme with two priority classes for \(P = 1\) as an example. We experiment with two prioritization schemes: One is to control the persistence probability of UB algorithm specified in [6], and the other one differentiates \(U\). In the first scheme, we set \(p^{[1]} = 1, p^{[2]} = 0.5\) and \(U^{[1]} = U^{[2]} = 20\), while we have \(p^{[1]} = p^{[2]} = 1, U^{[1]} = 20\) and \(U^{[2]} = 30\) for the case of controlling \(U\). In both cases, we have \(M^{[1]} = M^{[2]} = 35\) and
Chapter 3. Existing Backoff Algorithms for Single-Buffered Terminals

Figure 3.5: Access prioritization scheme of UB algorithm with differential $p$.

$L^{[1]} = L^{[2]} = 5$. The horizontal axis of each figure denotes the traffic load of the sum of each priority class, i.e., $M \epsilon = \sum_i M^{[i]} \epsilon^{[i]}$. First, we note that the performance differentiation does not seem to be outstanding in the under-loaded region. We can see that access priority can be effectively provided in the overloaded region for both schemes. In Figs. 3.4(a)-3.4(b) where $p^{[1]} = 1$ and $p^{[2]} = 0.5$ are used, the performance differentiation in $\tau$ is not as distinguishable as $\overline{d}$. To obtain a significant differentiated in $\tau$, we decrease $p^{[2]}$ to $2^{-4}$ while keeping $p^{[1]}$ fixed at 1. The results are shown in Figs. 3.5(a)-3.5(b). The smaller $p$ is used for the lower priority class, the larger performance differentiation is achieved in $\tau$ and $\overline{d}$. Note that $1/p$ corresponds to the initial delay at each retransmission opportunities. Similarly, controlling $U$ affects both throughput and delay performances. It should be noted that terminals with a high priority show a lower $P_d$, when prioritization is achieved by controlling $p$ in comparison with controlling $U$ as shown in Figs. 3.4(c)-3.4(d). In other words, controlling $p$ provides terminals of high priority with better (more desirable) performance in all aspects.
4. Access prioritization in the BEB algorithm: In Figs. 3.6(a)-3.6(d), we examine access prioritization in BEB with $p$ and $K$ for $P = 1$. As in Figs. 3.4(a)-3.4(d) for the UB algorithm, controlling $p$ provides terminals of higher priority with better performance in all aspects. However, controlling $K$ provides terminals of lower priority with lower $P_d$. Additionally, terminals with smaller $K$ in BEB get both higher throughput and lower delay, while they have a higher packet dropping probability. In other simulations (not shown here) in which $W_0$ is controlled, the same performance characteristics as controlling $K$ are observed. Therefore, controlling $p$ (initial access delay differentiation at each retransmission opportunity) gives higher priority terminals better performance in all aspects for both backoff algorithms.

5. Dynamic Window Assignment in UB algorithm: In Figs. 3.7(a)-3.7(b), we first test the DWA algorithm in static traffic condition in which $\epsilon$ does not change during the entire simulation. As an example, we set $P = 3$. It is observed that the algorithm achieves maximum throughput and minimum delay. In Figs. 3.7(c)-3.7(d), we set initial $\epsilon$ from 0.0031 to 0.095. Then we increase it by 1.8 times at the 10000-th slot. The results are obtained only from simulations and are time-averaged. Here, the horizontal axis indicates the average value of $\epsilon$, e.g. $\bar{\epsilon} = 0.0031(1 + 1.8)/2 = 0.0043$. In such a time-varying traffic condition, the algorithm achieves the maximum throughput and the minimum delay as well. Thus, under the perfect orthogonality assumption and low $P_d$, it is concluded that $1/U_t$ can approximate the optimum retransmission probability mentioned in [10][55].

6. Choice of UB and BEB: By controlling the window size of the UB algorithm, it can achieve a similar throughput performance as the BEB algorithm. Since BEB self-adapts to the traffic load, it is incorporated in the IEEE 802.3 Ethernet and IEEE 802.11 WLAN standards, which are intended for networks that have no central controller for traffic load (backlog size) estimation. While BEB works in a distributed manner, it suffers from a
Chapter 3. Existing Backoff Algorithms for Single-Buffered Terminals

Figure 3.6: Access prioritization scheme of BEB algorithm with differential $p$ and $K$. 
relatively high variance in access delay compared to UB. If a central controller is available and capable of estimating the backlog size, e.g., a BS in LTE, it may be preferable to use the UB algorithm due to its lower delay variance and equal throughput performance in comparison with the BEB algorithm.

3.4 Summary

In this chapter, we have examined UB and BEB algorithms with retry limit for the random access channels in LTE and IEEE 802.16 systems, respectively. We have analyzed their performance in terms of throughput, mean and variance of packet retransmission delay and packet dropping probability, and examined the stability properties of both algorithms. We have shown that stability mainly depends on the uniform window size and the maximum backoff stage and also that systems employing these algorithms are either stable or unstable, but do not show bistability behaviour. Most of our analytical results have shown good agreement with simulations except the delay variance of systems with small retry limits. By adjusting the parameters of each backoff algorithm properly, the two algorithms show virtually identical throughput, but different delay variances. We have also shown that controlling the persistence probability provides effective access priority control with respect to various performance metrics. Finally, we have proposed a dynamic window assignment algorithm, which is a simple extension of an existing Bayesian broadcasting algorithm. The dynamic window assignment algorithm outperforms the existing backoff algorithms in any traffic condition, as long as perfect orthogonality assumption among random access code holds.
Chapter 3. Existing Backoff Algorithms for Single-Buffered Terminals

Figure 3.7: Performance comparison of UB algorithm with/without DWA algorithm.
Chapter 4

Backoff algorithm with Power Ramping in LTE

Capture effects due to the differences of the signal levels, times of arrival, or other properties of colliding packets at the receiver have been widely studied over the past decades [58][59][60][61] by examining system’s throughput and delay performances and stability under various capture models. In order to exploit the throughput gain by such capture effects, some transmission power selection schemes in S-ALOHA systems has been also proposed in [62][63][64][65], in which the transmission power is selected based on a PMF among a set of transmission powers.

However, instead of selecting the transmission power optimally based on a PMF as in the literature, in a practical system such as LTE, terminals first estimate path loss from a downlink reference signal. Based on the assumption that path loss is symmetric over down- and uplink, a terminal transmits its packet by adding additional power to a target received power in order to compensate for the estimated path loss. If such a packet transmission is not successful, the terminal retransmits its packet by increasing the transmission power previously used to the next level, which is called the PR scheme [6][66]. The rationale of this scheme is to increase the transmission success probability independent of a terminal’s location by meeting a target received power adaptively, while SINR is kept above a given threshold. However, the optimal PR step size and the number of steps remain to be determined.
Chapter 4. Backoff Algorithm with Power Ramping in LTE

As previous work, in [63] Leung considered optimization problems of maximizing system throughput without exceeding some power constraint or minimizing the mean power consumption, while sustaining a minimum throughput under the perfect capture model, in which a packet with the strongest power level among packets simultaneously transmitted can capture the receiver, unless more than one packets of the strongest power level have been transmitted. In [64] LaMaire et al. considered optimization problems in order to maximize the capture probability with respect to the values of power levels and the probability distribution of selecting those values, when the number of terminals transmitting a packet under various capture models is known. Note that retransmission control is not considered in [63][64]. In [65] some power level selection algorithms with EB algorithm are considered under the saturated traffic condition, in which a finite number of terminals always have a packet to transmit. In [63][65], retransmitted packets from backlogged terminals are approximated by Poisson arrivals, regardless of finite or infinite user model. Especially, [67] examined the performance of various PR schemes with UB algorithm used in UMTS Terrestrial Radio Access (UTRA) by assuming an infinite user model. While the physical layer of UTRA is based on Wideband CDMA, in this thesis we are interested in LTE which physical layer is based on OFDMA. However, from the MAC layer’s viewpoint, the PR scheme with UB algorithm in UTRA [67] is almost identical to that of UMTS-LTE, except that the random access and the data transmission are separated over OFDMA sub-channels in LTE. In contrast with [67], the main contribution of this are summarized as follows:

1. We evaluate the performance of the PR scheme with UB algorithm by assuming statistically independent, identically distributed (i.i.d.) backoff processes in the terminals, even though in reality terminals’ backoff processes with the PR scheme are correlated. By obtaining the correlation coefficient of terminals’ backoff processes
from simulations, we show that the terminals’ backoff processes are so weakly (positively) correlated that the analysis based on i.i.d. assumption can agree well with simulations.

2. Under ideal channel conditions and assuming perfect path loss estimations, we show that the PR scheme is effective only when the SINR threshold is below $-3$ dB, compared to the system without the PR scheme.

3. In the PR scheme under consideration, a packet is dropped when a retransmission with the maximum transmit power is not successful. We define the capacity as the maximum acceptable number of terminals with a given packet arrival rate, which does not violate some specified packet dropping probability. We show that this capacity can be improved by the PR scheme with UB algorithm.

4. We incorporate the initial transmission power level selection based on a PMF in [63][64][65] into the PR scheme with UB algorithm. With respect to the average transmit power consumption and packet dropping probability, we show that the lowest-power-first transmission scheme is the most efficient compared to the other PMFs.

5. We propose a heuristic UB algorithm called a terminal-assisted UB algorithm (TAUBA) with PR, which helps a base station to estimate the backlog size based on the waiting-time information given in the packets successfully received. We show that TAUBA improves the system throughput close to its maximum and minimizes the packet dropping probability given any population size and packet arrival rate of each terminal, compared to static UB algorithms.

The rest of this chapter is organized as follows. In Section 4.1, we introduce the PR scheme in UMTS-LTE. The performance analysis is presented in Section 4.2. We discuss
some numerical examples in Section 4.3. Concluding remarks are given in Section 4.4.

4.1 System Model: Retransmission Backoff with Power Ramping

As we shall see, the random access of LTE consists of two parts. The first part is to transmit a RAP, and the second part is to transmit the terminal’s specific data on the uplink channel obtained from the RAP transmission. When an accessing terminal performs the first part of the random access, it first estimates the path loss to the serving cell from downlink reference signal received power. Then, the terminal randomly chooses one of the orthogonal RAPs and transmits it over a PRACH according to the PRACH configuration index. Let $p_t^{(i)}$ denote the $i$-th RAP’s transmission power for $1 \leq i \leq K$, which is determined by

$$p_t^{(i)} = \min(p_{\text{max}}, p_R + \xi_L + (i - 1)\Delta p + O),$$

(4.1)

in which $p_{\text{max}}$, $p_R$, $\xi_L$ and $\Delta p$ denote the maximum preamble transmission power, the target preamble received power, the estimated path loss and the PR step size, respectively. The parameter $O$ denotes an offset which is determined by the preamble format [68]. There are four different RAP formats which can be chosen based on the serving cell size. For analytical simplicity, we ignore this offset. Note that, if the path loss estimated by terminals is below a threshold, e.g., severe path loss, it is specified in [68] that the terminals add more power, e.g., 5 dB, to (4.1). In what follows, we assume that interference signals with severe path loss can be neglected, so long as the total interference power due to MAI is sufficiently large and dominant. Denote by $p_i$ the received power of a RAP at the serving cell, which is transmitted with power level $p_t^{(i)}$. Under the perfect path loss estimation
Chapter 4. Backoff Algorithm with Power Ramping in LTE

Assumption, we can write \( p_i \) for \( 1 \leq i \leq K \) as

\[
p_i = p_R + (i - 1)\Delta p,
\]

(4.2)

in which \( p_{\text{max}} \) has been assumed to be equal to \( p_K \). Then, we can write the probability that a RAP received at power level \( p_i \) can be successfully detected as

\[
Pr \left( \frac{p_i}{\sum_{j=1}^{K} n_j p_j + \gamma} > \beta \right) = I \left( \frac{p_i}{\sum_{j=1}^{K} n_j p_j + \gamma} > \beta \right),
\]

(4.3)

in which \( \beta \) and \( \gamma \) are the SINR threshold and other-cell interference, respectively. Additionally, \( n_j \) denotes the number of terminals transmitting their RAP at power level \( p^{(j)}_i \). The function \( I(x) \) is an indicator function, which is one if \( x \) is satisfied, or zero elsewhere. Note in (4.3) that we assume that multiple access interference is the dominant factor that affects RAP detection. The serving cell now broadcasts over downlink the RAR message. This RAR message indicates the identifiers of the RAPs that have been successfully received and indicates the uplink data channel, called physical uplink shared channel (PUSCH), in which the successful terminals should transmit their messages in the second part of the random access as described in the next paragraph. If more than one terminals have transmitted the same RAP, and if the transmission power of their RAP would meet (4.3), all these terminals will find their random access successful. When a terminal having transmitted its RAP with power level \( p^{(i)}_t \) does not find its RAP identifier in some RAR messages or within some specified time window, it regards its transmission as unsuccessful, i.e., implicit N-ACK. Then the terminal will perform the UB algorithm of window size \( U \) with PR scheme as follows. Note that the window size \( U \) is given in the RAR message. In the UB algorithm, the terminal randomly chooses a backoff counter between 0 and \( U - 1 \) and decrements this counter every slot. When this counter hits zero, the terminal transmits a
RAP with power level $p_{i}^{(i+1)}$. Upon a retransmission failure, the terminal repeats this UB algorithm jointly with the PR scheme. Upon a failure at the $K$-th retransmissions, the terminal drops the random access and reports it to a higher layer. If the retransmission is successful, the terminal performs the second part of the random access as described below.

In the second part of the random access, each terminal with a successful RAP transmission transmits a message with its unique identity in a PUSCH, by using H-ARQ. This will be given in Section 6.1.1. In this chapter we focus on the performance of the first part of the random access, i.e., the PR scheme with the UB algorithm.

### 4.2 Performance Evaluation

#### 4.2.1 Performance Metrics

Suppose there are $M$ terminals in a serving cell of LTE. We assume that one PRACH appears every slot, each corresponding to one RAP transmission time. A RAP is generated every slot in each terminal based on a Bernoulli trial with probability $\epsilon$, and each terminal can accommodate only one RAP. Moreover, in contrast with implicit NACK for RAP transmission as described above, for analytical simplicity we assume that the explicit feedback from a serving cell regarding RAP transmissions, i.e., success or failure, is immediately broadcasted to the terminals without any error. Further, we assume that each terminal performs the UB algorithm, immediately after it has a packet, i.e., DFT. In what follows, we will use the terms packet and RAP interchangeably. Define $\pi_E$ and $\pi_{i,j}$ as the stationary state probability that a terminal has no packet and that a terminal is in the $j$-th backoff counter for $0 \leq j \leq U - 1$ at the $i$-th backoff stage for $1 \leq i \leq K$. Further denote by $p_{s,i}$ the probability that a packet is successfully transmitted at the $i$-th backoff
stage. We then write the state balance equation for \( \pi_E \) as

\[
\pi_E = (1 - \epsilon)\pi_E + \sum_{i=1}^{K-1} p_{s,i} \pi_{i,0} + \pi_{K,0}. \tag{4.4}
\]

In order to generalize our to include the transmission power selection based on a PMF in [63][64][65], we denote by \( l_i \) for \( 1 \leq i \leq K \) the probability that a terminal chooses the \( i \)-th backoff stage with transmission power \( p^{(i)} \) and starts the UB algorithm (\( \sum_{i=1}^{K} l_i = 1 \)), when a packet is generated. At the first backoff stage, we have

\[
\pi_{1,j} = \frac{\epsilon l_1}{U} \pi_E + \pi_{1,j+1} \quad \text{for} \quad 0 \leq j \leq U - 2 \quad \text{and} \quad \pi_{1,U-1} = \frac{\epsilon l_1}{U} \pi_E. \tag{4.5}
\]

We can write \( \pi_{i,j} \) for \( 2 \leq i \leq K \) and \( 0 \leq j \leq U - 2 \) as

\[
\pi_{i,j} = \frac{\epsilon l_i}{U} \pi_E + \frac{1 - p_{s,i-1}}{U} \pi_{i-1,0} + \pi_{i,j+1}. \tag{4.6}
\]

For \( j = U - 1 \), we have

\[
\pi_{i,U-1} = \frac{\epsilon l_i}{U} \pi_E + \frac{1 - p_{s,i-1}}{U} \pi_{i-1,0}. \tag{4.7}
\]

By some manipulation on (4.5)-(6.4), we can write \( \pi_{i,j} \) as

\[
\pi_{1,j} = \frac{U - j}{U} \epsilon l_1 \pi_E \quad \text{for} \quad 0 \leq j \leq U - 1 \quad \text{and} \quad \pi_{1,U-1} = \frac{U - j}{U} (\epsilon l_1 \pi_E + p_{c,i-1} \pi_{i-1,0}) \quad \text{for} \quad 2 \leq i \leq K \quad \text{and} \quad 0 \leq j \leq U - 1 \tag{4.8}
\]

\[
\pi_{i,j} = \frac{U - j}{U} (\epsilon l_i \pi_E + p_{c,i-1} \pi_{i-1,0}) \quad \text{for} \quad 2 \leq i \leq K \quad \text{and} \quad 0 \leq j \leq U - 1 \tag{4.9}
\]
Chapter 4. Backoff Algorithm with Power Ramping in LTE

with $p_{c,i} = 1 - p_{s,i}$. We can further simplify (4.8) and (4.9) as

$$
\pi_{i,j} = \epsilon U - j \left( l_i + \sum_{k=1}^{i-1} l_k \prod_{j=k}^{i-1} p_{c,j} \right) \pi_E.
$$

(4.10)

By using the normalization condition, $\pi_E + \sum_{i=1}^{K} \sum_{j=0}^{U-1} \pi_{i,j} = 1$, we can get

$$
\pi_E = \left[ 1 + \epsilon \cdot 0.5(U + 1) \left( 1 + \sum_{i=2}^{K} \sum_{k=1}^{i-1} l_k \prod_{j=k}^{i-1} p_{c,j} \right) \right]^{-1}.
$$

(4.11)

By assuming that terminals’ backoff processes are i.i.d., we can write the packet transmission success probability at the $i$-th backoff stage as

$$
p_{s,i} = \sum_{n_0 + n_1 + \cdots + n_K = M-1} \Pr \left( \mathbf{p}_i > \beta \left( \sum_{j=1}^{K} n_j p_j + \gamma \right) \right) \frac{(M-1)!}{n_0! n_1! \cdots n_K!} \prod_{j=1}^{K} \pi_{n_j}^{n_j}.
$$

(4.12)

in which $\pi_{\cdot}$ denotes the probability that a terminal does not transmit a packet in a slot, i.e. $\pi_{\cdot} = 1 - \sum_{j=1}^{K} \pi_{j,0}$. Note that we have $p_{s,1} = p_{s,2} = \cdots = p_{s,K}$, if $\mathbf{p}_1 = \mathbf{p}_2 = \cdots = \mathbf{p}_K$.

We can now obtain the system throughput $\tau$ by

$$
\tau = \sum_{n_0 + n_1 + \cdots + n_K = M} \left( \sum_{i_1 = 0}^{n_1} \cdots \sum_{i_K}^{n_K} (i_1 + i_2 + \cdots + i_K) \prod_{j=1}^{K} \binom{n_j}{i_j} p_{s,j}^{n_j - i_j} \right) \\
\times \frac{M!}{n_0! n_1! \cdots n_K!} \prod_{j=1}^{K} \pi_{j,0}^{n_j} = M \sum_{i=1}^{K} p_{s,i} \pi_{i,0}.
$$

(4.13)

For the mean value of the packet retransmission delay (either successfully transmitted or dropped), we obtain the average backoff delay $\overline{d}$ by

$$
\overline{d} = 0.5(U + 1) \sum_{i=1}^{K-1} l_i \left( 1 + \sum_{h=i}^{K-1} \prod_{j=1}^{h} p_{c,j} \right) + 0.5(U + 1) l_K.
$$

(4.14)
in which we take it into consideration that \(0.5(U + 1)\) slots are taken on average every backoff stage. Denote by \(P_d\) the packet dropping probability that a terminal drops the packet unsuccessfully transmitted after the \(K\)-th retransmission attempt. We can obtain \(P_d\) by

\[
P_d = p_{c,K}\pi_{K,0}.
\]  

(4.15)

In addition, denote by \(\bar{p}\) the average power consumption by a terminal. We obtain \(\bar{p}\) by

\[
\bar{p} = \sum_{i=1}^{K} p_i \pi_{i,0} / \sum_{i=1}^{K} \pi_{i,0}.
\]  

(4.16)

The computational procedure is based on Brouwer’s fixed point theorem [69], which is as follows. We rewrite (4.11) and (4.12) respectively by

\[
\pi_{E}^{(\ell+1)} \triangleq F\left(p_{c,1}^{(\ell)}, p_{c,2}^{(\ell)}, \ldots, p_{c,K-1}^{(\ell)}\right) \quad \text{and} \quad p_{c,i}^{(\ell+1)} \triangleq G_i(\pi_{E}^{(\ell+1)}),
\]  

(4.17)

in which the superscript of \(\pi_{E}^{(\ell)}\) and \(p_{c,i}^{(\ell)}\) denotes the iteration index, \(\ell = 0, 1, 2, \ldots\). By setting \(p_{c,i}^{(0)} = 0.5\), we evaluate (4.17) iteratively until \(|\pi_{E}^{(\ell+1)} - \pi_{E}^{(\ell)}| < \varepsilon\). It is noticeable that, for each \(\ell\), \((M + K - 2)!/((K - 1)!(M - 1)!))\) evaluations on \(p_{s,i}\) are needed in (4.12). The uniqueness of the solution in (4.17) can be guaranteed, if \(|\partial F/\partial \pi_{E}| \leq \nu < 1\). We check this in numerical evaluations on (4.17).

We now evaluate the efficiency of the PR scheme compared to the system without PR. To this end, we define the throughput gain of the system with PR relative to the system without PR as \(\tau_p\) (packets/slot/dBm) = \((\hat{\tau} - \tilde{\tau})/(\bar{p} - p_R\)), in which \(\hat{\tau}\) and \(\tilde{\tau}\) denote \(\tau\) of the system with PR and without PR based on (4.13), respectively. Furthermore, it has been shown that a system with a finite number of retransmissions is always stable in
### Table 4.1: Throughput gain $\tau_p$ with $\beta$

<table>
<thead>
<tr>
<th>$U$</th>
<th>$M$</th>
<th>6</th>
<th>4</th>
<th>2</th>
<th>$-1$</th>
<th>$-1.5$</th>
<th>$-2$</th>
<th>$-2.5$</th>
<th>$-3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>20</td>
<td>0.0152 &amp; 0.0141 &amp; 0.0387 &amp; $-0.0440$ &amp; $-0.0301$ &amp; $-0.0320$ &amp; 0.0622 &amp; 0.0867</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>0.0116 &amp; 0.0136 &amp; 0.0348 &amp; $-0.0015$ &amp; 0.0173 &amp; 0.0230 &amp; 0.0693 &amp; 0.0764</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>0.0073 &amp; 0.0093 &amp; 0.0225 &amp; 0.0275 &amp; 0.0466 &amp; 0.0604 &amp; 0.0999 &amp; 0.1242</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>0.0040 &amp; 0.0046 &amp; 0.0139 &amp; 0.0311 &amp; 0.0484 &amp; 0.0633 &amp; 0.0974 &amp; 0.1239</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>0.0020 &amp; 0.0021 &amp; 0.0073 &amp; 0.0225 &amp; 0.0385 &amp; 0.0488 &amp; 0.0723 &amp; 0.0960</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>0.0006 &amp; 0.0012 &amp; 0.0037 &amp; 0.0154 &amp; 0.0249 &amp; 0.0353 &amp; 0.0501 &amp; 0.0707</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>60</td>
<td>0.0066 &amp; 0.0079 &amp; 0.0228 &amp; 0.0290 &amp; 0.0504 &amp; 0.0666 &amp; 0.1107 &amp; 0.1392</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>0.0053 &amp; 0.0061 &amp; 0.0170 &amp; 0.0313 &amp; 0.0487 &amp; 0.0647 &amp; 0.0971 &amp; 0.1285</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Chapter 3, due to $P_d$ in (4.15). Thus, we shall consider $\tau$ satisfying $P_d \leq 0.01$ as a maximum achievable throughput of this system, or $M$ satisfying $P_d \leq 0.01$ as the maximally allowable $M$, denoted by $M^*$.

### 4.2.2 Correlation Coefficient on Backoff Process

Finally, against the i.i.d assumption made in (4.12), we examine the correlation of $M$ terminals’ backoff process via simulation. To this end, let $b_i$ denote the backoff stage of the $i$-th terminal every slot during a simulation. We set $b_i = 0$ for the idle state and $b_i = j$ for the $j$-th backoff stage, i.e. $j = 1, 2, \ldots, K$, respectively. As a time series, $b_i$ is collected for the $i$-th terminal during a simulation. Then we define the average correlation coefficient of a terminal’s backoff process to other terminals as

\[
\rho = \frac{1}{M(M-1)} \sum_{i=1}^{M} \sum_{j=1, j \neq i}^{M} \frac{E[(b_i - E[b_i])(b_j - E[b_j])]}{\sqrt{Var(b_i)} \sqrt{Var(b_j)}}.
\]

in which $Var(x)$ denotes the variance of a random variable $x$, and we have $-1 \leq \rho \leq 1$. For $\rho > 0$, i.e., a positively correlated backoff process of $M$ terminals, each terminal’s backoff process has a tendency to move up to higher backoff stage together with other terminals, or move back to the idle state together. For $\rho < 0$, we expect that some terminals move
Algorithm 4.1 - Terminal-Assisted Uniform Backoff Algorithm

1: Initialize $U_0 = 20$ (as an example) and broadcast it. Do the following at $t = 1, 2, \ldots$.
2: if $N_t > 0$ and $P_t > 0$ then
3: $U_t = \text{nint}\left[\alpha U_{t-1} + (1 - \alpha) \frac{1}{N_t} \sum_{i=1}^{N_t} W_i\right]/s_\tau$.
4: else if $N_t = 0$ and $P_t > 0$ then
5: $U_t = \text{min}(\text{nint}[\alpha_u U_{t-1}], U_{\text{max}})$.
6: else
7: $U_t = \text{max}(\text{nint}[\alpha_d U_{t-1}], U_{\text{min}})$.
8: end if

up to a higher backoff stage, while other terminals move back to the idle state.

4.2.3 Terminal-Assisted UB Algorithm (TAUBA)

We now introduce TAUBA for the system with PR. To begin with, the symbols used in this algorithm are summarized below:

- $W_i$: Waiting-time counter of the $i$-th terminal.
- $U_t$: Window size of UB algorithm broadcasted by eNodeB at time $t$.
- $N_t$: Number of packets successfully received at eNodeB at time $t$.
- $P_t$: Total received powers of packets measured by eNodeB at time $t$.
- $s_\tau$: Scaling factor of window size and $s_\tau \geq 1$.
- $U_{\text{max}}$ (or $U_{\text{min}}$): The maximum (or minimum) of window size.
- $\alpha$: Smoothing factor for $U_t$ and $0 < \alpha < 1$.
- $\alpha_u$: Increasing factor of $U_t$ and $\alpha_u > 1$.
- $\alpha_d$: Decreasing factor of $U_t$ and $0 < \alpha_d < 1$. 
Chapter 4. Backoff Algorithm with Power Ramping in LTE

In Algorithm TAUBA, the notation, \( \text{nint}[x] \), indicates the nearest integer function, which chooses the integer close to \( x \). Every slot, eNodeB broadcasts \( \mathcal{U}_t \) based on Algorithm TAUBA. Each terminal sets \( \mathcal{W}_i = 0 \) for \( 1 \leq i \leq M \), when it has no packet to send, or when its random access is terminated, either success or dropping a packet. Upon a packet arrival, the terminal conducts UB algorithm with \( \mathcal{U}_t \) (randomly chooses a backoff counter between 0 and \( \mathcal{U}_t \)) and PR, while keeping on increasing its waiting-time counter by one at each slot. When the terminal (re)transmits its packet, it appends the waiting-time counter, \( \mathcal{W}_i \), to the packet. In practice, \( \mathcal{W}_i \) can be appended to either RAP or Msg3. If eNodeB would receive the packet successfully, it can estimate the current backlog size by assuming that \( \mathcal{W}_i \) from the packet successfully received would be proportional to the current backlog size. Note that more than one packet could be successfully received based on \( \beta \) in (4.3). Thus, if \( \mathcal{N}_t > 1 \), eNodeB averages \( \mathcal{W}_i \) over \( \mathcal{N}_t \). Additionally, we can use smaller window size than the actual backlog size for the system with \( \mathcal{N}_t > 1 \) in order to allow more than one retransmissions. The 2nd and 3rd lines of Algorithm 4.1 reflect this.

If the packet (re)transmission is not successful, say time \( t + \Delta t \), the terminal conducts UB algorithm with \( \mathcal{U}_{t+\Delta t} \) in conjunction with PR. This is repeatedly done, until the terminal would transmit the packet successfully or drops the packet after using \( p_t^{(K)} \). Therefore, \( \mathcal{W}_i \) is a cumulative counter of summing up all the backoff counters used. Furthermore, every slot eNodeB measures the power of received packets so that it can detect heavily congested random channel, when \( \mathcal{N}_t = 0 \) and \( \mathcal{P}_t > 0 \). In this case, since eNodeB cannot have the information of \( \mathcal{W}_i \), it increases \( \mathcal{U}_t \) by multiplying \( \alpha_u \), which is expressed in the 4th and 5th lines. If \( \mathcal{N}_t = \mathcal{P}_t = 0 \), then eNodeB reduces its window size by multiplying \( \alpha_d \), under the assumption that the random access channel becomes gradually idle. We explain in the next section how to select these parameters, \( \alpha \), \( \alpha_u \), \( \alpha_d \), \( s_r \) and so on.
Chapter 4. Backoff Algorithm with Power Ramping in LTE

Figure 4.1: The effect of power ramping and $U$ on the system performance.
4.3 Numerical Studies

We build a simulation program in MATLAB. Each simulation is carried out for 20000 slots and time-averaged. We set $p_R = -90$ dBm, $\Delta p = 2$ dB and $\epsilon = 0.095$, unless otherwise stated.

1. Effect of SINR threshold: In Table 4.1, we examine $\tau_p$ against $\beta$ with $K = 5$. The results in Table 4.1 are obtained by simulations. The validation of this simulation will be shown in Figs. 4.1(a)-4.2(d). In Table 4.1, some negative values imply that the system with PR is not better than that without PR. When $M$ is increased, $\tau_p$ gradually dies out. We can improve $\tau_p$ by introducing larger $U$. We can see that PR is quite effective only when $\beta \leq -2.5$ dB. In what follows, we set $\beta = -3$ dB in order to characterize the PR effect.

2. Effect of PR and $U$: In Figs. 4.1(a)-4.1(d), we examine the effect of PR and $U$ on the system performances by comparing it with the system without PR. We set $l_1 = 1$ and $l_i = 0$ for $2 \leq i \leq K$ and $K = 4$. While $\tau$, $M^*$ and $\overline{d}$ can be improved by PR as shown in Figs. 4.1(a)-4.1(c), more power consumption is needed as shown in Fig. 4.1(d), e.g. about 3.3 dBm at $M^* = 34$ in Fig. 4.1(d). Note that the terminals without PR always consume $-90$ dBm. When we increase $U$ in the system with PR, we can see that larger $U$ improves $\overline{P}$ and $P_d$ significantly without significant increase of $\overline{d}$. Moreover, $U$ should be dynamically controlled, since larger $U$ reduces $\tau$ and increases $\overline{d}$ for small $M$ in Figs. 4.1(a) and 4.1(b).

3. Initial transmission power selection based on a PMF: In Figs. 4.2(a)-4.2(d), we consider a random transmission power selection scheme based on 1) Exponential decreasing selection (ES): $l_i = 2^{-(i-1)}l_1$ for $2 \leq i \leq K$ and 2) Uniform selection (US): $l_i = 1/K$ for $1 \leq i \leq K$ for $K = 4$ and $U = 15$. It is shown that $\tau$ seems insensitive to the PMF of selecting initial transmission power in Fig. 4.2(a). The higher the probability of selecting
Chapter 4. Backoff Algorithm with Power Ramping in LTE

Figure 4.2: The effect of the initial transmission power selection on the system performance.
4. **Correlation**: Fig. 4.3 depicts $\rho$ for $K = 5$ when PR is used. For the case without PR, we use $\epsilon = 0.125$ and $U = 10$ and set $K = 100$ as the maximum number of retransmissions. It is found that each terminal’s backoff process is positively correlated, i.e., $\rho > 0$. When PR is used, $\rho$ dies out as $M$ increases, which is caused by packet dropping after failure of the $K$-th retransmission. Compared to $U = 10$, larger $U = 20$ makes $\rho$ lower for $\epsilon = 0.095$, which means that larger $U$ reduces collisions. When a large number of retransmissions are allowed ($K = 100$), $\rho$ becomes relatively higher than the cases with $K = 5$. However, we can see that $\rho$ does not exceed 0.135, which indicates that the backoff processes are very weakly correlated. Fig. 4.3 explains why our analysis based on i.i.d assumption agrees well with simulations.

5. **TAUBA**: In this example, we use $K = 5$, $\beta = -3$ (dB), $\epsilon = 0.075$ and $U_0 = 10$. For
Chapter 4. Backoff Algorithm with Power Ramping in LTE

Figure 4.4: Performance of TAUBA against some static UB algorithms.
the parameters of TAUBA, we set $\alpha = 0.95$ and $\alpha_d = 0.95$ by assuming that the backlog size of this system slowly varies over time. When eNodeB observes $N_t = 0$ and $P_t > 0$ in some consecutive slots and use $\alpha_u \geq 2$, some terminals just joining the ongoing contentions unexpectedly could have very large window size compared to actual backlog size. On the other hand, if we use $\alpha_u$ close to one, the ongoing (congested) contentions may not be easily resolved. To avoid this, we use $\alpha_u = 1.25$. As we have seen that the maximum throughput of this system is close to 1.05 in Fig. 4.1(a), we set $s_\tau = 1.05$. Finally, we set $U_{\min} = 2$ and $U_{\max} = 256$, which implies that 8 bits are enough to describe $W_i$ and $U_t$.

For convenience, in Fig. 4.4(a) we call the range of population sizes before $\tau$ reaches the maximum, e.g., $10 \leq M \leq 40$ for $U = 20$, the lightly loaded region; otherwise the heavily loaded region. Compared to static UB algorithms that use fixed values of $U = 10$, 20 or 30, TAUBA maximizes $\tau$ very close to the maximum (1.05) in the heavily loaded region, while $\tau$ of static UB algorithms decreases and goes to zero as $M$ increases. Over the lightly loaded region, TAUBA improves $\tau$ compared to the static UB algorithms. For $\bar{d}$ and $\bar{d}_v$, TAUBA minimizes them in the lightly loaded region, while $\bar{d}$ and $\bar{d}_v$ of TAUBA approach to the maxima of $\bar{d}_s$ and $\bar{d}_v$s for the static UB algorithms in the heavily loaded region in Figs. 4.4(b) and 4.4(c). Note that these static UB algorithms with $U = 10$, 20 and 30, respectively, yield $P_d = 0.023$, 0.00927 and 0.0023 at $M = 80$ with delay performances shown in Figs. 4.4(b) and 4.4(c), while TAUBA achieves $P_d < 10^{-5}$ for $10 \leq M \leq 80$.

In Fig. 4.4(d), we can see that TAUBA minimizes the mean power consumption in the heavily loaded region, while it consumes more power in the lightly loaded region. The use of different $\epsilon$ values does not change the performance behaviours of TAUBA as shown in Fig. 4.4(a)-4.4(d). In conclusion, based on very low $P_d$ and the minimal use of $\bar{p}$, it is conjectured that the window size of the first or possibly the second backoff stage of TAUBA is optimized to maximize $\tau$ particularly in the heavily loaded region.
4.4 Summary

In this chapter we have examined the PR scheme with UB algorithm in LTE in conjunction with initial power transmission selection based on some PMFs, by assuming that uplink pathloss estimation is perfect. We have evaluated the performance of this system based on i.i.d assumption on $M$ terminals’ backoff processes and compared it to simulation. The good agreements between analysis and simulation have been explained by weak positive correlation among $M$ terminals’ backoff processes due to the retransmission limit built in the PR scheme. It is also shown that the PR scheme is effective, when the SINR threshold is less than $-2.5$ dB. In the PR scheme, throughput and delay are improved, while more terminals without violating $P_d < 0.01$ can be allowed at the cost of a higher average transmission power. In the probabilistic initial transmission power selection schemes given, $\tau$ is insensitive to the PMFs of selecting initial transmission power. Further, increasing the probability of selecting the highest power can increase $P_d$, while reducing $\bar{d}$. Therefore, the LPFT is the most efficient one with respect to $P_d$ and $\bar{d}$. In the presence of the PR scheme, it is also needed to control $U$ dynamically in order to maximize $\tau$ according to the number of backlogged terminals. To this end, we have proposed TAUBA, which estimates the backlog size based on terminals waiting-time counters and takes throughput gain by PR into account. By simulations, we have shown that TAUBA improves the system throughput close to its maximum and minimizes mean access delay and packet dropping.
Chapter 5

Queueing Performance of Multichannel ALOHA Systems

Various Internet services, such as Web-browsing with rich and multimedia-embedded contents, networked gaming, interactive multimedia and telemedicine, have been gradually deployed over various wireless networks, e.g., IEEE 802.11 WLANs, UMTS, LTE and IEEE 802.16 WiMAX systems. Moreover, recent innovative advances of smart-phones and tablet personal computers have accelerated the emergence of new services, e.g., location-based service. Such services are mostly carried by Web-browsing, which is emerging as one of the most widely used data applications over wireless and wireline networks [70].

It is well known that the traffic characteristics of Web-browsing depend on user behaviour, type of Web-browsers, interaction of the Transmission Control Protocol (TCP) senders and receivers, and versions of the Hyper-Text Transfer Protocol (HTTP), and exhibit long-range dependency [70][71][72]. These might make it difficult to develop a unifying source traffic model for evaluating the performance of wireless access networks. In this chapter, we consider uplink traffic in a wireless access system due to Web-browsing, generated by HTTP over TCP from the client in the mobile device to the Web-server in the fixed network. In this scenario, the uplink traffic generated at mobile terminals mostly consists of main/inline object requests and TCP Acknowledgements (ACKs) as shown in Fig. 5.1. It is shown that TCP SYN and Web-request are generated at the moments that the user initiates Web-browsing or requests an object, while most of TCP ACKs are
generated when main/inline objects are being downloaded. Once downloading of objects is completed, the user reads those objects and no uplink traffic is generated during this time, until the user initiates a new Web-request. This observation indicates that the Poisson or independent source model might not be suitable for modelling uplink traffic of Web browsing [71][73] especially in performance studies of random access protocols of wireless systems.

Recently IEEE 802.11 WLANs have been widely deployed in hotspots to provide public Internet access. However, there is a growing demand for not only more bandwidth but anywhere always-on connectivity to the Internet. IEEE 802.16 systems and LTE are candidate solutions to meet such a demand. Particularly, the uplink access procedure in both IEEE 802.16 systems and LTE employs demand-assignment multiple access, in which a terminal transmits an orthogonal random access preamble in order to get an uplink channel where uplink bandwidth requests could be sent later. After the terminal receives a grant for an uplink bandwidth request, it could transmit actual data packets. However, due to such a 4-way handshake, TCP ACK transmissions are expected to suffer from long delays [74]. In order to improve the bandwidth request mechanism, IEEE 802.16 system employs a polling mechanism to solicit for uplink bandwidth request [30], while LTE adopts a dedicated scheduling request channel in which time-synchronized terminals are periodically scheduled for uplink bandwidth request transmissions [6][75][76]. However, uplink packets from Web-browsing, e.g., TCP SYN, Web-request and TCP ACKs, have very short lengths (e.g., small IP packet of 40 bytes), which makes it inefficient to employ the demand assignment multiple access procedure as described above to request bandwidth for each packet. Instead, it may be desirable to send these packets directly over the RACHs without adding to the traffic load of these channels.

In this chapter, we consider uplink packet transmissions for Web-browsing (client to
server traffic in Fig. 5.1) over the RACHs of LTE. Our investigation focuses on the contention resolution mechanism applied to the orthogonal random access preambles.

Thus, our work will help the performance estimation of delivering delay-tolerant Web-traffic by random access in LTE. To capture the behaviours of the RACHs of LTE [6], which employ CDMA-OFDMA, we model the RACHs as a multichannel S-ALOHA system under the perfect orthogonality condition among random access preambles. We model the uplink traffic from Web-browsing by an \( N \)-state MMBP whose parameters could be matched to measured traffic [77][78]. While the uplink traffic statistics might depend on downlink traffic and user behaviour, this study focuses on the uplink MAC only, and the MMBP traffic model allows the effects of correlations between uplink packet arrivals on the MAC performance to be investigated. For more realistic system modelling, we also consider the collision resolution algorithm standardized in LTE [6], i.e., the UB algorithm with retry limit, operating in multi-buffered terminals, which have a finite queue size to buffer a number of uplink packets.

There is a long research history of queuing studies on S-ALOHA systems with multi-
buffered terminals, even in conjunction with information-theoretical perspective \cite{79,80} (and references therein). Since the evolution of terminals’ queuing processes over time is coupled with each other, analytical models, even approximate ones, which have been developed so far have mostly suffered from the curse of dimensionality. As good approximations, there are some analytical queuing models for multi-buffered terminals in S-ALOHA systems in \cite{81,82,83}. However, none of these models considers correlated traffic arrivals or window-based backoff with retry limit. On the other hand, queuing models for multi-buffered terminals have been developed \cite{50,84,85} for IEEE 802.11 WLANs employing CSMA with collision avoidance (CSMA/CA) with binary exponential backoff. While previous work has mainly focused on constructing and validating queuing models, the impacts of retry limit or correlated traffic arrivals on the system performance have not been explored. Internet traffic like HTTP has been taken into account in the performance studies of multi-buffered terminals in IEEE 802.11 WLANs \cite{86,87,88}. However, these queuing studies focus on the interaction between TCP senders and receivers, and assume infinite queue size and unlimited retransmissions. In contrast, our work takes into account of finite queue size, retry limit, and correlated traffic arrivals.

In addition to these queuing studies \cite{50,81,82,83,84,85}, ALOHA systems with source traffic having long-range dependency have been examined in \cite{89,90,91}. These studies commonly concluded that ALOHA systems with Poisson traffic arrivals show worst system performance than those with traffic having long-range dependency, which is contradictory to earlier results found in conventional queuing systems \cite{92}. Note that the work in \cite{89,90,91} is based only on simulations and does not consider the dynamics of window-based backoff algorithms. In contrast to previous work, the main contributions of our work are summarized as follows:

1. We model the queuing process of multichannel S-ALOHA access with UB and retry
limit, and evaluate system performance in terms of system throughput, queue length and delay, probabilities packet dropping due to retry limit and packet blocking due to finite queue size, and mean and variance of packet retransmission time with respect to the degree of arrival correlation, number of channels, retry limit and uniform window size. Results are applicable to Web-browsing with uplink packets transmitted over the RACHs.

2. In contention resolution using UB in single-buffered terminals, it is known [15] that performance is optimized by dynamically changing the window size according to the number of backlogged terminals. We present simulation results to show that this optimization could not be applied to the system with multi-buffered terminals and correlated arrivals, in which case some fixed window size shows better performance.

## 5.1 System Model

### 5.1.1 Uplink Traffic Source Model

**An N-state MMBP Model**

As previously mentioned, Web-browsing generates TCP-SYNs, Web-requests and TCP ACKs over the wireless uplink, as shown in Fig. 5.1. Note that the number of TCP ACKs depends on the amount of main/inline objects downloaded over downlink in conjunction with maximum transmission unit and the delayed ACK option. In addition, multiple TCP connections can be established during one HTTP session [70][71], or multiple Web-browsing sessions can be initiated by a user. Furthermore, parsing time could be considered between main and the first inline object downloading, which affects inter-arrival times between TCP ACKs generated during this ON period. On the other hand, no uplink traffic is generated when the user is reading the downloaded objects during the OFF period, which depends
on user-behaviour. Due to these conditions, it is not easy to draw meaningful statistical data from measurements on uplink traffic to characterize it. Without loss of generality, we assume that an N-state MMBP can capture the ON-OFF alternating behaviour of Web browsing shown in Fig. 5.1. Note that the parameters of a two-state MMBP have been matched to measured traffic data in [77][78] and references therein.

Denote the underlying state transition matrix of an MMBP at every slot boundary by $Q$. The element of $Q$ in the $i$-th row and the $j$-th column, denoted by $q_{ij}$ for $0 \leq i, j \leq N-1$, indicates the state transition probability from the $i$-th state to the $j$-th state. In each state, say the $i$-th state, a packet is generated by a Bernoulli trial with probability $\epsilon_i$. Let row vector $\tilde{\beta}$ denote the stationary state probability of this MMBP. It satisfies $\tilde{\beta} = \tilde{\beta} \cdot Q$ and $\tilde{\beta} \cdot \bar{e}^T = 1$, in which $\bar{e}$ is a unit row vector, and $T$ denotes a vector or matrix transpose. Throughout this chapter, the length of $\bar{e}$ conforms to the corresponding matrix or vector, which makes a matrix or vector product possible. Let $diag[\epsilon_i]$ denote a diagonal matrix with elements $\epsilon_i$ for $0 \leq i \leq N - 1$. Then, the mean and variance of the number of packets generated by this MMBP can be obtained by $\bar{\nu} = \bar{e} \cdot diag[\epsilon_i] \cdot \tilde{\beta}^T$ and $\sigma^2 = \bar{\nu}(1 - \bar{\nu})$, respectively.

For characterization of the second order statistics of an MMBP, we consider the auto-correlation function of the number of packet arrivals. Denote by $a^{(m)} \in [0,1]$ the number of packets generated at the $m$-th slot and by $s^{(m)}$ the source state at that slot, respectively. Then we can write the normalized auto-correlation function of the number of packet arrivals of MMBP as

$$R_N(n) = \frac{1}{\sigma^2} \mathbb{E} [(a^{(m+n)} - \bar{\nu})(a^{(m)} - \bar{\nu})] = \frac{1}{\sigma^2} \left( \mathbb{E} [(a^{(m+n)}a^{(m)}) - \bar{\nu}^2] \right).$$

(5.1)

where $\mathbb{E}[x]$ denotes the expectation of random variable $x$. We can write $\mathbb{E} [a^{(m+n)}a^{(m)}]$ for
Chapter 5. Queueing Performance of Multichannel ALOHA Systems

\( n \geq 1 \) by

\[
R(n) = \mathbb{E}[a^{(m+n)}a^{(m)}] = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{k_1=0}^{1} \sum_{k_2=0}^{1} k_1 k_2 \Pr[a^{(m+n)} = k_2 | s^{(m+n)} = j] \times \Pr[a^{(m)} = k_1 | s^{(m)} = i] \Pr[s^{(m+n)} = j | s^{(m)} = i] \Pr[s^{(m)} = i]
\]

\[
= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \epsilon_j \epsilon_i \Pr[s^{(m+n)} = j | s^{(m)} = i] \Pr[s^{(m)} = i]
\]

\[
= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \epsilon_i \epsilon_j \bar{q}_{ij}^n \beta_i = \bar{\beta} \cdot \text{diag}[\epsilon_i] \cdot Q^n \cdot \text{diag}[\epsilon_i] \cdot \vec{e}^T,
\] (5.2)

in which \( \bar{q}_{ij}^n \) is the \( i \)-th row and the \( j \)-th column element of the \( n \)-step state transition probability matrix \( Q^n \). For \( n = 0 \), we can get

\[
R(0) = \sum_{i=0}^{N-1} \sum_{k=0}^{1} k^2 \Pr[a^{(m)} = k | s^{(m)} = i] \Pr[s^{(m)} = i] = \sum_{k=0}^{N-1} \epsilon_k \beta_k,
\] (5.3)

which leads to \( R_N(0) = 1 \).

For simplicity of illustration, consider a two-state MMBP, with \( \beta_0 = (1 - q_{11})/(2 - q_{00} - q_{11}) \) and \( \beta_1 = 1 - \beta_0 \). In addition, we obtain the probability that a source is in the \( i \)-th state for \( n \) slots, \( \zeta_i(n) \), and its mean, \( \bar{t}_i \), for \( i = 0, 1 \) respectively by

\[
\zeta_i(n) = p_{ii}^{n-1}(1 - p_{ii}) \quad \text{and} \quad \bar{t}_i = 1/(1 - p_{ii})
\] (5.4)

Particularly for an ON-OFF source \( (\epsilon_0 = 0 \text{ and } \epsilon_1 \neq 0) \) as a special case, we can respectively characterize \( R_N(n) \) and \( R(n) \) for \( n \geq 1 \) by

\[
R_N(n) = c \cdot \omega^n \quad \text{and} \quad R(n) = \bar{\epsilon}^2 + \bar{\epsilon}(\epsilon_1 - \bar{\epsilon})\omega^n
\] (5.5)

where \( c = \bar{\epsilon}(\epsilon_1 - \bar{\epsilon})/\sigma^2 \) and \( \omega = p_{00} + p_{11} - 1 \). If \( \epsilon_1 = 1 \), i.e., a packet is always generated
every slot during ON-state, we have simply $R_N(n) = \omega^n$. Thus, for this two-state ON-OFF source, $\omega$ plays a key factor in characterizing the correlation on packet arrivals.

**Long-Range Dependence (or heavy-tailed distribution) in TCP traffic model**

The long range dependence of a stochastic (or traffic) process can be characterized by its autocorrelation function ($R(n)$) as \[93\]

$$R(n) = \varsigma \cdot n^{-\kappa} \quad \text{for} \quad n \to \infty \quad \text{and} \quad 0 < \kappa < 1$$ \hspace{1cm} (5.6)

where $\varsigma > 0$ is a constant. Compared to the short range dependency which decays exponentially as shown in (5.2), this decays hyperbolically. In [94] Andersen and Nielsen show how to construct a traffic with long-range dependence by superposing four 2-state Markov Modulated Poisson Processes (MMPPs) [95]. In the following queueing model in Section 5.2.2, MMPP traffic model can be easily incorporated. Due to MMBP traffic model, our queueing process is a block-structured Quasi-Birth-Death (QBD) process. However, our queueing process with MMPP traffic model becomes a blocked-structured $M/G/1$ type queue (or $MMPP/G/1$ queue), to which the block Gauss-Seidel iteration described later can be also applied with some modifications in order to obtain the stationary state probability, or the algorithm in [96] can be used.

### 5.1.2 Access Protocol and Backoff Algorithm

This chapter also considers the random access protocol and the UB algorithm given in Section 1.1 and Section 3.2.1. The packet transmission success probability denoted by $p_s$ is also given in Section 3.2.1.
5.2 Queueing Performance Evaluation

5.2.1 Approximation on Correlated Multi-queue

Let $q_i$ denote the queue size of the terminal $i$ among $M$ terminals in steady state. Assume that the arrival process and the backoff process for all terminals are identical. Suppose the joint probability that the queue size of $k$ terminals selected from $M$ terminals is greater than 0. Based on i.i.d. assumption on terminals’ queueing processes, this can be approximately written by $M!(k!(M-k)!)(\Pr[q_i > 0])^k(1 - \Pr[q_i > 0])^{M-k}$, in which any terminal can be the $i$-th terminal. This is always less than or equal to the exact joint probability. In what follows, our analysis is based on the i.i.d assumption that leads to the RHS. However, in Section 5.3 we examine the correlation of $M$ terminals’ queueing processes as follows.

Denote by $q_i^{(t)}$ a time series of queue length of the $i$-th terminal collected every slot during a simulation. Then, we measure the correlation coefficient $\varrho$ over $M$ terminals’ queueing process as

$$\varrho = \frac{1}{M(M-1)} \sum_{i=1}^{M} \sum_{j=1, j \neq i}^{M} \frac{E\left[(q_i^{(t)} - E[q_i^{(t)}])(q_j^{(t)} - E[q_j^{(t)}])\right]}{\sqrt{\text{Var}(q_i^{(t)})} \sqrt{\text{Var}(q_j^{(t)})}}. \quad (5.7)$$

For $\varrho > 0$ each terminal’s queue would increase or decrease together. For $\varrho < 0$ some of terminal’s queue would decrease, while the rest of them would increase. If our analysis might agree well with simulation, it can be said that the correlation of the queueing process is so weakly correlated.

In addition, we have discussed the stability condition of multi-buffered terminals in an S-ALOHA system in Chapter 1. Due to finite buffer size, this system always satisfies (1.6). We also have seen the bistability of single-buffered terminals in an S-ALOHA system. If the system with multi-buffered terminals would have bistability, the probability that $k$ out
of $M$ terminals’ queue are not empty might have a bimodal distribution. As long as the analysis based on i.i.d. assumption agrees well with simulations, we can conjecture that this probability can be approximated by $M!/(k!(M-k)!)(Pr[q_i > 0])^k(1−Pr[q_i > 0])^{M−k}$, which means no bistability in the S-ALOHA system with multi-buffered terminals.

5.2.2 Multi-buffered Terminals

The state space of the queue in a given terminal can be described by $\{(k, j, i, u) : 0 \leq k \leq K, 0 \leq j \leq J−1, 0 \leq i \leq N−1, 0 \leq u \leq U−1\}$, in which $k$, $j$, $i$ and $u$ respectively denote the number of packets in the queue, the $j$-th retransmission attempt, the $i$-th source traffic state and the $u$-th backoff counter in the terminal. Denote by $\bar{\varphi}_k = [\varphi_{k,j,i,u}]$ for $1 \leq k \leq K$, $0 \leq j \leq J−1$, $0 \leq i \leq N−1$ and $0 \leq u \leq U−1$ the row vector of stationary state probability that at an arbitrary slot boundary a terminal has $k$ packets in its queue, its backoff counter is at the $u$-th state, it is making the $j$-th retransmission attempt, and its traffic source is in state $i$. In each $\bar{\varphi}_k$, $\varphi_{k,j,i,u}$ is lexicographically ordered. Additionally, row vector $\bar{\varphi}_0 = [\varphi_{0,i}]$ denotes the stationary probability that a terminal has no packet in its queue. The subscript $i$ of $\varphi_{0,i}$ denotes the $i$-th source state for $0 \leq i \leq N−1$. For $K \geq 2$, we can write a set of state balance equations for $\bar{\varphi}_k$ for $0 \leq k \leq K$ as follows:

$$\bar{\varphi}_0 = \bar{\varphi}_0Q_b + \bar{\varphi}_1C \quad \text{and} \quad \bar{\varphi}_1 = \bar{\varphi}_0F + \bar{\varphi}_1B + \bar{\varphi}_2D,$$

$$(5.8)$$

$$\bar{\varphi}_k = \bar{\varphi}_{k−1}A + \bar{\varphi}_kB + \bar{\varphi}_{k+1}D \quad \text{for} \quad 2 \leq k \leq K−1,$$

$$(5.9)$$

$$\bar{\varphi}_K = \bar{\varphi}_{K−1}A + \bar{\varphi}_KB_a \quad \text{and} \quad \bar{\varphi}_0\mathbf{e}^T + \sum_{k=1}^{K} \bar{\varphi}_k\mathbf{e}^T = 1.$$

$$(5.10)$$

For $K = 1$, we have

$$\bar{\varphi}_0 = \bar{\varphi}_0Q_b + \bar{\varphi}_1C \quad \text{and} \quad \bar{\varphi}_1 = \bar{\varphi}_0F + \bar{\varphi}_1B_a$$

$$(5.11)$$
where the matrix $Q_b$ is expressed by

$$Q_b = (I - \text{diag}[\epsilon_i]) \cdot Q. \quad (5.12)$$

In the above, $I$ denotes an identity matrix whose size conforms to that of $\text{diag}[\epsilon_i]$. The matrix $Q_b$ indicates the state transitions of a traffic source without a packet arrival. We can write an $NJU \times N$ matrix $C$ in (5.8) as

$$C = \left[ \begin{array}{c c c c} C^T_0 & C^T_0 & \cdots & C^T_0 \\ \vdots & \vdots & \ddots & \vdots \\ C^T_1 & \end{array} \right] \cdot \left( J - 1 \right)$$

where $C_0$ and $C_1$ are respectively given by

$$C_0 = Q_b \otimes \vec{c}_0^T \quad \text{and} \quad C_1 = Q_b \otimes \vec{c}_1^T \quad (5.14)$$

and $\otimes$ denotes the Kronecker product of two matrices or of a matrix and a vector. The row vectors $\vec{c}_0$ and $\vec{c}_1$ of $U$ elements can be expressed as

$$\vec{c}_0 = \begin{bmatrix} p_s & 0 & 0 & \cdots & 0 \end{bmatrix} \quad \text{and} \quad \vec{c}_1 = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \end{bmatrix}. \quad (5.15)$$

For $J = 1$, we have $C = C_1$. In (5.14), $C_0$ indicates the state transition of a traffic source with a successful packet transmission, while successful and unsuccessful packet transmissions are included in $C_1$. The $N \times NJU$ matrix $F$ in (5.8) can be written as

$$F = \left[ \begin{array}{c c c c} Q_a \otimes \vec{a} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \end{array} \right] \quad (5.16)$$

where $0$ denotes a zero matrix of $N \times NU$ size. Throughout this chapter, $0$ denotes a block matrix of all zeros, whose size is appropriately chosen. The matrix $Q_a$ and the row vector
\( \vec{a} \) can be expressed by

\[
Q_a = \text{diag}[\epsilon_i] \cdot Q \quad \text{and} \quad \vec{a} = [1/U \ 1/U \ \cdots \ 1/U]
\]

where \( Q_a \) denote the state transitions of a traffic source with a packet arrival. Before describing the matrices \( B, B_a, D \) and \( A \), we define \( \zeta_{a,0}, \zeta_{a,1}, \zeta_{a,2}, \zeta_{b,0}, \zeta_{b,1} \) and \( \zeta_{b,2} \) as

\[
\zeta_{a,0} = Q_a \otimes \theta_0, \quad \zeta_{a,1} = Q_a \otimes \theta_1, \quad \zeta_{a,2} = Q_a \otimes \theta_2 \quad \text{and} \quad \zeta_{b,0} = Q_b \otimes \theta_0, \quad \zeta_{b,1} = Q_b \otimes \theta_1, \quad \zeta_{b,2} = Q_b \otimes \theta_2,
\]

in which \( \theta_0, \theta_1 \) and \( \theta_2 \) are \( U \times U \) matrices respectively given by

\[
\theta_0 = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad \theta_1 = \begin{bmatrix} p_c/U & \cdots & \cdots & p_c/U \\ 0 & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & 0 \end{bmatrix}
\]

and

\[
\theta_2 = \begin{bmatrix} p_s/U & \cdots & \cdots & p_s/U \\ 0 & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & 0 \end{bmatrix}
\]

We shall explain the physical meaning of the matrices in (5.18) below, while construct-
Chapter 5. Queueing Performance of Multichannel ALOHA Systems

ing $B$, $B_a$, $D$ and $A$ matrices. We can write $B$ as

$$B = \begin{bmatrix}
\zeta_{a,2} + \zeta_{b,0} & \zeta_{b,1} & 0 & \cdots & \cdots & \cdots & 0 \\
\zeta_{a,2} & \zeta_{b,0} & \zeta_{b,1} & 0 & \cdots & \cdots & 0 \\
\zeta_{a,2} & 0 & \zeta_{b,0} & \zeta_{b,1} & 0 & \cdots & 0 \\
\vdots & 0 & 0 & \ddots & \ddots & 0 & \cdots & 0 \\
\vdots & 0 & 0 & \ddots & \ddots & 0 & \cdots & 0 \\
\zeta_{a,2} & \cdots & \cdots & 0 & \zeta_{b,0} & \zeta_{b,1} \\
\zeta_{a,1} + \zeta_{a,2} & 0 & 0 & 0 & \cdots & \cdots & 0 & \zeta_{b,0}
\end{bmatrix}, \tag{5.21}$$

which indicates the state transition from $k$ to itself, caused by events including backoff counter’s decrement without a packet arrival ($\zeta_{b,0}$), successful transmission with a packet arrival ($\zeta_{a,2}$), transmission failure without a packet arrival ($\zeta_{b,1}$), transmission failure with a packet arrival at the last retransmission stage ($\zeta_{a,1}$). When $J = 1$, we have $B = \zeta_{b,0} + \zeta_{a,1} + \zeta_{a,2}$.

Next, the matrix $D$ can be expressed as

$$D = \begin{bmatrix}
\zeta_{b,2} & 0 & 0 & \cdots & \cdots & 0 \\
\zeta_{b,2} & 0 & 0 & \cdots & \cdots & 0 \\
\zeta_{b,2} & 0 & 0 & \cdots & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\
\zeta_{b,1} + \zeta_{b,2} & 0 & 0 & \cdots & 0 & 0
\end{bmatrix}, \tag{5.22}$$

which represents the state transition from $k + 1$ to $k$ packets in the queue, caused by one of the following events: a successful transmission without a packet arrival ($\zeta_{b,2}$), or packet dropping upon $L$ retransmissions ($\zeta_{b,1} + \zeta_{b,2}$). For $J = 1$, we have $D = [\zeta_{b,1} + \zeta_{b,2}]$. 

101
Computational Procedure:

1. Set $p_s^{(0)} = 0.5$, do the following for $\nu = 1, 2, \ldots$
   
   (a) Construct the matrices $B, B_a, F, Q_b, A$ and $D$.
   
   (b) Set the initial probabilities, $\phi_k^{(0)} = \phi_k^{(0)} = ((KJU + 1)N)^{-1}$ for $1 \leq k \leq K$.
   
   (c) For $l = 1, 2, \ldots$, compute via block Gauss-Seidel iteration, using (5.24).
      
      Stop if $|\phi_{0,i}^{(l+1)} - \phi_{0,i}^{(l)}| < \varepsilon$ for $0 \leq i \leq N-1$.
   
   (d) Compute $p_s^{(\nu+1)}$ using (3.2) by substituting $p_r$ obtained from (5.26).

2. Stop if $|p_s^{(\nu+1)} - p_s^{(\nu)}| < \varepsilon_p$. Otherwise, go to (a) with $p_s^{(\nu+1)}$.

Finally, we can write the matrix $A$ as

$$A = \begin{bmatrix}
\zeta_{a,0} & \zeta_{a,1} & 0 & \cdots & \cdots & \cdots & 0 \\
0 & \zeta_{a,0} & \zeta_{a,1} & 0 & \cdots & \cdots & 0 \\
0 & 0 & \zeta_{a,0} & \zeta_{a,1} & 0 & \cdots & 0 \\
\vdots & 0 & 0 & \zeta_{a,0} & \zeta_{a,1} & \cdots & 0 \\
\vdots & \vdots & \vdots & \zeta_{a,0} & \zeta_{a,1} & \cdots & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & \zeta_{a,0}
\end{bmatrix}$$

(5.23)

which represents the state transition from $k$ to $k + 1$ packets in the queue, which is caused by one of the following events: backoff counter’s decrement with a packet arrival ($\zeta_{a,0}$), or an unsuccessful transmission with a packet arrival ($\zeta_{a,1}$). If $J = 1$, we have $A = \zeta_{a,0} + \zeta_{a,1}$.

Finally, we have $B_a = B + A$.

In order to obtain $\phi_k$, we transform (5.8)-(5.10) into a block Gauss-Seidel iteration form
Chapter 5. Queueing Performance of Multichannel ALOHA Systems

[97] as follows:

\[
\vec{\varphi}^{(l+1)}_0 = \vec{\varphi}_1 (I - Q_b)^{-1},
\]

\[
\vec{\varphi}^{(l+1)}_1 = \left( \vec{\varphi}_0^{(l+1)} F + \vec{\varphi}_2^{(l)} D \right) (I - B)^{-1},
\]

\[
\vec{\varphi}^{(l+1)}_k = \left( \vec{\varphi}_{k-1}^{(l+1)} A + \vec{\varphi}_{k+1}^{(l)} D \right) (I - B)^{-1} \quad \text{for } 2 \leq k \leq K - 1,
\]

\[
\vec{\varphi}^{(l+1)}_K = \vec{\varphi}_{K-1}^{(l+1)} A (I - B_a)^{-1}.
\] (5.24)

For \( K = 1 \) we have

\[
\vec{\varphi}^{(l+1)}_0 = \vec{\varphi}_1 (I - Q_b)^{-1} \quad \text{and} \quad \vec{\varphi}^{(l+1)}_1 = \vec{\varphi}_0^{(l+1)} F (I - B_a)^{-1}.
\] (5.25)

In (5.24) and (5.25), normalization is needed at each iteration \( l \), i.e., \( \vec{\varphi}_k^{(l+1)} = \vec{\varphi}_k^{(l+1)} / \eta \) and \( \eta = \sum_{k=0}^{K} \vec{\varphi}_k^{(l+1)} e^T \). We are now in a position to obtain \( p_r \) and \( p_s \). The retransmission probability \( p_r \) can be expressed as

\[
p_r = \sum_{k=1}^{K} \sum_{j=0}^{J-1} \sum_{i=0}^{N-1} \varphi_{k,j,i,0}.
\] (5.26)

which can be substituted back into (3.2). One can obtain the stationary probability \( \vec{\varphi}_k \) by the computational procedure at the top of this page.

As performance metrics, denote the mean number of packets buffered in a terminal and the waiting time of a packet by \( \overline{L} \) and \( \overline{D} \), respectively. We can obtain

\[
\overline{L} = \sum_{k=1}^{K} k \vec{\varphi}_k e^T.
\] (5.27)
By Little’s result, we can also obtain $\overline{D}$ by

$$\overline{D} = \frac{L}{(1 - P_B) \overline{\epsilon}}$$

which indicates the average delay of packets either successfully transmitted or dropped by retry limit. Denote the probability of packet dropping due to retry limit and the probability of packet blocking due to a full queue by $P_D$ and $P_B$, respectively. Then, the total packet loss probability $P_L$ is given by

$$P_L = P_D + P_B,$$

in which the packet dropping probability $P_D$ is obtained by

$$P_D = p_c \sum_{k=1}^{K} \sum_{i=0}^{N-1} \phi_{k,J-1,i,0},$$

and the packet blocking probability $P_B$ is given by

$$P_B = \sum_{j=0}^{J-1} \sum_{i=0}^{N-1} \sum_{u=0}^{U-1} \phi_{K,j,i,u}.$$ 

The unnormalized system throughput $\tau$ (packets/slot) can be obtained by

$$\tau = \sum_{n=0}^{M} \sum_{k=0}^{n} k \Lambda(k | n, P) \Omega(n, M, p_r).$$

We then obtain normalized throughput for this channel $\tau_N$ (packets/slot/code) by $\tau_N = \tau / P$. Finally we examine the mean busy period, $\overline{B}$, which is the average time period from the time that a packet arrives at an empty queue to the time that the queue becomes
empty again. It is given by [98]

$$\overline{B} = \frac{\overline{x}}{1 - \rho}$$  \hspace{1cm} (5.33)$$

where $\overline{x}$ and $\rho$ are the mean packet service time and the system utilization, i.e. $\rho = 1 - \overline{\varphi_0e^T}$.

In order to obtain $\overline{x}$, we denote by $x_k$ the probability that it takes $k$ slots for a terminal to transmit a packet. Further we define a matrix $S$ of size $JU \times JU$ for $J \geq 2$ as

$$S = \begin{bmatrix}
\theta_0 & \theta_1 & 0 & \cdots & \cdots & 0 \\
0 & \theta_0 & \theta_1 & 0 & \cdots & 0 \\
0 & 0 & \theta_0 & \theta_1 & \cdots & 0 \\
0 & 0 & \cdots & \cdots & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & \theta_0
\end{bmatrix}.$$ \hspace{1cm} (5.34)

As shown in (5.34), $S$ consists of the main and upper block-diagonal matrices, $\theta_0$ and $\theta_1$, respectively. Other elements are all zeros. For $J = 1$, we have $S = \theta_0 + \theta_1$. Due to DFT protocol, we have $x_1 = 0$. We can write $x_k$ for $k \geq 2$ as

$$x_k = \tilde{a}_s \cdot S^{k-2} \cdot \tilde{c}_s,$$ \hspace{1cm} (5.35)

in which $\tilde{a}_s$ and $\tilde{c}_s$ are respectively given by

$$\tilde{a}_s = \left[ \frac{1}{U} \cdots \frac{1}{U} \frac{1}{U} 0 0 \cdots 0 0 \right] \quad \text{and} \quad \tilde{c}_s = \left[ \overline{c}_0 \overline{c}_0 \cdots \overline{c}_0 \overline{c}_1 \right].$$ \hspace{1cm} (5.36)

For $J = 1$, these $U$ element vectors are modified as follows: $\tilde{a}_s = [1/U 1/U \cdots 1/U]$ and $\tilde{c}_s = [1 0 0 \cdots 0]$. One can obtain the generating function of $x_k$ as $X(z) = \sum_{k=2}^{\infty} x_k z^k =
Figure 5.2: Effect of multichannel on multi-buffered system, $p_{00} = 0.85$, $p_{11} = 0.65$, $U = 25$, $J = 3$, $K = 10$. 
Then we have
\[
\overline{\mathbf{r}} = X'(1) = \bar{a} (I - S)^{-1} \left( 2I + S(I - S)^{-1} \right) \bar{c}.
\] (5.37)

### 5.2.3 Single-buffered Terminals

In this part we consider a system, in which a terminal can hold only one packet for transmission. A packet is generated by an MMBP as well, but only when a terminal is in the idle state. Intuitively, since packet arrivals are correlated only in the idle state, this system’s behaviour is identical to the system with an independent source whose average number of packet arrivals per slot is equal to that of MMBP. However, for comparison with the system with \( K = 1 \) previously introduced in Section 5.2.2, we deal with this system as follows:

Denote by \( \varphi_0 = [\varphi_{0,1}] \) and \( \varphi_1 = [\varphi_{1,i,j,u}] \) respectively the stationary state probability vector that a terminal has no packet in the \( i \)-th source state, and that a terminal is retransmitting a packet at the \( u \)-th backoff counter of the \( j \)-th retransmission attempt in the \( i \)-th source state. Define \( Q_c \) and \( Q_s \) by

\[
Q_c = Q \otimes \bar{c}_s \quad \text{and} \quad Q_s = Q \otimes S
\] (5.38)

where \( \bar{c}_s \) is given by (5.36). Note that \( Q_c \) and \( Q_s \) describe the state transition of traffic source and packet retransmission process, while excluding the event of a packet arrival. Then, this system satisfies the following set of state balance equations:

\[
\varphi_0 = \varphi_0 Q_b + \varphi_1 Q_c \quad \text{and} \quad \varphi_1 = \varphi_0 F_0 + \varphi_1 Q_s
\] (5.39)
Figure 5.3: Performance comparisons of multi-buffered systems with various source correlation, $p_{11} = 0.6$, $U = 20$, $K = 10$. 
with normalization condition \((\vec{\varphi}_0 + \vec{\varphi}_1)\vec{e}^T = 1\). The matrix \(F_0\) is written by

\[
F_0 = Q_a \otimes \vec{a}_s
\]  

(5.40)

where \(\vec{a}_s\) is given by (5.36). We can get \(\vec{\varphi}_0\) and \(\vec{\varphi}_1\) by the similar way in (5.25) and \(p_s\) by (3.2). We can write \(p_r\) as

\[
p_r = \sum_{i=0}^{N-1} \sum_{j=0}^{J-1} \varphi_{1,i,j,0}.
\]  

(5.41)

The packet dropping probability is obtained by \(P_D = p_c \sum_{i=0}^{N-1} \varphi_{1,i,L-1,0}\). The retransmission delay is obtained from (5.37).

5.3 Numerical Studies

In order to verify our analysis, we build a simulation program with MATLAB. Each simulation result was obtained by a single run of 15000 slots, except Table 5.1. Each result in Table 5.1 was obtained by averaging 5 simulations, each running for 30000 slots. In the computational procedure for analytical results, we set \(\varepsilon_{\varphi} = \varepsilon_p = 10^{-6}\). For an independent Bernoulli process, we set \(\epsilon_0 = \epsilon_1 \neq 0\) in the analysis and simulation with MMBP model. We use a two-state ON-OFF MMBP to model the uplink traffic of Web-browsing. Unless otherwise stated, we set \(P = 1, U = 20, J = 6, K = 10\) and \(M = 100\). We also set \(p_{00} > p_{11}\) by assuming that the mean duration of an OFF period, i.e., reading times of a user, is greater than that of an ON period in (5.4). In Figs. 5.2(a)-5.6(c), lines and symbols denote analytical and simulation results, respectively.

1. Effect of multichannel access: In Figs. 5.2(a)-5.2(c), we examine the effects of the number of parallel RACHs, \(P\), on various performance metrics. Mostly, our analysis and
Chapter 5. Queueing Performance of Multichannel ALOHA Systems

simulation agree well with each other. In particular, we compare the real throughput $\overline{\tau}$ as well as the normalized throughput $\overline{\tau}_N$ for $P = 1, 2, 3$ in Fig. 5.2(a). As expected, a large $P$ increases the maximum of $\overline{\tau}$ by a factor of $P$ in Fig. 5.2(a). Correspondingly, this improves $\overline{q}$ and $P_L$ in Figs. 5.2(b) and 5.2(c), respectively. However, when normalized with the number of channels, the maximum value of $\overline{\tau}_N$ remains unchanged, but is shifted toward the right for larger $P$. For single channel access, i.e., $P = 1$, $\overline{\tau}$ goes to zero when $\overline{q}$ grows beyond one, as we can observe in Figs. 5.2(a) and 5.2(b). In contrast, Fig. 5.2(a) shows that under heavy traffic, some minimum level of throughput can be maintained when a larger value of $P$ is employed. In other words, multichannel S-ALOHA access reduces collisions and accepts more packets by distributing the system load evenly over the available channels.

2. Effect of source arrival correlation: In Figs. 5.3(a)-5.3(d), we use two different $p_{00} = 0.65$ and $0.75$ with fixed $p_{11} = 0.6$, i.e., source arrivals are positively correlated by $\omega = 0.25$ and $0.35$. We additionally depict the system performance with independent Bernoulli arrivals. As expected, the system performance is less saturated for larger $p_{00}$. While various performance metrics behave differently according to $\omega$, we find that tradeoff curves of $\overline{\tau}$ vs. $D$ in Fig. 5.3(d) seem to be insensitive to $\omega$. Based on the MMBP model, it is concluded that the system shows the same performance characteristics, as long as the same offered traffic $\overline{\tau}$ is applied to each terminal irrespective of $\omega$. Thus, referring to [89][90][91], Poisson arrival process could be used in conservative performance evaluation for multichannel S-ALOHA random access.

3. Single-buffered vs. multi-buffered terminals: We compare single-buffered and multi-buffered systems with $K = 1$ in Figs. 5.4(a)-5.4(d). In the multi-buffered system with $K = 1$, incoming packets are blocked, while only one packet is in retransmission. However, packet blocking cannot occur in the single-buffered system, since packets are not generated
Figure 5.4: Performance comparisons of single- (bufferless) and multi-buffered systems, 
$p_{00} = 0.75, p_{11} = 0.6, U = 20, J = 6, K = 1.$
during ongoing packet retransmissions. Therefore, $P_B$ in Fig. 5.4(d) refers to the multi-buffered system. The performances of the two systems seem to be almost identical in Figs. 5.4(a)-5.4(c). We define the saturation point of $\epsilon_1$ as the value that maximizes $\tau$. Fig. 5.4(d) shows that beyond the saturation point $\epsilon_1 = 0.012$, $P_B$ increases sharply over 10%. Note that a high $P_B$ can be prevented by using a large $K$. In addition, we depict the busy period $\overline{B}$ in Fig. 5.4(b), which sharply increases beyond the saturation point. It is observed that single-buffered and multi-buffered systems show almost identical performances except $P_B$. Referring to Figs. 5.2(a)-5.2(b) and 5.3(a)-5.3(b), when mean queue length exceeds one, the system throughput degrades severely except in the case of multichannel access. Accordingly, buffering arriving packets in S-ALOHA can be effective only in multichannel access. For single channel S-ALOHA, the single-buffered assumption may be enough to evaluate system performance except $P_B$.

4. Effect of retry limit $L$ and Correlation: As expected, the results in Figs. 5.5(a)-5.5(c) indicate that more collisions occur with larger $L$ values. For example, $\tau$ and $\overline{q}$ get worse in Figs. 5.5(a) and 5.5(b) beyond the saturation point $\epsilon_1 = 0.0125$. In other words, a small $L$ tends to stabilize the system by dropping packets that have not been sent successfully after a small number of retransmissions. On the other hand, a large $L$ prevents unnecessary early packet dropping for $\epsilon_1 < 0.0125$ in Fig. 5.5(c) and improves $\tau$ in Fig. 5.5(a) for $0.01 \leq \epsilon_1 \leq 0.0175$. It is observed that unnecessary early packet dropping occurs predominantly in Fig. 5.5(c) for $\epsilon_1 \leq 0.0175$ with $L = 2$. Thus, it can be concluded that $\tau$ and $\overline{q}$ can be improved by using a small $L$, at the expense of packet drops caused by the retry limit, while a large $L$ helps to prevent unnecessary early packet drops at traffic levels below the saturation point. Note that $P_B$ becomes a dominant factor of $P_L$ beyond the saturation point, i.e., once the channel gets congested. As defined in (5.7), we now check the correlation coefficient $\varrho$ by simulations according to $J$. We have $\varrho = 2.5545 \times 10^{-4}$.
Figure 5.5: Performance comparisons of multi-buffered systems with different retry limit $L$, $p_{00} = 0.85$, $p_{11} = 0.65$, $U = 20$, $K = 10$. 
(\(J = 6\)), 0.0034 (\(J = 8\)), 0.027 (\(J = 10\)), 0.1073 (\(J = 12\)) for a Bernoulli arrival of \(\epsilon_1 = 0.01\) and \(U = 20\). When more retransmissions are allowed (larger \(J\)), the queueing process is more positively correlated. However, finite retransmissions keep \(\rho\) small. For \(\epsilon_1 = 0.005\) (smaller traffic load) and \(J = 12\), we have \(\rho = 0.0021\), which indicates a very low positive correlation between terminals’ queues. It is concluded that the queueing process is so weakly (positively) correlated that the analysis based on i.i.d assumption can show agreement with simulations.

5. Effect of window size \(U\): Fig. 5.6(a) shows that a large \(U\) improves effectively the minimum \(\tau\) that the system can sustain at a high \(\epsilon_1\), while it increases \(\bar{q}\) (and correspondingly \(\bar{D}\)) and \(P_L\) as shown in Figs. 5.6(b) and 5.6(c), respectively. This comes from the fact that a large \(U\) causes packets to be held longer in the terminal’s buffer, while spreading the retransmission opportunities over time. Under light traffic, i.e., when the channel is not saturated, a smaller \(U\) is preferable to reduce \(\bar{D}\). Even though larger \(U\) values might be favourable in order to improve \(\tau\) in the saturated system as shown in Fig. 5.5(a), the resulting degradation in queuing performance at the terminal, as shown in Figs. 5.5(b) and 5.5(c), may not be acceptable in terms of meeting the user’s QoS requirements. Note that \(P_B\) becomes much more dominant in \(P_L\) beyond the saturation point for large \(U\) in Fig. 5.6(c) compared to the case for large \(L\) in Fig. 5.5(c). In summary, although small \(L\) and large \(U\) values yield better \(\tau\) especially for the saturated channel, this is achieved at the expense of \(P_L\). For Web-browsing, loss of TCP-ACK packets or HTTP requests can be harmful to the users’ experience. In this regard, it may only be effective to transport such type of packets over a RACH with retry limit when the channel is lightly loaded, i.e., unsaturated.

Finally, we test various window sizes with \(p_{00} = 0.98\) and \(p_{11} = 0.8\) in Table 5.1 in order to find the optimal \(U\) that maximizes \(\tau\) and minimizes \(\bar{d}\) through simulations. For
Figure 5.6: Performance comparisons of multi-buffered systems with different window size $U$, $p_{00} = 0.85$, $p_{11} = 0.65$, $L = 2$, $K = 10$. 
comparison, we consider that the BS has perfect knowledge of the number of backlogged terminals in every slot, which is denoted by $p$. Based on this backlog information, the BS assigns $U$ to be exactly the same as the current backlog size, which is known to be optimal for an S-ALOHA system with single-buffered terminals and Poisson arrivals [15]. Table 5.1 shows that backoff window assignment based on the exact backlog size does not yield optimal $\bar{\tau}$ and $\bar{d}$ for multi-buffered terminals with correlated arrivals ($M = 100$, $J = 5$, $K = 10$, $P = 1$). In fact, under such conditions, the fixed assignment scheme, e.g., $U = 15$ for $\epsilon_1 = 0.053$ and $U = 25$ for $\epsilon_1 = 0.056$, can give much better performance.

6. End-to-End TCP performance: So far, we have discussed a single hop performance between a BS and terminals. In order to provide high and reliable end-to-end TCP performance, the system should provide low packet dropping probability and low access delay. Although the system shows quite different behaviours after its peak throughput performance, i.e. 0.368, in Figs. 5.2, 5.3, 5.5, and 5.6 given the parameters of UB algorithm, the system should operate before its throughput reaches peak performance.

Table 5.1: Performance metrics with various backoff window sizes.
5.4 Summary

In this chapter, we have analyzed the queuing performance of a multichannel S-ALOHA system employing the UB algorithm with retry limit for contention resolution, which reasonably models the random access channels in LTE. The analytical model considers multi-buffered terminals with an MMBP traffic source that closely models the uplink traffic of Web-browsing. We have presented results to show that S-ALOHA can be effectively used for uplink transmissions of small IP packets, if the random access channel is unsaturated. Once the system is saturated, the terminals queuing performance is severely degraded. The use of multichannel random access can effectively reduce such performance degradation as well as improve system throughput. Further, compared to the more complex MMBP traffic model, Poisson arrivals can be used for a conservative performance evaluation on single- and multi-buffered terminals. Regarding the UB algorithm, we have observed that increasing the retry limit provides higher throughput and better queuing performance under light traffic load, while increasing the window size provides a higher throughput in a saturated channel, all at the expense of a higher packet loss rate. Users’ QoS requirements, e.g., for Web-browsing, may not be able to tolerate a high packet loss rate and should be taken into account when such traffic is sent over the random access channels. Finally, we have found that the optimal window size in the literature, based on perfect knowledge of the number of backlogged terminals, is not optimal for multi-buffered terminals with retry limit.
Chapter 6

Semi-Persistent Scheduling Algorithm in LTE

LTE has emerged as a candidate for fourth generation (4G) cellular networks, which will meet explosive demands on wireless bandwidth by providing peak data rates of 300 Mbps over downlink with $4 \times 4$ MIMO antennas and 75 Mbps over uplink. In order to use these radio resources efficiently, LTE adopts two scheduling mechanisms. The first one is dynamic scheduling, which allocates radio resources dynamically based on each terminal’s buffer status and radio channel state information. The size of resources together with the MCS would be allocated to meet some QoS requirements. To support this algorithm, terminals must send to eNodeB (a base station in LTE) certain uplink reporting messages either via a random access channel, or via some dedicated control channels in order to inform eNodeB of associated buffer status and channel quality. Although this algorithm is expected to improve total radio resource utilization in the provisioning of QoS for each terminal, it could consume substantial control signaling overhead, particularly for traffic with small, periodic and semi-static packets, e.g., VoIP or TCP ACK packets. The second mechanism is called SPS, which allocates an uplink traffic channel periodically without any additional control message during a traffic burst, e.g., the ON period of a VoIP traffic stream. At the beginning of an SPS operation, an initial uplink traffic channel is obtained either by random access or via an assigned control channel, over which a periodic reservation is made. When an assigned control channel is used for reserving a traffic channel at the
beginning of SPS operation, an scheduling request (SR) indicator based on on/off keying is transmitted first in an assigned control channel, called the physical uplink common control channel (PUCCH) [99]. Then eNodeB allocates an uplink grant that allows the terminal to send a buffer status reporting message as shown in Fig. 6.1. Finally eNodeB grants a traffic channel to the terminal. When the SR indicator is not sent over PUCCH, H-ARQ ACK and negative-ACK (NACK), and radio channel quality indicator (CQI) can share this PUCCH. While SPS with SR indicator (SRI) over PUCCH can reduce channel set-up times significantly, PUCCH for SRI may be wasted during long inactive periods of an application. On the other hand, when a terminal has lost time-synchronization during an inactive period, it is mandatory to perform random access at the beginning of the next SPS operation. Moreover, radio resource limitations for PUCCH would only allow applications with strict delay constraints to use the SPS with SRI.

![Figure 6.1: Signal flow of SPS with SRI or with random access.](image)

In SPS, the allocated traffic channel would be implicitly released when some empty transmission slots are found on the allocated traffic channel. This is called ‘implicit release after’, which is specified as either 2, 3, 4 or 8 transmissions [6]. Note that, in the SPS
operation, the size of resources and MCS are not changed, once they are allocated at the beginning of SPS. The allocation interval of the traffic channel is called the ‘SPS interval in uplink’ [6], which is specified as either 10, 20, 32, 40, 64, 80, 128, 160, 320 or 640 subframes. Note that VoIP traffic might typically use an SPS interval of 20 subframes [99], while other intervals might be used according to the uplink traffic characteristics. Hereafter, we denote the value of ‘implicit release after’ and ‘SPS interval in uplink’, respectively, by $K$ and $L$.

In SPS operation, $K$ is used for detecting and confirming an ensuing silence period, or can help to hold the reserved traffic channel to prevent it from being released due to some packets being lost over the radio channel. However, a large $K$ could cause low throughput of the reserved traffic channels. In this chapter, we investigate the feasibility of using SPS with initial random access for VoIP traffic in order to complement SPS with SRI, and characterize its performance based on $K$.

While SPS in LTE can be used for VoIP traffic as well as possibly uplink web traffic, VoIP traffic scheduling algorithms have been also developed in IEEE 802.16 systems\(^6\), which is another candidate for 4G networks. In [100], Lee et al. proposed a VoIP scheduling algorithm for IEEE 802.16 system, in which a terminal with VoIP traffic informs a BS of its voice traffic state through an uplink dedicated control channel. According to a one-bit indication in this dedicated control channel, BS allocates uplink resources to the terminal during ON periods of VoIP traffic and de-allocates them during OFF periods. Additionally, they proposed another VoIP scheduling algorithm [101], in which a bandwidth request is sent at the transition from OFF- to ON-state of VoIP traffic through the uplink resources periodically allocated by polling. In order to transmit the bandwidth request generated at the transition from OFF- to ON-state without random access, [100][101] considered VoIP scheduling algorithms with a control channel that is periodically allocated during OFF pe-

---

\(^6\)As in LTE, IEEE 802.16 systems are also based on OFDMA, in which some subcarriers are used for random access channels. Particularly, both systems employ a demand-assignment multiple access protocol consisting of 4-way handshaking procedure.
riods. In addition, when VoIP service is supported by an extended real-time polling service in IEEE 802.16 system, [102] examined signalling overhead and queuing performance of terminals and a base station. In order to prevent wastage of this dedicated uplink control channel [100] during OFF periods, Oh et al. [103] proposed a VoIP scheduling algorithm for IEEE 802.16 system that employs random access at the beginning of a talk spurt in order to obtain uplink resources. In this algorithm, uplink resources for VoIP traffic are periodically granted during ON periods, while silence descriptor frames are transmitted by random access during OFF periods. Note that VoIP scheduling with a dedicated control channel [100] and with random access [103] in IEEE 802.16 system is quite similar to SPS with SRI and with random access in LTE, respectively. When random access is used for reserving a traffic channel, its success may depend on contentions in the random access channels as well as current availability of traffic channels. When all traffic channels are currently occupied by other terminals, a requesting terminal will not be allocated a traffic channel even though the random access is successful. In this regard, [103] underestimates traffic channel assignment delay by assuming that the success of random access is independent of currently remaining traffic channels. In [104] Bi et al. consider that terminals use signalling channels at the beginning of VoIP talk spurs in an OFDMA system in order to obtain a traffic channel and then transmit VoIP packets in the reserved traffic channels once the signalling message would be granted, which is modelled by a tandem $M/M/m$ queuing system. This model is also very close to SPS with SRI. In [105] Rinne et al. provide a simulation study for Evolved Universal Terrestrial Radio Access (E-UTRA) for VoIP and best effort traffic streams, in which the standards of physical and medium access layers are considered in detail. Particularly, SPS operation is compared with a fully dynamic scheduling algorithm with and without packet bundling\footnote{In packet bundling, multiple VoIP packets are bundled based on channel quality and then transmitted together over the downlink by a dynamic scheduler.}, while round robin and
proportional fair channel-dependent scheduling algorithms are considered for BE traffic together with MIMO and some MCSs. Since the same MCS allocated at the beginning of a talk spurt would be used up to the end of the talk spurt, Jin et al. [106] consider an optimal MCS selection scheme with HARQ over a time-correlated fading channel with the objective of minimizing radio resource usage.

In contrast with the previous work that have focused on some underlying transmission schemes at the physical layer, such as MCS and H-ARQ with imperfect channel conditions while considering dedicated control channel, in this chapter we consider the interdependence between random access and traffic channel occupancy of VoIP traffic when the SPS with random access is employed. Our contributions are summarized as follows.

1. We characterize the performance of SPS with initial random access in LTE in terms of random access throughput and delay, traffic channel throughput, and packet dropping probability with a given delay constraint. Based on these performance metrics, we obtain VoIP capacity as the number of terminals without violating some packet dropping probability.

2. We examine by simulations and analysis system stability caused by random access with UB algorithm in terms of the probability distribution of the number of backlogged terminals in the system. We then show how to mitigate bistability of the system.

The remainder of this chapter is organized as follows. We introduce SPS with random access and make some assumptions in Section 6.1. In Section 6.2 we analyze this system by using Equilibrium Point Analysis (EPA) [20] as an approximation. We verify our analysis against simulations in Section 6.3. Concluding remarks are given in Section 6.4.
6.1 Models and Assumptions

6.1.1 Random Access for SPS Algorithm

In addition to the description on the RACH in LTE in Section 1.1, we introduce more details in the specification of LTE RACH. In an LTE radio frame, each PRACH occupies six radio resource blocks\(^8\) (RBs) in the frequency domain. The PRACH appears periodically, as specified by the PRACH configuration index. For instance, one PRACH is found every two subframes when the PRACH configuration index is 12 \([66]\). At most one PRACH is allowed in a subframe, while only one PRACH is available in the frequency domain of a slot. Note that, in an LTE FDD frame structure, one radio frame of 10 msec consists of 10 subframes, each of which consists of two slots. Thus, one slot length is 0.5 msec, which corresponds to one packet transmission time. Each accessing terminal randomly chooses one of 64 orthogonal RAPs called Zadoff-Chu sequences and transmits it over a PRACH. Some of the RAPs would be reserved for contention-free random access, whereby eNodeB directly allocates them to some terminals. In SPS with initial random access, contention-based random access is used for reserving a traffic channel that is periodically allocated to a terminal. Suppose that \(P\) RAPs are available for contention-based random access. First the requesting terminal selects one of \(P\) RAPs at random and transmits it with initial power \(F\) in a PRACH. When multiple terminals transmit the same RAP in the same PRACH, a RAP collision occurs. Secondly, eNodeB broadcasts the RAR on the downlink control channel within a predefined time-window as shown in Fig. 6.1. From the RAR message terminals obtain the backoff parameter, \(U\), timing advance command, the RAP identifier and the uplink RBs for the successful terminals to use. If a requesting terminal cannot find its RAP identifier, it regards itself as backlogged and increases its RAP transmission counter by one. If the counter is equal to the maximum number of allowed

\(^8\)Each radio RB is formed by 12 subcarriers.
preamble transmissions, denoted by $R_L$, then the terminal drops its RAP retransmission and reports this to the higher layer. Otherwise, the terminal selects a random backoff time between 0 and $U$ (msec), and delays its retransmission by the selected backoff time. When a terminal retransmits a RAP, it increases the RAP transmission power by $\Delta F$, subject to the constraint that the RAP transmission power cannot exceed $F_{\text{max}}$. This is called the power ramping scheme, which is repeated up to the maximum of RAP counter until the terminal transmits a RAP successfully. Third, a terminal finding its RAP identifier in the RAR message transmits a message called $ Msg_3 $ [6] via H-ARQ conveying information such as a radio resource control (RRC) connection request, or tracking area update. We assume that the terminal’s buffer status reporting in Fig. 6.1 could be transmitted via $Msg_3$ in SPS with random access in Fig. 6.1. If this message is successfully transmitted, eNodeB finally allocates a traffic channel, which the corresponding terminal will use periodically.

Based on the random access procedure above [6][66], our assumptions are summarized as follows:

A1) We assume that there are $M$ terminals in this system, each of which has an infinite queue to store incoming VoIP packets.

A2) Since LTE has various PRACH configurations, as an example we assume that one slot consists of one PRACH and $D$ traffic OFDMA subcarriers, with each slot corresponding to one packet transmission time.

A3) We assume perfect orthogonality among RAPs transmitted in the PRACH, i.e., no multiple access interference occurs between different RAPs. Therefore, the LTE random access channels can be regarded as a multichannel S-ALOHA system. Although the standard includes a power ramping scheme for RAP transmissions over the PRACH, we do not take power ramping into account to keep our model simple and tractable.
A4) In practice, when a RAP collision occurs, all the terminals that transmitted the same RAP will receive the same traffic channel to transmit SR requests. The eNodeB allows the RAP-colliding terminals to simultaneously transmit an Msg3 in the channel allocated to all of them [6], with the expectation that the collisions on this channel would be resolved by HARQ. If terminals cannot transmit Msg3 successfully up to the maximum of retransmissions of HARQ, they will repeat RAP transmission. We simplify our model by excluding Msg3 transmissions via HARQ. Moreover, we conservatively assume that a RAP collision destroys all transmissions involved and the corresponding terminals retransmit randomly selected RAPs based on UB algorithm after RAR reception. We further assume that a terminal gets an uplink traffic channel right after successful RAP transmission, skipping the Msg3 transmission. In fact, an Msg3 transmission after a successful RAP transmission merely adds a deterministic delay to the random access delay.

A5) For a terminal transmitting a unique RAP, i.e., successful RAP transmission, its random access cannot be declared as successful if there is no traffic channel available for assignment to it. In this paper, a random access is said to be successful, when a terminal receives a traffic channel assignment after successful RAP transmission. Thus, a successful random access (different from a successful RAP transmission) is the synonym of a successful traffic channel reservation. Accordingly, the random access delay is defined as the time that is taken for a terminal to reserve a traffic channel successfully, after it transmits the first RAP. Furthermore, we assume that terminals get feedback on success or failure of random access, just after RAP (re)transmissions, without any error, i.e., right before the beginning of the next slot.

A6) Compared to UB algorithm with retry limit [6], our analytical model uses UB algorithm with infinite retransmissions, $R_L = \infty$. However, our simulation model
Chapter 6. Semi-Persistent Scheduling in LTE

implements UB algorithm with retransmission limit for a better representation of a real system.

A7) In addition to A5, if the number of terminals successfully transmitting RAPs, say \( z \), is greater than the number of available traffic channels, say \( d \), eNodeB randomly chooses \( d \) among \( z \) terminals and allocates them \( d \) traffic channels. Then, \( z - d \) terminals will be backlogged due to the lack of available traffic channels and will perform the backoff algorithm in A6, i.e., these terminals suffer from a reservation blocking.

6.1.2 Uplink Traffic Model

When each VoIP packet is generated every 20 msec by an adaptive multi-rate (AMR) voice codec at 12.2 kbps [99], a total of 50 packets/sec are generated with the payload size of 31 bytes. We approximately model this VoIP traffic stream by a two-state MMBP [77]. We assume that in every slot a traffic source that is in the OFF-state may change its state to the ON-state with probability \( 1 - \beta \), and a traffic source that is in the ON-state may change its state from to the OFF-state with probability \( 1 - \alpha \). This assumption gives average ON and OFF periods as \( 1/(1 - \alpha) \) and \( 1/(1 - \beta) \) (slots), respectively. We further assume that one packet is always generated at the transition from the OFF-state to the ON-state, while a packet is generated every slot based on a Bernoulli trial with probability \( \sigma \) during the rest of the ON period. As in [107], in this paper we assume the average ON and OFF periods to be 1000 and 1350 (msec), which respectively correspond to \( \alpha = 0.999 \) and \( \beta = 0.9993 \) for 1 msec slots. We also set \( \sigma = 0.05 \), which represents a 20 msec average inter-arrival time of VoIP packets during the ON periods. Particularly, after a terminal would reserve a traffic channel by random access, we assume in the next section that no more than one packet can be generated in the ON state during each transmission period.
Chapter 6. Semi-Persistent Scheduling in LTE

Table 6.1: Symbols used in SPS algorithm with initial random access

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P$</td>
<td>Number of orthogonal random access preambles</td>
</tr>
<tr>
<td>$M$</td>
<td>Total number of terminals</td>
</tr>
<tr>
<td>$U$</td>
<td>Window size of uniform backoff algorithm</td>
</tr>
<tr>
<td>$L$</td>
<td>SPS interval in uplink. Allocated data channel appears every $L+1$ slots</td>
</tr>
<tr>
<td>$K$</td>
<td>Implicit release after. Maximum of allowed empty transmissions on an allocated traffic channel</td>
</tr>
<tr>
<td>$D$</td>
<td>Number of data channels</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>The probability that a packet is generated in the ON state in slot</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>The probability of VoIP traffic source being in the ON (OFF) state in a slot</td>
</tr>
<tr>
<td>$c_i$</td>
<td>Number of backlogged terminals with window size $i$</td>
</tr>
<tr>
<td>$n$</td>
<td>Number of terminals with ON state of VoIP traffic source</td>
</tr>
<tr>
<td>$f$</td>
<td>Number of terminals with OFF state of VoIP traffic source</td>
</tr>
<tr>
<td>$t_0$</td>
<td>Number of terminals in the transmission state, when VoIP traffic source is in ON (OFF) state</td>
</tr>
<tr>
<td>$t_1$</td>
<td></td>
</tr>
<tr>
<td>$r_{ij}^{(k)}$</td>
<td>Number of terminals with a reserved traffic channel. The indexes, $k$, $i$ and $j$ correspond to $R_{ij}^{(k)}$</td>
</tr>
</tbody>
</table>

of $L+1$ slots.

6.1.3 Semi-Persistent Scheduling in LTE

To begin with, we summarize the symbols used in describing the SPS withinitial random access in Table 6.1. Recall that, after the traffic channel reservation via random access, terminals transmit their VoIP packets in the allocated traffic channels every $L+1$ slots. During this period, no more packets are generated in the ON-state. Thus, packet arrivals in the ON state of our MMBP model depends on the terminal’s reservation state. An allocated traffic channel is implicitly released, when $K$ consecutive empty transmissions are observed in the allocated traffic channel. After the reserved traffic channel has been released, the terminal should perform random access again when it needs to send more VoIP packets. The SPS with initial random access we propose can be regarded as an extension of the packet reservation multiple access (PRMA) in [107], which is based on S-ALOHA.
for a traffic channel reservation. Compared to the PRMA with $P = D = K = 1$ where
a permission probability is used as a backoff algorithm, terminals in our model reserve
a traffic channel by multichannel S-ALOHA ($P \geq 1$, $D \geq 1$, and $K \geq 1$) and use UB
algorithm. Fig. 6.2 shows the possible states of each terminal with respect to the random
access, which can be explained as follows:

1. A terminal’s uplink traffic source behaves every slot based on MMBP model in Section
6.1.2. When the terminal has not started the random access to reserve a traffic
channel, the state of the source is depicted by the ON- and the OFF-state in Fig. 6.2,
which is also the state of the terminal. While the terminal does not have a reserved
traffic channel, a transition from the OFF- to the ON-state (always accompanied by
a VoIP packet arrival) or the arrival of a VoIP packet while the terminal is in the ON-
state causes the terminal to contend for a traffic channel assignment by accessing the
random access channel. This moves the terminals state to one of the contention states
(\text{CON}_i for $0 \leq i \leq U$) based on the UB algorithm with delayed first transmission.

2. Once a terminal enters the \text{CON}_i state for reserving a traffic channel, we assume
that its VoIP source state is still in the ON state until a successful random access,
as a VoIP source is unlikely to change its state during the signalling delay in [104].
Every slot a terminal in the \text{CON}_i state moves into the \text{CON}_{i-1} state with probability
one, according to the UB algorithm. Note that, while a terminal is in the \text{CON}_i for
$0 \leq i \leq U$, packets can be generated. This will be discussed in detail in Sections
6.2.2. Terminals in the \text{CON}_0 state will select a RAP randomly and transmit it in
the PRACH.

3. When a terminal reserves a traffic channel, it moves from the \text{CON}_0 state into the
traffic transmission state, denoted by the \text{TX}_0 state. The \text{TX}_1 state indicates the
traffic transmission state in which the terminal’s voice traffic state is in the OFF state.

4. During $L$ subsequent slots after the $TX_0$ or the $TX_1$ state, the terminal is said to be in the $i$-th reservation state, which is denoted by the $RES_{i,j}^{(k)}$ for $1 \leq i \leq L$ and $1 \leq k \leq K$ (this superscript denotes the counter of the empty transmissions). Note that during these states the terminal can have a packet in the ON state of VoIP source, and some slots later VoIP source goes to the OFF state. Accordingly, the subscript $j = 0 (1)$ denotes the ON (OFF) state of VoIP traffic source with a packet, while $j = 2 (3)$ denotes the ON (OFF) state without a packet.

5. Finally, the $RES_{0,j}^{(k)}$ state for $2 \leq k \leq K + 1$ and $j = 2, 3$ represents an empty transmission on the traffic channel reserved for a terminal, which can be reached from the $RES_{L,j}^{(k-1)}$ state without a packet arrival for $j = 2, 3$. 

Figure 6.2: State transition diagram of terminals with SPS algorithm with random access.
6.2 Analysis

6.2.1 Equilibrium Point Analysis

Based on Fig. 6.2, we denote the numbers of terminals in the ON-, the OFF-, the TX$_0$-, the TX$_1$-, the RES$_{i,j}^{(k)}$, and the CON$_\ell$ states by $N$, $F$, $T_0$, $T_1$, $R_{i,j}^{(k)}$ and $C_\ell$ (called the state variable), respectively. The state space of this system can be represented by \{$C_\ell$ for $0 \leq \ell \leq U$, $T_0$, $T_1$, $N$, $F$, $R_{i,j}^{(k)}$ for $1 \leq k \leq K$ and $j = 0, 2, 3$, $R_{0,j}^{(k)}$ for $2 \leq k \leq K + 1$, $j = 2, 3$\}. If a multidimensional Markov chain is considered to describe $M$ terminals’ states, the number of possible states could be as large as $M^{2+U+1(D+1)K(5+4(L-1))+2}$ states in Fig. 6.2, in which $T_0$, $T_1$, and $R_{i,j}^{(k)}$ take the value from 0 to $D$. Typically we use $M \geq 70$, $U \geq 10$, $L = 20$, $K \geq 1$ and $D = 5$ in Section 4.3. Instead of finding a joint probability of $M$ terminals’ states, by using EPA we focus on the number of terminals in a state at equilibrium. In what follows, we first obtain a system equilibrium function denoted by $F(c_0)$, in which $c_0$ is the number of terminals in the CON$_0$ state at equilibrium. Then, we obtain the system performance metrics in terms of random access delay, throughput, and packet (front-end) dropping probability.

To this end, we denote the equilibrium value of each state variable by small letters, $c_\ell$, $t_0$, $t_1$, $n$, $f$, $r_{i,j}^{(k)}$, respectively. Additionally, we define the following row vectors, \( \vec{r} = [t_0 \ t_1] \), \( \vec{r}_{1}^{(k)} = [r_{1,0}^{(k)} \ r_{1,2}^{(k)} \ r_{1,3}^{(k)}] \) for $1 \leq k \leq K$, \( \vec{r}_{i}^{(k)} = [r_{i,0}^{(k)} \ r_{i,1}^{(k)} \ r_{i,2}^{(k)} \ r_{i,3}^{(k)}] \) for $1 \leq k \leq K$ and $2 \leq i \leq L$, \( \vec{r}_{0}^{(k)} = [r_{0,2}^{(k)} \ r_{0,3}^{(k)}] \) for $2 \leq k \leq K + 1$. Based on A2 the state transition occurs at each slot boundary. By assuming that the expected flow into and out of each state variable are equal [20], we construct the following state balance equations: First, at the RES$_{i,j}^{(k)}$ state for $1 \leq k \leq K$ and $0 \leq j \leq 3$, we have

\[
\begin{align*}
\vec{r}_{1}^{(k)} &= \vec{r}^H \mathcal{H}^{k-1} g, \\
\vec{r}_{2}^{(k)} &= \vec{r}^H \mathcal{H}^{k-1} g \hat{h} \quad \text{and} \quad \vec{r}_{i}^{(k)} &= \vec{r}^H \mathcal{H}^{k-1} g \hat{h} \hat{h}^{i-2} \quad \text{for} \ 3 \leq i \leq L, \\
\end{align*}
\]
in which $\mathcal{H} = g \hat{h} h^{L-2} u_0$, and the matrices, $g$ and $\hat{h}$, are respectively expressed as

$$
g = \begin{bmatrix}
\sigma\alpha & 0 & (1-\sigma)\alpha & 1-\alpha \\
1-\beta & 0 & 0 & \beta
\end{bmatrix}
$$

and

$$
\hat{h} = \begin{bmatrix}
\alpha & 1-\alpha & 0 & 0 \\
0 & 0 & 0 & 0 \\
\sigma\alpha & 0 & (1-\sigma)\alpha & 1-\alpha \\
1-\beta & 0 & 0 & \beta
\end{bmatrix}.
$$

(6.2)

In (6.1), for instance, $\vec{r}_1^{(1)} = \vec{t} g$ indicates that the number of terminals in the TX$_0$- and the TX$_1$ state moves to one of the RES$_{(1)}^{(1)}$ state for $j = 0, 2, 3$ with probability $g$. The matrices $h$ and $u_0$ are expressed as

$$
h = \begin{bmatrix}
\alpha & 1-\alpha & 0 & 0 \\
1-\beta & \beta & 0 & 0 \\
\sigma\alpha & 0 & (1-\sigma)\alpha & 1-\alpha \\
1-\beta & 0 & 0 & \beta
\end{bmatrix}
$$

and

$$
u_0 = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
(1-\sigma)\alpha & 1-\alpha \\
0 & \beta
\end{bmatrix}.
$$

(6.3)

For the RES$_{(k+1)}^{(j)}$ state for $j = 2, 3$, we can write

$$
\vec{r}_0^{(k+1)} = \vec{t} \mathcal{H}^k \text{ for } k = 1, 2, \cdots, K.
$$

(6.4)

We denote by $\Lambda_r(c_0)$ the number of packets which successfully reserve the remaining traffic channels by random access, given total of $r$ currently reserved traffic channels and $c_0$ terminals at the CON$^0$ state. We can get the total number of reserved traffic channels $r$ by

$$
r = \sum_{k=1}^{K} \vec{r}_L^{(k)} \vec{e}_4 = \sum_{k=1}^{K} (\mathcal{H}^{k-1}g\hat{h}h^{L-2}) \vec{e}_4,
$$

(6.5)
in which $\vec{e}_n$ represents a unit column vector whose length is $n$. At the TX$_0$- and the TX$_1$ state we can write

$$
\begin{align*}
t_0 &= \sum_{k=1}^{K} \tilde{r}_L^{(k)} u_1^o + \Lambda (r, c_0) \quad \text{and} \quad t_1 = \sum_{k=1}^{K} \tilde{r}_L^{(k)} u_1^*,
\end{align*}
$$

(6.6)

in which the column vectors, $u_1^o$ and $u_1^*$, are expressed as

$$
\begin{align*}
\begin{bmatrix}
\alpha \\
1 - \beta \\
\sigma \alpha \\
1 - \beta
\end{bmatrix}
\quad \text{and} \quad
\begin{bmatrix}
1 - \alpha \\
\beta \\
0 \\
0
\end{bmatrix}.
\end{align*}
$$

(6.7)

Referring to Fig. 6.2, the left hand side and right hand side in (6.6) respectively indicate the expected flow out of $t_0$ (or $t_1$) and into $t_0$ (or $t_1$). Using (6.1), we can rewrite $t_1$ in (6.6) as

$$
\begin{align*}
t_1 &= \tilde{t} \sum_{k=1}^{K} \mathcal{H}^{k-1} \hat{g} \hat{h} L^{-2} u_1^* \Rightarrow t_1 = a_0 t_0 + a_1 t_1,
\end{align*}
$$

(6.8)

from which we have $t_1 = a_0/(1 - a_1) t_0$. Note that the constants $a_0$ and $a_1$ are numerically determined. We then express $\tilde{t}$ in terms of $t_0$ as

$$
\tilde{t} = t_0 \vec{w},
$$

(6.9)

in which we have $\vec{w} = [1 \ a_0/(1 - a_1)]$. By substituting (6.9) into (6.5), we can get $\vec{r}$ by

$$
\begin{align*}
\vec{r} \triangleq \vec{r}(t_0) &= t_0 \vec{w} \sum_{k=1}^{K} (\mathcal{H}^{k-1} \hat{g} \hat{h} L^{-2}) \vec{e}_4.
\end{align*}
$$

(6.10)
In (6.6) we can get $\Lambda(r, c_0)$ by

$$\Lambda(r, c_0) = \sum_{z=1}^{c_0} m_r(z) \Gamma(z|c_0, P), \quad (6.11)$$

in which $m_r(z) = \min(z, \max(D - r, 0))$ is the conditional mean of successfully reserved traffic channels by random access, which is based on A7. Further, $\Lambda(z|c_0, P)$ is the probability that $z$ among $c_0$ terminals select a unique RAP given a total of $P$ RAPs, which comes from A3 in the Section 6.1.1 and is given in (3.3). Due to fluid approximation of EPA, $c_0$ is a real number for $0 \leq c_0 \leq M$ in (6.11), which makes (6.11) computationally intractable. By assuming that the terminals at the CON$_0$ state transmit their packet by a Poisson distribution with mean $c_0$, we rewrite (6.11) as

$$\Lambda(r, c_0) = \sum_{k=1}^{\infty} \sum_{z=1}^{k} m_r(z) \Gamma(z|k, P) \frac{c_0^k}{k!} \exp^{-c_0}. \quad (6.12)$$

At the ON state we have

$$(1 - \sigma)\alpha r_0^{(K+1)} = (\sigma\alpha + (1 - \alpha)) n. \quad (6.13)$$

At the OFF state we can write

$$(1 - \alpha)r_{0,2}^{(K+1)} + \beta r_{0,3}^{(K+1)} + (1 - \alpha)n = (1 - \beta)f. \quad (6.14)$$

At the CON$_U$ state we get

$$\frac{1}{U + 1} \left[ \sigma\alpha \left( n + r_{0,2}^{(K+1)} \right) + (1 - \beta) \left( f + r_{0,3}^{(K+1)} \right) + c_0 - \Lambda(r, c_0) \right] = c_U. \quad (6.15)$$
At the CON \( i \) state for \( 0 \leq i \leq U - 1 \) we also get
\[
\frac{1}{U + 1} \left[ \sigma \alpha \left( n + r_{0,2}^{(K+1)} \right) + (1 - \beta) \left( f + r_{0,3}^{(K+1)} \right) + c_0 - \Lambda(r, c_0) \right] + c_{i+1} = c_i. 
\] (6.16)

Using (6.15) we can rewrite \( c_i \) for \( 0 \leq i \leq U \)
\[
c_i = \frac{U + 1 - i}{U + 1} \left[ \sigma \alpha \left( n + r_{0,2}^{(K+1)} \right) + (1 - \beta) \left( f + r_{0,3}^{(K+1)} \right) + c_0 - \Lambda(r, c_0) \right].
\] (6.17)

At the CON\(_0\) state we then have
\[
\sigma \alpha \left( n + r_{0,2}^{(K+1)} \right) + (1 - \beta) \left( f + r_{0,3}^{(K+1)} \right) = \Lambda(r, c_0),
\] (6.18)
in which the left hand side indicates the expected flow into the backlog states, i.e., CON\(_i\) for \( 0 \leq i \leq U \), while \( \Lambda(r, c_0) \) indicates the expected output flow from the CON\(_0\) state.

After some tedious manipulation on (6.13)-(6.18), we have
\[
r_{0,2}^{(K+1)} + r_{0,3}^{(K+1)} = \Lambda(r, c_0) \Rightarrow t_0 \bar{w} \overline{H^K e_2} = \Lambda(\tilde{r}(t_0), c_0),
\] (6.19)
in which (6.10) has been used. Eq. (6.19) means that the flow out of the RES\(_{0,2}^{(K+1)}\) and the RES\(_{0,3}^{(K+1)}\) state is equal to the flow out of the CON\(_0\) state. If we can express (6.19) as a function of \( c_0 \) only, we can find a solution \( c_0 \) of satisfying (6.19). Thus, we only need to express \( t_0 \) as a function of \( c_0 \). To this end, we use the fact that the sum of the number of terminals distributed over each state should be equal to \( M \), which is expressed as
\[
\tilde{e}_2 + \sum_{k=1}^{K} \sum_{i=1}^{L} \tilde{r}_i^{(k)} e_4 + \sum_{k=2}^{K+1} r_0^{(k)} e_2 + \sum_{i=0}^{U} c_i + n + f = M.
\] (6.20)
From (6.13) and (6.14), we have

\[ n = \frac{(1 - \sigma)\alpha}{\sigma\alpha + (1 - \alpha)} r^{(K+1)}_{0,2} \]

and

\[ f = \frac{1}{1 - \beta} \left[ (1 - \alpha) \left( 1 + \frac{(1 - \sigma)\alpha}{\sigma\alpha + (1 - \alpha)} r^{(K+1)}_{0,2} + \beta r^{(K+1)}_{0,3} \right) \right]. \]

Then, we can get

\[ n + f = t_0 \bar{w} \mathcal{H}^K v \bar{e}_2, \]

in which the matrix \( v \) is expressed by

\[ v = \begin{bmatrix} \frac{(1 - \sigma)\alpha}{\sigma\alpha + (1 - \alpha)} & \frac{1 - \alpha}{(1 - \beta)(\sigma\alpha + (1 - \alpha))} \\ 0 & \frac{\beta}{(1 - \beta)} \end{bmatrix}. \]

By substituting (6.18) into (6.17), we also have

\[ c_i = ((U + 1 - i)/(U + 1))c_0. \]

Then, by substituting (6.1), (6.4), (6.23) and (6.25) into (6.20), we rewrite (6.20) as

\[ t_0 = \frac{M - 0.5(U + 2)c_0}{\bar{w} \left[ \bar{e}_2 + \sum_{k=1}^{K} \mathcal{H}^{k-1} (g + g \hat{h} + \sum_{i=3}^{L} g \hat{h} h^{i-2}) \bar{e}_4 + \sum_{k=1}^{K} \mathcal{H}^k \bar{e}_2 + \mathcal{H}^K v \bar{e}_2 \right]} = \mathcal{G}(c_0). \]

Finally by using (6.19) and (6.26), we can obtain \( \mathcal{F}(c_0) \) as

\[ \mathcal{F}(c_0) = \mathcal{G}(c_0) \bar{w} \mathcal{H}^K \bar{e}_2 - \Lambda(\mathcal{G}(c_0), c_0) = 0. \]

Some remarks on \( \mathcal{F}(c_0) \) can be given as follows:
1. The solution(s) of (6.27), which is often called equilibrium point [20][107], can be numerically found for $0 \leq c_0 \leq M$. Hereafter, we denote the equilibrium point(s) by $c_0^\ast$.

2. The system stability can be guaranteed, when (6.27) has a unique $c_0^\ast$ for $0 \leq c_0 \leq M$. If (6.27) has multiple solutions, the system is said to be unstable (including bistable). In order to estimate the bistability region of this system in terms of $U$ or other system parameters, $F(c_0)$ should be at least thrice differentiable [108]. For our system, it is hard to find either a closed form solution of (6.27), or to find such bistability regions due to the recursive form and non-differentiability in (6.12).

3. When $M$ is increased, $F(c_0)$ has multiple equilibrium points, which implies that the random access channel gets congested. Then, the backlog process oscillates between some local stable equilibrium points so that the throughput and delay performances at locally stable equilibrium points can be only obtained for some finite time period. Then, EPA analysis shows discrepancies against simulation, because the simulation results are time-averaged values, while EPA analysis represents the performance at an arbitrary equilibrium point among multiple points. This phenomenon has been observed in [20][107].

4. For the multiple equilibrium points found by $F(c_0)$, sometimes it is difficult to confirm these points by simulation, because we cannot know prior how long the backlog process stays at each one of multiple equilibrium points in a simulation (See also [109]). In such cases, simulation run-time should be long enough to reflect this behaviour of the backlog process. The first exist time (FET) for a backlog process to become unstable starting from a stable region has been studied in S-ALOHA and PRMA systems [110][111]. However, it is only possible to obtain the FET when a
multidimensional Markov chain is used for describing $M$ terminals’ states.

In order to obtain the system performance metrics, we denote the throughput of the random access channel and the traffic channel by $\zeta_a$ and $\zeta_d$. With $c_0^*$ numerically obtained, we can respectively obtain $\zeta_a$ and $\zeta_d$ by

$$\zeta_a = \Lambda(r, c_0^*) \quad \text{and} \quad \zeta_d = \bar{t}\bar{e}_2. \quad (6.28)$$

Denote the random access delay by $\bar{d}$ (slots). Using Little’s result [20], one can get $\bar{d}$ by

$$\bar{d} = c/\zeta_a = 0.5(U + 2)c_0^*/\zeta_a \quad (6.29)$$

where $c$ denotes total number of backlogged terminals, i.e., $c = \sum_{i=0}^{U} c_i$. Additionally, we denote by $P_B$ the reservation blocking probability, i.e., random access failure due to lack of traffic channels. We can obtain $P_B$ by

$$P_B = (\bar{\Lambda} - \zeta_a) / \bar{\Lambda} \quad (6.30)$$

where $\bar{\Lambda}$ is the mean number of RAPs successfully transmitted. This can be obtained by

$$\bar{\Lambda} = \sum_{k=1}^{\infty} \sum_{z=1}^{k} z \cdot \Gamma(z|k, P) (c_0^*)^k / k! \cdot \exp^{-c_0^*}. \quad (6.31)$$

### 6.2.2 Packet Dropping Probability

In what follows, we assume that the time scale of random access delay is small compared to that of the VoIP source so that VoIP source would not go back to the OFF state during the $\text{CON}_i$ state. Due to A1, incoming packets are stored without packet dropping occurring due to a full buffer. However, front-end packet dropping occurs when the waiting time
of a packet at the front-end of the buffer exceeds a delay constraint \( \tau \), the probability of which is denoted by \( P_d \). Recall that random access is initiated at the state transition from the OFF- to the ON-state with one packet arrival. As more packets may arrive before a traffic channel is successfully reserved, the first few packets stored in the queue may be successively dropped if their waiting time exceeds \( \tau \) before random access is successful, resulting in front-end clipping of a talk spurt. Denote by \( W \) the random variable of random access time (or delay) that the first packet at the beginning of a talk spurt in ongoing contention would experience. Additionally, the random variables of the remaining random access time of the first packet experienced by an arriving packet and that of the number of packets generated during the remaining random access time are denoted by \( Y \) and \( J \), respectively. Then, we can write \( P_d \) as

\[
P_d = \sum_{j=0}^{\infty} \frac{1}{1+j} \left[ 1(W > \tau) \cdot \Pr[W > \tau] \Pr[J = j] + \sum_{i=1}^{j} 1(Y + i \cdot L > \tau) \cdot \Pr[Y + i \cdot L > \tau, J = j] \right],
\]

in which indication function \( 1(x) \) is one if the condition \( x \) is satisfied, and zero otherwise. In (6.32) the first term depicts the dropping condition of the first packet, when its random access time exceeds \( \tau \). The second term denotes the mean number of dropped packets in the front of the queue, when the transmission time of each of \( J \) packets takes exactly \( L \) slots by SPS operation. Dividing the sum of these two terms by \( 1+j \) yields the packet dropping probability given \( j + 1 \) arrivals including the first packet. In order to obtain (6.32), we first consider the PMF of \( W \) in (6.32). To this end, we consider an absorbing Markov chain which consists of an absorbing state \( A \) and the \( i \)-th backoff state, \( b_i \) for \( 0 \leq i \leq U \). If the absorbing state represents the successful random access, the random access time is then described by the absorption time that a Markov process will be absorbed into the
absorbing state, starting from one of the backoff states, \( b_i \) for \( 0 \leq i \leq U \). Denote by \( Q \) the state transition matrix for this Markov chain, in which the \( i \)-th row and the \( j \)-th column element of \( Q \) for \( 0 \leq i \) and \( j \leq U \) is denoted by \( q_{ij} \), i.e., the state transition probability from \( b_i \) to \( b_j \). We write \( q_{ij} \) as

\[
q_{ij} = \begin{cases} 
(1 - P_s)/(U + 1) & \text{if } i = 0 \text{ and } j = 0, \ldots, U \\
1 & \text{else if } i = j - 1 \text{ for } 0 \leq i \leq U \\
0 & \text{otherwise},
\end{cases}
\]

in which the random access success probability is obtained by \( P_s = \tau_a/c_0^* \). We then write \( \Pr[W = k] \) as

\[
\Pr[W = k] = \bar{\xi}^{(0)} Q^{k-1} \bar{v}^T \text{ for } k = 1, 2, \ldots 
\tag{6.33}
\]

in which two row vectors of length \( U + 1 \), \( \bar{\xi}^{(0)} \) and \( \bar{v}^T \), are respectively expressed by \( \bar{\xi}^{(0)} = [\xi^{(0)}_i] \) for \( 0 \leq i \leq U \) and \( v = [P_s \ 0 \ 0 \ldots 0] \). Since the backoff randomly starts from one of \( U + 1 \) states when a packet arrives while the terminal has not reserved a traffic channel, we have \( \xi^{(0)}_i = 1/(1 + U) \). Note that (6.33) can be verified also by (6.29), i.e., \( \bar{d} = \sum_{k=0}^\infty k \Pr[W = k] \). Denote by \( X \) the random access time experienced by an arriving packet during ongoing random access of the first packet. Based on the residual life time theory of \( M/G/1 \) queuing system in [112], the PMF of this specific random access time can be determined by

\[
\Pr[X = x] = \left(x/\bar{d}\right) \Pr[W = x].
\tag{6.34}
\]
Since $Y$ is uniformly distributed over $X$, we have

$$\Pr[Y = y | X = x] = \frac{1}{x} \quad \text{for} \quad 1 \leq y \leq x. \quad (6.35)$$

Removing the conditioning in (6.35) by applying (6.34), we have

$$\Pr[Y = y] = \sum_{x=y+1}^{\infty} \Pr[Y = y | X = x] \Pr[X = x] = \frac{1}{d} \left(1 - \sum_{x=0}^{y} \Pr[W = x]\right) \quad (6.36)$$

where we have $\Pr[W = 0] = 0$. We can rewrite (6.32) as

$$P_d = \sum_{j=0}^{\infty} \frac{1}{j + 1} \left(\sum_{w=\tau+1}^{\infty} \Pr[W = w] + \sum_{i=1}^{j} \Pr[Y > \max(\tau - i \cdot L, 0)|J = j]\right) \Pr[J = j], \quad (6.37)$$

in which (6.36) is applied. Since packets are generated based on Bernoulli trials with probability $\sigma$ during the residual random access time of the first packet, we can obtain $\Pr[J = j]$ by

$$\Pr[J = j] = \sum_{y=1}^{\infty} \binom{y}{j} \sigma^j (1 - \sigma)^{y-j} \Pr[Y = y]. \quad (6.38)$$

### 6.3 Numerical Results and Discussions

In order to verify our analysis, we built a simulation model in MATLAB, in which one slot corresponds to 1 msec. We run each simulation for 150000 slots and time-average the results, with $L = 20$, $\tau = 50$, and unlimited $R_L$, unless otherwise stated. At the beginning of a simulation we randomly distributed $M$ terminals over the ON-, the OFF-, and the CON$\ell$ states for $0 \leq \ell \leq U$. The parameters of VoIP traffic are given in Section 6.1.2.
Note that VoIP capacity for SPS with initial random access is defined as the maximum of allowable terminals $M$ without exceeding a predefined packet dropping rate. In the following examples, we consider one PRACH in a slot, which takes six radio RBs. Note that 18 terminals can typically use one PUCCH of one radio RB in SPS with SRI. Thus, 108 terminals can be supported by six RBs in SPS with SRI.

**Bistability:** We first examine bistability of this system represented by (6.27). We denote the PMFs of $c_0$ and $c$ in the system by $\pi_{c_0}$ for $0 \leq c_0 \leq M$, which is obtained by simulations. In Fig. 6.3 we depict respectively $\mathcal{F}(c_0)$ and two examples of $\pi_{c_0}$ for $P = 2$, $D = 5$, $U = 10$, $K = 2$, $M = 100$ and $L = 20$ obtained from two simulation runs. As $\mathcal{F}(c_0)$ shows three roots of (6.27), we can observe two different distributions of $\pi_{c_0}$, which implies bistability. As previously mentioned, we can obtain the system performances at locally stable equilibrium points for some finite time period in a simulation. To mitigate such a bistability, we consider three schemes. First, we apply a retry limit $R_L$ in simulations, which limits the number of retransmissions. If the number of retransmissions exceeds $R_L$, the terminal drops the packet and gets back to the ON-state. We denote by $P_r$ the
Table 6.2: Stabilized system performance with various parameters

<table>
<thead>
<tr>
<th></th>
<th>$R_L = 5, U = 10$</th>
<th>$R_L = \infty, U = 15$</th>
<th>$R_L = \infty, U = 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P = 2, K = 2$</td>
<td>$P = 2, K = 2$</td>
<td>$P = 5, K = 2$</td>
<td></td>
</tr>
<tr>
<td>$\zeta_d$ (packets/slot)</td>
<td>1.38</td>
<td>1.38 (Ana: 1.43)</td>
<td>1.41 (Ana: 1.44)</td>
</tr>
<tr>
<td>$d$ (slots)</td>
<td>6.92</td>
<td>9.87 (Ana: 9.5)</td>
<td>6.45 (Ana: 6.25)</td>
</tr>
<tr>
<td>$P_d \times 100(%)$</td>
<td>0.24</td>
<td>0.9 (Ana: 0.7)</td>
<td>0.11 (Ana: 0.1)</td>
</tr>
<tr>
<td>$P_r \times 100(%)$</td>
<td>0.50</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

probability that a packet is dropped due to this retry limit, which is given in the first column of Table 6.2. Secondly we control $U$, and thirdly increase $P$. The corresponding results are given in the second and third columns of Table 6.2. While increasing $U$ or reducing $R_L$ also stabilize the system by dropping the packet forcibly, $P_d$ and/or $P_r$ would be sacrificed. Therefore we can conclude that increasing $P$ from 2 to 5 is the most effective way to mitigate instability, by relieving congestion in the random access channel. Thus, a sufficient number of RAPs should be secured according to the number of VoIP traffic users, and then $R_L$ can be further applied to prevent bistability. On the other hand, a large $U$ might not be desirable for delay-constrained applications, since it causes large random access delays. Note that the system shown in Fig. 6.3 is bistable with 100 terminals. However, it can be seen that 100 terminals can be supported with $\tau = 50$ and 1% packet dropping rate in the two stabilized systems with $U = 15$, or with $P = 5$ shown in the second and third columns of Table 6.2.

Comparison with PRMA: In Fig. 6.4, we compare the packet dropping rate, i.e., $P_d \times 100(\%)$, of SPS with random access ($P = D = 2$) against that of PRMA with $U = 5$ and $K$. Note that the original PRMA [107] uses a permission probability as the backoff algorithm, while the PRMA presented in Fig. 6.4 uses the UB algorithm. As expected, the VoIP capacity in SPS with random access is almost doubled under a 1% packet dropping rate due to increased $P$ and $D$. The discrepancies between analysis and simulation at $M = 26$ in PRMA and $M = 50$ in SPS with random access can be explained by the fact that the
systems have multiple equilibrium points with these population sizes. Thus, the simulation results are time-averaged values in bistable systems. While the VoIP capacity of PRMA depends only on a single parameter (permission probability in [107] or $U$ here), that of SPS with random access depends on the parameters $U$, $P$, $D$ and $K$.

The effect of $K$: Recall that $K$ is used for detecting a silence period. We examine the effects of $K$ on the system performance with $U = 10$, and $P = D = 5$ in Figs. 6.5(a)-6.5(d). In Fig. 6.5(a) we can observe that $\zeta_d$ slightly decreases due to longer holding times of traffic channels as $K$ increases. While $\zeta_d$ shows good agreements between analysis and simulation results, this is not the case with the packet dropping rate and $\overline{d}$, particularly for larger $M$ and $K$ in Figs. 6.5(b) and 6.5(c). Although the systems with those parameters are stable, EPA cannot evaluate system performance reliably. This is possibly caused by two reasons. First, by examining the PMF of the random access time, $W$, for $M = 115$ and $M = 125$ in Fig. 6.5(d), we can see that analysis and simulation results show some agreement for $K = 1$, while high fluctuations and disagreements are found either for $K = 8$ or for $M = 125$. When we increase $K$, the number of backlogged terminals
decreases as small as \( c^*_0 < 0.1 \) in analysis and simulation. Although the difference of \( c^*_0 \) between analysis and simulation (slightly underestimated by analysis) is quite small, the percentage of such differences is no longer negligible. In these cases, simulation results show a longer tail distribution of \( W \) than analytical results, which might cause mismatch between analytical and simulation results in Fig. 6.5(b). Secondly, even for a stable system, if the distribution of \( \pi_{c_0} \) (or \( \pi_c \)) is asymmetric, then it might result in mismatch between analysis and simulation results because \( c^*_0 \) in (6.27) represents only an averaged value for the equilibrium condition. In order to find \( K \) properly, we suggest the following cost function:

\[
\max_K \quad \gamma_1 \tilde{\zeta}_d(K) - \gamma_2 P_d(K, \tau)
\]

where \( \gamma_1 \) and \( \gamma_2 \) are weighting factors, and \( \tilde{\zeta}_d \) is the normalized traffic channel throughput, i.e., \( \tilde{\zeta}_d(K) = \zeta_d(K)/D \). In this example, \( K = 2 \) might be favourable, because \( \zeta_d \) gets lower with larger \( K \). However, the weight in (6.39) should be considered depending on QoS requirements.

The effect of window size \( U \): With \( P = D = 5 \) and \( K = 1 \) we vary \( U \) from 5 to 15 and use \( \tau = 40 \) and 50 in Figs. 6.6(a)-6.6(b). Fig. 6.6(a) depicts \( \zeta_d \), while the packet dropping rate is presented in Fig. 6.6(b). Smaller \( U \) (faster access) provides much lower \( P_d \), while \( \zeta_d \) is insensitive to \( U \) except \( U = 5 \) at \( M = 115 \) (highly congested). However, \( U = 5 \) causes the instability of the random access system for the system around \( M = 115 \) due to more frequent retransmissions which leads to congestion in the random access channels.

As VoIP capacity with \( \tau = 50 \), we can observe that maximally 130 terminals is acceptable with \( U = 15 \), while \( U = 10 \) can accept 130 VoIP terminals satisfactorily. We also observe \( P_d \) with a smaller \( \tau = 40 \) (slots), i.e., tighter delay constraint, which shows lower quality of VoIP traffic. Although some simplifications have been made in our model, we can see that 70 terminals can be admitted. In addition, we simulate the system with \( P = 64, \tau = 50, \)
Figure 6.5: The effect of $K$ on the system performance.
Chapter 6. Semi-Persistent Scheduling in LTE

U = 10, i.e., total number of available RAPs in LTE. We present simulation result only for this system due to computational complexity in (6.12) for such large $P$. Even though the packet dropping rate in the system with $P = 64$ and $U = 10$ is improved compared to the system with $P = 5$ and $U = 10$ (the same backoff window size), it is noticeable that the system with $P = 5$ and $U = 5$ shows compatible low packet dropping rate until $M = 115$. This implies that, rather than just increasing $P$, more dynamic control of $U$ can lead to large improvements of the packet dropping performance.

6.4 Summary

In this chapter we have examined the feasibility of SPS with initial random access for VoIP traffic in LTE by constructing an approximate model based on EPA to analyze its performance. In particular we have shown the VoIP capacity as the number of VoIP terminals that can be accommodated without exceeding some packet dropping probability,
when initial random access is used. From our study, it can be conjectured that SPS with initial random access might be alternatively used, if the cost of PUCCH for SRI not used in the OFF period of an application would be expensive. We have also investigated the effect of $K$, the number of empty transmission slots before the reserved traffic channel is released, and the effect of $U$ on VoIP capacity. Although our analytical model cannot provide reliable performance evaluations for large $K$, VoIP capacity have been estimated with various $U$ and $K = 1$. We have shown also the instability of SPS with random access based on UB algorithm (without retry limit), and discussed how to stabilize the system. We can conclude that, while a retry limit can effectively stabilize the system at the expense of a reasonable $P_r$, optimal control on $U$ would be more desirable.
Chapter 7

Conclusions and Future Work

This chapter summarizes the results and highlights the contributions of this dissertation. The direction of future research is also given.

7.1 Research Contributions

This thesis has proposed novel CRAs in an MPR S-ALOHA and examined the performance of existing CRAs with and without the PR scheme in MPR S-ALOHA for single- or multi-buffered terminals. Further as a reservation protocol, the performance of SPS with initial random access in LTE has been also examined. Particularly, we have characterized the performance of UB and BEB algorithms for single- or multi-buffered terminals in Chapters 3-5 with respect to the following three aspects. First, we have evaluated the performance of these systems based on i.i.d. assumption on backoff or queueing processes and showed the correlation coefficient of these processes from simulation. This has shown that, due to retry limit, the positive correlation among these processes is so weak that the performance evaluation based on i.i.d. assumption can agree well with simulation. Secondly, compared to S-ALOHA systems without retry limit [27][29][81], due to packet loss by retry limit, the stable capacity of S-ALOHA systems with retry limit should be understood by the maximum acceptable arrival rate without exceeding a predefined packet loss probability. Finally, the variance of access delay can be a metric to show access fairness in UB and BEB algorithms. The smaller variance the algorithms show, the more fair access we can
Chapter 7. Conclusions and Future Work

The results in each chapter are summarized as follows.

- In Chapter 2, two cross layer CRAs have been proposed in CDMA-based MPR S-ALOHA systems. When terminals are required to meet their transmission power via open loop power control, the proposed algorithms construct the system backlog information by measuring the power of MAI. Then, in the proposed algorithms, retransmission probability for a backlogged terminal is chosen slot-by-slot to maximize the conditional expected system throughput. For the IFT and DFT protocols, the performance of the algorithms under perfect power control is close to that predicted by the analytical performance model without estimation errors. For uncoded/coded systems under imperfect power control, system throughput decreases due to reduction of channel capacity. Specifically, the throughput efficiency in coded systems is maintained over 50% up to power control error of 2 dB. By examining FET and negative drift, the algorithms are shown to be stable.

- In Chapter 3, two existing CRAs, i.e., UB and BEB algorithms with retry limit, have been examined for the RACHs in LTE and IEEE 802.16 systems, respectively. Their performances are presented in terms of throughput, mean, and variance of packet retransmission delay and packet dropping probability, and the stability properties of both algorithms are examined. It is shown that stability is highly dependent upon the uniform window size and the maximum backoff stage and that the systems employing these algorithms are either stable or unstable, but do not show bistability behaviour. In addition, by adjusting the parameters of each backoff algorithm properly, it is found that the two algorithms show virtually identical throughput, but different delay variances. It is also shown that controlling the persistence probability provides effective access priority control with respect to various performance metrics. Finally,
a dynamic window assignment algorithm is proposed, which is a simple extension of an existing Bayesian broadcasting algorithm. The dynamic window assignment algorithm outperforms the existing backoff algorithms in any traffic condition as long as the perfect orthogonality assumption among random access code holds.

• Chapter 4 has examined the PR scheme with UB algorithm in LTE in conjunction with initial power transmission selection based on some PMFs. It has been assumed that uplink pathloss estimation is perfect. As in Chapter 4, it is also observed that a weak positive correlation over each terminal’s backoff process results from the retry limit. By examining the throughput gain per average transmission power consumption, it is shown that the PR scheme is effective when the SINR threshold is less than $-2.5$ dB. It is found that the PR scheme improves throughput, delay, and capacity of RACH (number of terminals without violating $P_d < 0.01$) at the cost of the average transmission power. In probabilistic initial transmission power selection schemes, $\tau$ is insensitive to the PMFs of selecting initial transmission power. Further, increasing the probability of selecting the highest power can increase $P_d$, while reducing $\bar{d}$. Therefore, the PR scheme of transmitting the lowest power first is the most efficient one with respect to $P_d$ and $\bar{p}$. In the presence of PR scheme, it is also needed to control $U$ dynamically in order to maximize $\tau$ according to the number of backlogged terminals. To this end, we proposed TAUBA, which estimates backlog size based on waiting-time counter by terminals and takes throughput gain by PR into account. By simulation, we showed that TAUBA improved the system throughput close to its maximum and minimized mean access delay and packet dropping.

• Chapter 5 has examined the queuing performance of a multichannel S-ALOHA system employing the UB algorithm with retry limit for contention resolution, which reasonably models the random access channels in LTE. It is shown that good agree-
ments between the analysis based on i.i.d. assumption and simulation come from weak positive correlation over each terminal’s queueing process due to retry limit. The analytical model considers buffered terminals with an MMBP traffic source that closely models the uplink traffic of Web-uplink transmissions of small IP packets, if the random access channel is unsaturated. It is shown that, once the system is saturated, the terminals’ queuing performance is severely degraded. The use of multichannel random access can effectively reduce such performance degradation as well as improve the system throughput. Further, compared to the more complex MMBP traffic model, this study has shown that Poisson arrivals can be used for a conservative performance evaluation on single- and multi-buffered terminals. When testing the UB algorithm, it is observed that increasing the retry limit provides a higher throughput and better queuing performance under light traffic load, while increasing the window size provides a higher throughput in a saturated channel. However, it also results in a higher packet loss rate. Users’ QoS requirements, e.g., for Web-browsing, may not be able to tolerate a high packet loss rate and thus, should be taken into account when such traffic is sent over the random access channels. Finally, it is found that the optimal window size in the literature, based on perfect knowledge of the number of backlogged terminals, is not optimal for buffered terminals with retry limit.

- Chapter 6 has explored the feasibility of SPS with initial random access for VoIP traffic in UMTS-LTE by constructing an approximate performance model based on EPA, and presented the MAC layer performances of the SPS. The VoIP capacity (provided by SPS with initial random access) is presented as the number of VoIP terminals that can be accommodated without exceeding some packet dropping probability. From this study, it can be recommended that SPS with initial random access
might be alternatively used, if the cost of PUCCH for SRI not used in the OFF period of an application would be expensive. The effect of $K$ (the number of empty transmission slots before the reserved traffic channel is released) on VoIP capacity is examined. From the numerical studies, it is shown that, while a large $K$ causes disagreements between analysis and simulations, it also results in high fluctuations in $P_d$. Based on the trade-off between $\zeta_d$ and $P_d$, it is determined that $K = 2$ is the optimal value in the given examples. This chapter has also examined the instability of SPS with random access based on UB algorithm (without retry limit), and how to stabilize the system. It is concluded that a retry limit can stabilize the system at the expense of a reasonable $P_r$, when the availability of 64 RAPs in the system is considered.

### 7.2 Suggestions for Future Work

1. **Performance Modeling of Slotted $p$-Persistent CSMA systems**
   By using an embedded Markov chain, this study proposes another performance model of slotted $p$-persistent CSMA systems for single-buffered terminals which have been examined by Takagi in 1983, Meditch in 1983, and Shoraby in 1987, respectively. This study is expected to show stability condition of slotted CSMA systems.

2. **Random Access Control for Multi-buffered terminals**
   In Chapter 5, we have shown that an optimal retransmission control for bufferless terminal cannot be used for buffered terminal. This study explores an optimal retransmission scheme for (unsaturated) buffered terminals.

3. **Stable Throughput Region of Secondary Users in Cognitive Radios**
   In Cognitive radios, secondary users opportunistically utilize the radio spectrums unused
by primary users. The throughput of secondary users can be affected by their own contentions on unused spectrum as well as by primary users’ behaviour. This work is to examine the stable throughput region of secondary users given a Markovian behaviour of primary users.

4. **Efficient Scheduling Scheme of Energy Harvesting Relay Network** This work considers a relay node which stores energy recharged by surrounding environments in its energy queue. This relay node transmits information provided by sensors or machines over time-varying wireless channels. In this work, an energy efficient scheduling algorithm would be developed in order to minimize energy consumption and meet delay constraint of information.
Bibliography


Bibliography


[73] *Universal Mobile Telecommunications System (UMTS); Selection procedures for the choice of radio transmission technologies of the UMTS (UMTS 30.03 version 3.1.0), TR 101 112 V3.1.0*, Nov. 1997.


