ABSTRACT

QOS GUARANTEES IN WIRELESS/MOBILE NETWORKS

by
Sunghyun Choi

Chair: Kang G. Shin

Recent years have witnessed a tremendous growth of research and development to provide mobile users a means of "seamless" communication through wireless media. This dissertation examines how to provide diverse Quality-of-Service (QoS) to mobile users generating/receiving heterogeneous traffic. It addresses QoS at two different levels: (1) packet-level QoS such as packet delivery delay, throughput, and error performance; and (2) connection-level QoS associated with connection setup and management.

To provide packet-level QoS, we develop a unified architecture for wireless LANs. We first define a polling-based medium access control (MAC) protocol for uplink accesses. Then, we address resource reservation, packet scheduling, connection-admission control, and how to handle location-dependent channel errors. With the proposed protocols, the delivery delays of real-time packets are bounded at the cost of some packet losses depending on the channel condition while allowing mobiles to have loss-free and fair accesses for transmitting non-real-time traffic.

We also develop an uplink code-division multiple access (CDMA) system with various packet-level QoS provisioning for wide-area cellular networks. Based on a transmission-rate request MAC protocol, we address the issues of resource reservation, packet scheduling, and connection-admission control. To satisfy the pre-defined error-performance requirements for each traffic class, we use a concatenated RS/convolutional code, and develop a new scheme for allocating mobiles appropriate power levels.

For connection-level QoS, we propose predictive, adaptive bandwidth reservation for hand-offs and admission control for newly-requested connections so as to keep the hand-off dropping probability below a pre-specified target assuming that each connection requires

a certain amount of bandwidth. Our schemes utilize history-based mobility estimation to predict user mobility. For the purpose of comparison, we also consider five other existing schemes. Through detailed simulations of various scenarios, we show the superiority of the proposed schemes to the other schemes.

Finally, we propose a unified architecture for wireless bandwidth management utilizing the concept of adaptive QoS. The allocated throughput to each connection is adapted depending on time-varying channel conditions and user mobility, where a connection's acceptable throughput range as well as adaptation constraints are specified during the admission control phase. The BS allocates bandwidth to each connection so as to maximize the service provider's aggregate reward. The proposed adaptive and non-adaptive schemes are compared to demonstrate the advantages of the adaptive scheme.

QOS GUARANTEES IN WIRELESS/MOBILE NETWORKS

by

Sunghyun Choi

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
1999

Doctoral Committee:
Professor Kang G. Shin, Chair
Assistant Professor Sugih Jamin
Professor Wayne E. Stark
Assistant Professor Kimberly M. Wasserman

To my wife Yoonjung

and

my parents

ACKNOWLEDGEMENTS

My utmost thank should go to God for His guidance and love for me throughout my life up to this far.

I am indebted to many individuals for their care and support given to me during my doctoral studies. First of all, I would like to express my deep gratitude to Professor Kang G. Shin. As my thesis advisor, he has provided me constant encouragement, insightful comments, and invaluable suggestions, which benefited not only the completion of this thesis, but also my career in a long time to come. I really enjoyed working with him, and I recall that I was so lucky to be part of his group since he was so concerned about his students. I also would like to thank Professors Wayne Stark, Kim Wasserman, and Sugih Jamin for serving on my dissertation committee.

I am also grateful to the former and present members of the Real-Time Computing Laboratory, especially, Seungjae Han, Soek-kyu Kweon, Youngjoo Suh, Jennifer Rexford, Ashish Mehra, Khawar Zuberi, Tarek Abdelzaher, Wu-chang Feng, Anees Shaikh, Xi Zhang, Sunghwan Moon, Babu Pillai, Songkuk Kim, Daji Qiao, and Haining Wang, for their friendship, discussion, and advice. I am really pleased to be part of all these talented colleagues. I am also thankful to my colleagues in the Systems Division, especially, Seongjun Oh, Jehong Jong, Jongtae Lim, Euisu Park, Byung-jae Kwak, and Nah-oak Song for their friendship and enjoyable technical discussions.

Thanks also go to the Department of EECS, the US Department of Transportation, the Korean government, and the Korea Foundation for Advanced Studies (KFAS) for providing financial support during my Ph.D. program.

My special thanks should go to my parents since they always have loved me, believed in me, and encouraged me in my study. I am also grateful to my parents-in-law and sister-in-law for their encouragement and prayer. Especially, I indebted to my sister-in-law who helped my wife and me in many ways during the final stage of this study. My final acknowledgments go to my wife Yoonjung for her dedicated sacrifice, support, understanding, and encouragement. This dissertation could not be completed without her presence beside me.

TABLE OF CONTENTS

DEDICATION ii

ACKNOWLEDGEMENTS iii

LIST OF TABLES vii

LIST OF FIGURES viii

CHAPTERS

1 INTRODUCTION 1
1.1 QoS-Sensitive Wireless/Mobile Networks 1
1.2 Overview of Existing Approaches 4
1.2.1 Wireless LANs with QoS Provisioning 4
1.2.2 QoS-Supporting CDMA Systems 6
1.2.3 Bandwidth Reservation for Hand-Offs 7
1.2.4 Adaptive QoS via Adaptive Bandwidth Management 8
1.3 Main Contributions 9
1.4 Organization of the Dissertation 10

2 WIRELESS LAN WITH QOS GUARANTEES 12
2.1 Introduction 12
2.2 Assumptions and System Specifications 13
2.2.1 Network Specification 13
2.2.2 Dynamic Time-Division Duplexed Transmission 14
2.2.3 Location-Dependent Errors and Their Control 15
2.3 Protocol Description 16
2.3.1 Downlink vs. Uplink 16
2.3.2 Real-Time vs. Non-Real-Time 17
2.3.3 Slotted Channels 18
2.3.4 Transmission Request Access 19
2.3.5 Error-Handling Mechanisms 20
2.3.6 Real-Time Communication Service 21
2.3.7 Non-Real-Time Communication Service 23
2.4 Run-Time Scheduling and Admission Control 24
2.4.1 Real-Time Traffic Scheduling 24
2.4.2 Admission Test 28
2.4.3 Non-Real-Time Traffic Scheduling 31

2.5 Experimental Results and Discussion 35
2.5.1 Assumptions and System Specification 35
2.5.2 Interworking of Real-Time Connections with Non-Real-Time Traffic 37
2.5.3 Non-Real-Time Communication Service 41
2.6 Conclusion 44

3 A QOS-SUPPORTING CDMA SYSTEM 47
3.1 Introduction 47
3.2 Modeling and Specifications 48
3.2.1 MC-CDMA Transmitter 48
3.2.2 Fading Channel and Power Control 51
3.2.3 Class-I Traffic Modeling 52
3.3 Protocol Description 53
3.3.1 Contention-Based Transmission-Request Accesses 54
3.3.2 Collision-Free Piggyback Class-II Rate Requests 55
3.3.3 Class-I Connection Establishment 56
3.3.4 Class-II Message Transmission 57
3.4 Error Performance 57
3.4.1 Demodulator and Convolutional Decoder 58
3.4.2 Reed-Solomon and CRC Decoders 58
3.5 Run-Time Control Schemes 60
3.5.1 Power-Level Allocation 60
3.5.2 Admission Control of Class-I Connections 62
3.5.3 Class-II Transmission Permission Control 63
3.6 Numerical Results and Discussions 65
3.6.1 Error Performance and Target SIRs 65
3.6.2 Admission Region for Class-I Connections 67
3.6.3 Performance of Class-II Traffic Transmission 67
3.7 Conclusion 71

4 BANDWIDTH RESERVATION FOR HAND-OFFS 73
4.1 Introduction 73
4.2 System Model 74
4.2.1 Communication Architecture 75
4.2.2 Cell Indexing 76
4.3 Mobility Estimation 76
4.3.1 Hand-Off Estimation Functions 78
4.4 Predictive, Adaptive Bandwidth Reservation and Admission Control 80
4.4.1 Bandwidth Reservation without Path Information 80
4.4.2 Bandwidth Reservation with Path Information 83
4.4.3 Control of Mobility Estimation Time Window 84
4.4.4 Admission Control 86
4.5 A Scheme with Estimation of Incoming and Outgoing Hand-Offs . 88
4.5.1 Three State Probabilities 89
4.5.2 Admission Control 90
4.6 Per-Connection Bandwidth Reservation Schemes 91
4.6.1 Control **AG**: No Hand-Off Drop 91

4.6.2 **BHARG**: Per-Connection Reservation after Admission . . 92
4.6.3 **NCBF**: Per-Connection Reservation before Admission . . . 93
4.7 Comparative Performance Evaluation 94
4.7.1 Simulation Assumptions and Specifications 94
4.7.2 Static Reservation 97
4.7.3 Adaptive Reservation without Path Information 99
4.7.4 Comparison among Alternatives without Path Information . 102
4.7.5 Adaptive Reservation with Path Information 109
4.7.6 Time-Varying Traffic/Mobility Case 109
4.7.7 Comparison of **AG**, **BHARG**, and **NCBF** 113
4.7.8 Comparison of **AC3** and **NCBF** 117
4.7.9 Comparison of **AC3** and **NAG**: One-Dimensional Case . . 120
4.7.10 Comparison of **AC3** and **NAG**: Two-Dimensional Case . . 128
4.8 Conclusion 132

5 BANDWIDTH MANAGEMENT FOR ADAPTIVE QOS 134
5.1 Introduction 134
5.2 System Description 135
5.2.1 Adaptive Error Control 135
5.2.2 Adaptive Application and Reward/Revenue Models 136
5.3 Adaptive QoS via Bandwidth Management 138
5.3.1 Bandwidth-Adaptation Mechanism 138
5.3.2 Connection-Admission Control 140
5.3.3 Adaptation Rule 141
5.4 Comparative Performance Evaluation 144
5.4.1 Simulation Assumptions and Specifications 144
5.4.2 Simulation Results and Discussion 146
5.5 Conclusion 149

6 CONCLUSIONS AND FUTURE WORK 150
6.1 Research Contributions 150
6.2 Future Research Directions 151

BIBLIOGRAPHY 154

LIST OF TABLES

Table

1.1 The classification of heterogeneous traffic. 3

2.1 Comparison of real-time and non-real-time communication services in the proposed protocol. 16

2.2 Performance of real-time traffic when non-real-time uplink traffic arrives with $\lambda_u = 0.0005$: $K = 20$ and $T_{req} = 200$. 38

3.1 Traffic classification for the proposed CDMA system. 48

4.1 Summary of the proposed admission-control schemes. 88

4.2 Status in each cell at the end of simulations when the offered load is 300 and $R_{vo} = 1.0$ with high user mobility. 107

4.3 Status in each cell at the end of simulations when the offered load is 300, $R_{vo} = 1.0$, and all mobiles follow one direction with high mobility. 108

4.4 Summary of comparison among **AC3**, **NAG**, **AG**, **BHARG**, and **NCBF**. 131

LIST OF FIGURES

Figure

2.1 The hidden terminal problem. . . . 14
2.2 The structure of a transmission-request slot. . . . 19
2.3 Real-time communication slot structures with channel probing. . . . 22
2.4 A round-robin queue for non-real-time communications. . . . 32
2.5 An example of non-real-time transmissions. . . . 33
2.6 A pseudo-code to determine which queue to serve when the wireless link becomes free. . . . 34
2.7 Two-state Markov chain modeled channel. . . . 35
2.8 The admission region for real-time connections: $K = 20$ and $T_{req} = 200$. . . 37
2.9 Average uplink class II-A message-transmission delay vs. non-real-time offered load: for different real-time connection-arrival rates λ_{rt}, $K = 20$ and $T_{req} = 200$. . . . 39
2.10 Average request-access latency vs. non-real-time offered load: $\lambda_{rt} = 0.001$, $K = 20$, and $T_{req} = 200$. . . . 40
2.11 Average message-transmission delay vs. non-real-time offered load when downlink or uplink traffic exists: $K = 20$ and $T_{req} = 200$. . . . 42
2.12 Average message-transmission delay vs. non-real-time offered load when downlink and uplink traffic arrive with the same rate, i.e., $\lambda_d = \lambda_u$: $K = 20$ and $T_{req} = 200$. . . . 42
2.13 Throughput vs. non-real-time offered load: $K = 20$ and $T_{req} = 200$. . . . 43
2.14 Average uplink class II-A message transmission delay vs. non-real-time offered load: with different values of T_{req}: $K = 20$. . . . 45
2.15 Average uplink class II-A message-transmission delay vs. non-real-time offered load: for $T_{req} = 600$ and with different values of K. . . . 45
3.1 Block diagram for an MC-CDMA transmitter: (a) an m-rate transmitter module with RS/convolutional concatenated encoders and interleavers, where $R'_b = R_b(Kb)/(Kb - H)$; (b) the integrated transmitter, where $m + n + o \leq M$. 49
3.2 The uplink and downlink frames along the time-axis. . . . 53
3.3 A FIFO queue for retransmission and a round-robin queue. . . . 64
3.4 Error probabilities vs. average SIR $\overline{\gamma}_o$ for the RS/convolutional concatenated coding system. . . . 65
3.5 Maximum total throughput at peak traffic W_{max} vs. class-II traffic target SIR $\hat{\gamma}_o^{dt}$ with $BW_{dt} = 0.2C$. . . . 66
3.6 Admission regions for the pairs (K_{vo}, K_{vi}). . . . 67

3.7 Average request access delay vs. offered load with $BW_{dt} = 50$. 69
3.8 Average message transmission delay vs. offered load with $BW_{dt} = 50$. . . . 69
3.9 Average message transmission delay vs. offered load. 71
4.1 Communication architectures among the MSC and BSs: (a) star topology with the MSC at the center, and (b) fully-connected network. 75
4.2 Indexing of cells. 76
4.3 An example of periodic windows to obtain hand-off estimation functions with $N_{win\_days} = 2$. 79
4.4 An example of the footprint of the hand-off estimation function for $prev = 1$. 80
4.5 An example of calculating $p_h(C_{0,j} \rightarrow next)$ when $prev(C_{0,j}) = 1$ and $next = 4$ using the footprint of $F_{HOE}(t_o, prev(C_{0,j}), next', T_{soj})$. 82
4.6 A pseudocode of the algorithm to adjust T_{est} in each BS. 85
4.7 A two-dimensional cellular structure. 95
4.8 P_{CB} and P_{HD} vs. offered load: static reservation with $B_r = 10$ BUs. 98
4.9 P_{CB} and P_{HD} vs. offered load: **AC3**. 100
4.10 Average target reservation bandwidth B_r and average bandwidth used B_u vs. offered load: **AC3**. 101
4.11 T_{est} and B_r vs. time when the offered load is 300 and $R_{vo} = 1.0$ for high user mobility: **AC3**. 103
4.12 P_{HD} at cells $<5>$ and $<6>$ vs. time when the offered load is 300 and $R_{vo} = 1.0$ for high user mobility: **AC3**. 104
4.13 Comparison among **AC1**, **AC2**, and **AC3** using P_{CB} and P_{HD} vs. offered load for high user mobility. 105
4.14 Comparison among **AC1**, **AC2**, and **AC3** using average number of B_r calculations for an admission test vs. offered load. 106
4.15 Comparison between **AC3** and **AC3 w/ PI** using P_{CB} and P_{HD} vs. offered load. 110
4.16 Comparison between **AC3** and **AC3 w/ PI** using the average number of numerical operations and comparisons vs. offered load. 111
4.17 Simulation results for the time-varying case: (a) mobiles' average speed and offered load vs. time of day and (b) P_{CB} and P_{HD} vs. time of day. 112
4.18 Comparison of **AG**, **BHARG**, and **NCBF** using P_{CB} vs. offered load. . . 114
4.19 Comparison of **AG**, **BHARG**, and **NCBF** using the average B_r and B_u vs. offered load for $R_{vo} = 1.0$. 115
4.20 Comparison of **NCBF** and **AC3** using P_{CB} and P_{HD} vs. offered load. . . . 118
4.21 Comparison of **NCBF** and **AC3** using the average B_r and B_u vs. offered load for $R_{vo} = 1.0$. 119
4.22 P_{HD} vs. estimation time T_{est}: **NAG**. 121
4.23 P_{HD} vs. estimation time T_{est} for $C = 20$ and $L = 40$: **NAG**. 122
4.24 Comparison of **NAG** and **AC3** using P_{CB} and P_{HD} vs. offered load. 123
4.25 Comparison of **NAG** and **AC3** using the average utilized bandwidth B_u vs. offered load. 124
4.26 Comparison of **NAG** and **AC3** using the dependency on the value of N_{quad} for $R_{vo} = 1.0$. 126

4.27 Complexity comparison of **NAG** and **AC3** using the number of hand-off probability p_h calculations for an admission test for $R_{vo} = 1.0$. 127
4.28 Complexity comparison of **NAG** (with $N_{quad} = 10$) and **AC3** (with $N_{quad} = 1$) using the average numbers of numerical operations and comparisons for an admission decision for $R_{vo} = 1.0$. 127
4.29 P_{CB} and P_{HD} vs. estimation time T_{est} in the two-dimensional environment with $C = 50$ and $L = 100$: **NAG**. 128
4.30 Comparison of **NAG** and **AC3** using P_{CB} and P_{HD} vs. offered load for the two-dimensional case. 129
4.31 Comparison of **NAG** and **AC3** using the dependency on the value of N_{quad} for $R_{vo} = 1.0$ and the two-dimensional case. 130
4.32 Complexity comparison of **NAG** (with $N_{quad} = 2$) and **AC3** (with $N_{quad} = 1$) using the average numbers of numerical operations and comparisons for an admission decision for $R_{vo} = 1.0$ and the two-dimensional case. 131
5.1 The reward function $F(b_g)$, where $b_g = r_c w_a$, and $w_{max\ (min)} = b_{max\ (min)}/r_c$. 137
5.2 The pseudo-code of an algorithm to resolve a bandwidth demand conflict among connections via downgrade in cell i. 142
5.3 The pseudo-code of a process to penalize an on-going connection $C_{i,j}$ which initiated a downgrade action in cell i. 143
5.4 The pseudo-code of an algorithm to allocate the residual bandwidth to connections (i.e., upgrade) in cell i. 144
5.5 The pseudo-code of an algorithm to re-allocate bandwidths to connections in cell i. 145
5.6 P_{CB} and P_{HD} vs. offered load. 146
5.7 W_r, W_a, and R vs. offered load. 147
5.8 W_a and number of on-going connections vs. time when the offered load is 100. 148

CHAPTER 1

INTRODUCTION

1.1 QoS-Sensitive Wireless/Mobile Networks

In recent years, there has been a rapid growth of research and development to provide mobile users a means of "seamless" communication through wireless media. This has made it possible to implement and deploy the current cellular systems, PCS (Personal Communication Systems), wide-area mobile data radio systems, and commercial wireless LANs (WLANs) [43, 35, 93, 25, 27, 78, 7, 26]. Cell phones based on cellular systems and PCS are now commonplace: one can easily purchase and use them in her/his daily life. For example, in some countries including Korea and Japan, more than 30% of the population are carrying cell phones. Currently, most wide-area cellular systems are used for voice communications with some limited support of low-rate data services while most WLANs are used for conventional data traffic based on a best-effort service.

There has been a tremendous growth of the Internet in terms of size, user population, and the number of applications. Along with the fast development of high-speed-yet-low-cost network hardware, and the growing demand for the integrated services supporting broadband multimedia such as audio and video as well as the conventional data traffic such as e-mail, telnet, and WWW, a large number of researchers have been looking into communication services with guaranteed Quality-of-Service (QoS) such as delivery delay and link bandwidth in wired networks [80, 103, 106, 5, 42, 55, 90]. Mobile users would expect at least similar services through wireless devices while they are moving around, or without the headache/limitation of wiring. Part of this expectation will come true in the near future thanks to remarkable research and development efforts currently underway. In this dissertation, we first identify diverse wireless/mobile network-specific QoS issues, then develop possible solutions.

Unlike in wired networks which assume low error rates and stationary users, there are several factors in wireless/mobile networks which make it very difficult to provide QoS guarantees:

- Resources in wireless networks are scarcer than in wired networks. Wireless links, in general, provide much lower bandwidths than the wired counterpart. This disparity is expected to hold in future even though rapid progress is being made for high-speed wireless transmission, due mainly to the physical limitation of the wireless media.

- Wireless channels are inherently unreliable and prone to location-dependent, time-varying, and bursty errors due to noise, multipath fading, shadowing, and interferences. Even with channel coding, diversity-combining, and power control techniques, their unreliability is much higher than that of wired links.

- Users tend to move around during a communication session causing hand-offs between adjacent cells. The current trend in cellular networks to reduce cell size in order to accommodate more mobile users in a given area will make it even more difficult to deal with the mobility-related problems.

Due to these distinct characteristics of wireless/mobile networks, it is necessary to develop some mechanisms tailored to support QoS for mobile users. Specifically, we consider infrastructure-based "cellular networks", each of which is composed of a wired backbone network and a (possibly large) number of base stations (BSs) which cover the whole geographical area in service. The area covered by a BS is called a *cell*. A mobile,[1] while staying in a cell, communicates with its communication partner, which may be a node connected to the wired network or another mobile, through the BS in the same cell. When a mobile moves into an adjacent cell during a communication session, a *hand-off* will enable the mobile to maintain connectivity to the communication partner, i.e., the mobile will start to communicate through the new BS, hopefully without noticing any difference. Note that the downlink (BS-to-mobile) and uplink (mobile-to-BS) in a cell can respectively be thought of as the end-most and front-most links of a multi-hop connection. Since wireless links usually have much less bandwidth and much worse error characteristics than the wired counterpart, the former might become a bottleneck to the end-to-end communication performance.

Various QoS in wireless/mobile networks can be considered and achieved at two abstraction levels:

[1]We use the term "mobiles" to refer to mobile or portable devices, e.g., hand-held handsets or portable computers.

	Class I	Class II-A	Class II-B
Name	real-time	non-real-time	
Examples	voice & video	remote login	e-mail
Delay	bounded	sensitive	tolerable
Loss	loss-tolerant	zero-loss	

Table 1.1: The classification of heterogeneous traffic.

Packet-level QoS: like packet-delay bounds, throughput, and error performance. These issues have been studied extensively for wired networks, but only limited efforts have been reported for wireless networks. Resource reservation, packet scheduling, and connection-admission control as well as a proper medium access control (MAC) protocols are the key components of solutions to these QoS issues. Channel coding and power control are also essential to error performance in wireless networks. We will consider two types of wireless networks in which diverse packet-level QoS is provided.

Connection-level QoS: related to connection establishment and management, which are very important, especially in dealing with user mobility. For connection establishment, location management and routing are important issues [60, 101, 85, 82]. In QoS-sensitive cellular networks, how to re-route connections efficiently upon hand-offs is also very important [98, 53, 70]. One of the most important connection-level QoS issues is how to control (or reduce) hand-off drops due to the insufficient bandwidth available in the new cell a user moves to. We will consider two connection-level QoS parameters: the probability P_{CB} of blocking new connection requests and the probability P_{HD} of dropping hand-offs.

Depending on its distinct characteristics and packet-level QoS requirements, heterogeneous traffic is categorized into two classes as summarized in Table 1.1: (1) class-I real-time traffic such as voice and video requires bounded delays, but can usually tolerate some packet losses; and (2) class-II traffic like the conventional data services requires loss-free transmission, but requires no bounded delay. Class II can be divided further into two subclasses: (a) class II-A which is delay-sensitive like FTP and remote log-in; and (b) class II-B which is delay-insensitive like paging and e-mail. Class II-A is given priority over class II-B.

Wireless networks can be divided into mainly two types as wired networks: local-area network (LAN) and wide-area network (WAN). For many reasons including geographical cell size, allocated frequency band, channel characteristics, main target applications, and economy, LANs and WANs have been developed based on different physical and link layer protocols, respectively. Specifically, in terms of the MAC sub-layer, most commercial wireless LANs are based on random access MAC protocols targeting at non-real-time data communications while most cellular WANs are targeted at voice calls and use circuit-switching MAC protocols.

Various wireless MAC protocols can be categorized into two types, depending on how a wireless link is shared among different mobiles: time-division multiple access (TDMA) and code-division multiple access (CDMA). Conceptually, these two schemes can be briefly explained as follows: with TDMA-based protocols, time slots are defined, and different mobiles transmit their packets in different time slots. On the other hand, with CDMA-based protocols, large time slots or frames are defined, where the time duration of a frame corresponds to many TDMA time slots under the same conditions of wireless bandwidth and modulation schemes, and a whole frame is used by all the mobiles in a cell with different spreading codes. Note that we categorize all the schemes, which do not use any spread-spectrum modulation for the multiple access purpose, as a TDMA variant. For example, even CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance), which is not based on fixed-size time slots, is categorized into this type. Most wireless LANs are adopting TDMA variants since CDMA is not good for typical LAN environments as discussed later while both TDMA and CDMA variants have been adopted for WANs. In the following, some existing approaches to support diverse QoS in wireless/mobile networks are discussed.

1.2 Overview of Existing Approaches

1.2.1 Wireless LANs with QoS Provisioning

WLANs are emerging as an attractive alternative or complementary to wired LANs [78, 2], because they enable us to set up and reconfigure LANs easily without incurring the cost of wiring. They are characterized as high-speed wireless systems which cover relatively small geographical areas, as compared to other wireless systems such as cellular, PCS, and mobile data radio systems. WLANs are expected to be the solution to the problem of meeting the growing demand that mobile clients should have access to the existing high-speed wired networks.

Recently, there have been significant research efforts to support packet-level QoS guarantees in wireless (local-area) networks utilizing time-slot allocation-based MAC protocols. A wireless MAC protocol in [15] can be considered as a polling scheme using stop-and-go queueing for real-time traffic, and FIFO queueing for non-real-time traffic. However, this scheme can support only a finite number of delay bounds due to the inherent limitation of stop-and-go queueing. The authors of [12] dealt with the scheduling and admission control for a TDMA system to support variable bit rate (VBR) connections with different packet dropping probabilities. However, it deals with the uplink only, and connections are allowed to have only two different delay bounds. Both of the schemes do not provide any proper means to handle wireless channel errors. Remote-Queueing Multiple Access (RQMA) in [36] also addresses the problem of providing QoS to heterogeneous traffic using the existing packet scheduling algorithms proposed for wired networks. This scheme also handles channel errors via FEC/ARQ hybrid. However, how to handle location-dependent errors, which will be discussed below, was not specifically considered. An emerging WLAN MAC standard, IEEE 802.11, provides both real-time and non-real-time communication services in a WLAN using polling and CSMA/CA [28]. IEEE 802.11 does not specify the polling order, admission control, and how to handle channel errors all of which are essential for real-time communication.

QoS-supporting wireless systems have recently been studied in the context of wireless ATM. The distributed queueing request update multiple access (DQRUMA) is used for BAHAMA as a MAC protocol [34, 52]. DQRUMA is a demand-assignment access scheme, which has some limited support for real-time traffic due to its contention-based access of transmission requests. The MAC protocol for a wireless ATM in [13] uses a polling scheme. However, it deals with uplink access only. The Seamless Wireless ATM Network (SWAN) [1] also uses a polling scheme as a MAC protocol, but such details as polling scheduling and admission control were left unaddressed. WATMnet [88] uses a dynamic TDD/TDMA as the MAC to support heterogeneous traffic. The MAC protocol in WAND (MASCARA) [81] is another MAC protocol for a wireless ATM system (named WAND) using reservation and contention-based TDD/TDMA techniques. The authors of [61] proposed a wireless MAC protocol for wireless ATM, in which the access and scheduling procedures are decentralized. Basically, none of the above wireless ATM schemes deals with location-dependent and bursty errors explicitly. The authors of [77] provide a good survey on wideband local access schemes comparing both wireless LAN and wireless ATM technologies.

Recently, there have been research efforts on packet scheduling to handle location-dependent and bursty errors based on channel-condition prediction mechanisms and packet transmission deferments [6, 67, 37, 74, 86, 65]. Most of the work has focused on how to provide (long-term) fair access to a wireless link among connections in the presence of location-dependent errors [67, 37, 74, 86, 65]. Moreover, some of these proposals have limitations, e.g., the issue of expediting the retransmission of packets was not addressed [37, 67, 74]. The schemes in [67, 74, 65] attempt to compensate for the bandwidth of connections which experienced deferments by reducing the bandwidth of the other connections which received more bandwidth than promised before. The scheduling policy in [86] does not have any explicit concept of compensation, nor does it address how to bound packet delays either.

1.2.2 QoS-Supporting CDMA Systems

Code-Division Multiple-Access (CDMA) is emerging as a promising technique for future cellular wide-area wireless communication systems. As claimed by many researchers [38, 99], CDMA in cellular environments offers several advantages over other wireless access techniques such as high spectral efficiency, soft capacity, soft handoff, and increased system capacity. Most third-generation (3G) wireless systems, which will appear by early 2000s, adopted CDMA technology for wireless link access, e.g., WCDMA of Europe and Wideband cdmaOne of the US [29, 75].

To support multi-rate transmissions with DS-CDMA systems, many techniques have been proposed such as multi-modulation CDMA,[2] multi-processing gain CDMA, and multi-code CDMA [76, 48, 46]. Multi-code (MC) CDMA is known to have some advantages over other methods as claimed in the literature; the multi-modulation CDMA degrades the performance for the users with high data rates [76], and the multi-processing gain CDMA is expected to cause problems to users with a very high source rate to have too small a processing gain to maintain good cross correlation among different user codes [46]. MC-CDMA is expected to work well with multimedia traffic; when it integrates multimedia traffic, traffic streams with significantly different transmission rates can be easily integrated into a unified architecture, with all the transmission channels having the same bandwidth and spread spectrum processing gain.

One weakness of general CDMA techniques is that the data rate of an individual user can hardly be commensurate with the system bandwidth because of the spreading factor even with above-mentioned multi-rate CDMA techniques. In this regard, a carefully-developed Time-Division Multiple Access/Time-Division Duplexing (TDMA/TDD) system like those discussed in Section 1.2.1 may yield less delays for a small population of users generating

[2] This technique seems to be less known compared to the other two. Basically, it varies the modulation depending on the required maximum rate. For example, the modulation is selected among BPSK, QPSK, and 16-QAM to support a basic rate, up to two times the basic rate, and up to four times the basic rate, respectively [76].

very bursty traffic (e.g., a typical wireless LAN environment).

There have been some proposals to support different types of traffic in CDMA systems. The authors of [92] considered how to support voice and data in a CDMA system using the slotted ALOHA for data transmission. A demand-assignment access control scheme was proposed in [63] to support efficient multi-rate transmissions according to the mobile's demand. The authors of [39] proposed a receiver-oriented code channel and receiver scheme for data traffic using the ALOHA access. But, all of these have some form of deficiencies on their own: multi-rate is not supported [92, 39], or class-I traffic is not supported [63].

1.2.3 Bandwidth Reservation for Hand-Offs

As in QoS-guaranteed wired networks, mobile users would like to continue communication sessions even after a number of hand-offs. To this end, the BS in each cell should reserve a certain amount of bandwidth for hand-offs. The notion of bandwidth reservation for hand-offs was first introduced in the mid-80s [45]. In this scheme, a set of channels are permanently reserved for hand-offs. This reserved bandwidth is not allowed to be used for accepting new connections so that P_{HD} may be kept lower than P_{CB}. It was shown that this static reservation scheme is optimal in the sense of minimizing a linear objective function of the connection blocking probability and the hand-off dropping probability when both new and hand-off connection arrivals are Poisson, and connection durations are exponentially-distributed [87]. We will henceforth call this a *static* reservation scheme. As will be shown later in Chapter 4, this static reservation scheme cannot effectively handle a variety of connection bandwidths, traffic loads, and users' mobility.

The authors of [71] advocated the connection hand-off dropping probability as an important connection-level QoS parameter in wireless/mobile networks, and designed a distributed call admission-control scheme to keep the connection hand-off dropping probability below a specified limit. With their scheme, the BS obtains the required bandwidth for both the existing and hand-off connections after a certain time interval, then performs admission control so that the required bandwidth may not exceed the cell capacity. Their scheme was shown to be better than the static reservation scheme. The authors of [66] extended this scheme as a part of their proposal to accommodate heterogeneous connection bandwidths, and studied the effects of design parameters used in the scheme. The main problems of these schemes are: (1) they assumed the time duration a mobile spends in a cell before a hand-off is exponentially-distributed, which is impractical; and (2) there is no specified mechanism to predict which cells mobiles will move to.

The shadow cluster concept was suggested in [58] to estimate future resource requirements and perform admission control in order to limit the hand-off dropping probability, in which the shadow cluster is a set of cells around an active mobile. This scheme is based on the precise knowledge of each user mobility, depending on the location and time, which they assumed given. How to determine the shadow cluster is also not defined clearly. Moreover, their scheme is computationally too expensive to be practical. There were also limited efforts to estimate mobility in order to reserve bandwidth for hand-offs according to this mobility estimation. The authors of [66] explored mobility estimation for an indoor wireless system based on both mobile-specific and cell-specific observation histories. User-specific observation of mobility is expected to be costly and not accurate in general.

There also have been research efforts for adaptive bandwidth reservation. The author of [56] suggested bandwidth reservation depending on the existing connections in adjacent cells. However, the scheme lacks such details as how much of bandwidth should be reserved. The bandwidth-reservation and admission-control schemes in [95] assume that the mobility of users is predictable, that is, mobility can be characterized by the set of cells the mobile is expected to visit during the lifetime of the mobile's connection. This assumption, however, does not hold for most wireless/mobile networks. Moreover, the scheme reserves the required bandwidth at every cell and node in the mobility specification, which is usually excessive.

1.2.4 Adaptive QoS via Adaptive Bandwidth Management

Adaptive QoS in wireless/mobile networks has been proposed in order to manage scarce wireless resources efficiently in the dynamic environment with user mobility. By adaptive QoS, we imply that the QoS level is adapted over time depending on the fluctuation of available resources where the range of acceptable QoS levels is contracted during the admission control phase. Lee proposed a framework of QoS adaptation [56]. The authors of [9] proposed a selective dropping of real-time traffic in case of cell congestion as a part of their design of transport and network layer protocols for mobile systems. Filtering the communication protocol at proxy servers in a wired network was also reported in [102], which can be used as a QoS-adaptation method. Depending on the protocol location in the network hierarchy (i.e., above or below the socket layer), high and low level proxies were defined. In a stationary environment (e.g., without hand-offs), how the proposed filters work with HTTP, NFS, and TCP was discussed in terms of latency and throughput.

Bharghavan's Prayer project at University of Illinois also considers adaptive service [66, 64]. They are building an adaptive service model, adaptation algorithms, and a testbed for indoor mobile applications. Campbell's group at Columbia University also presented a platform called *mobiware* to support adaptive QoS using active filters located in the BSs [10, 59]. Filters are Java-coded, and dispatched from mobiles to a BS or between two BSs during the hand-off process. Using MPEG-coded video traffic, two types of filters (one to drop packets selectively, and the other to shape rate dynamically) were demonstrated in an ATM testbed in the presence of hand-offs in [4]. Badrinath's group at Rutgers also presented a fair rate adaptation rule which considers not just the fairness in the current cell, but also in many cells which the mobile traverses throughout its connection's life [96]. Basically, the existing schemes consider user mobility issues only, i.e., the bandwidth fluctuation due to hand-offs, connection setup and termination, even though the wireless link availability can fluctuate due to wireless channel conditions as well as user mobility.

1.3 Main Contributions

The objective of this dissertation is to provide diverse QoS to mobile users generating/receiving heterogeneous traffic in wireless/mobile networks. The main contributions of this dissertation are as follows:

- **A unified architecture of WLAN with QoS provisioning:** we address how to provide various packet-level QoS to heterogeneous traffic in a WLAN experiencing location-dependent, time-varying, and bursty channel errors. We propose (1) polling-based MAC protocol for uplink access, (2) scheduling both real-time and non-real-time packets under the dynamic TDD transmission, and (3) admission control of real-time connection requests.

- **A QoS-supporting uplink CDMA system:** an uplink MC-CDMA architecture is developed to provide diverse packet-level QoS to heterogeneous traffic in wide-area cellular systems. Addressed are (1) MAC protocol, (2) transmission-power allocation to meet error performance requirements, (3) admission control of real-time connection requests, and (4) non-real-time packet scheduling.

- **Predictive and adaptive bandwidth reservation for hand-offs:** to handle a connection-level QoS, we propose predictive, adaptive bandwidth reservation for hand-offs and admission control for new connections so as to keep the hand-off dropping probability below a pre-specified target. We develop a cell-specific history-based mobility estimation which is practically feasible. Based on the mobility estimation of users, the bandwidth to be reserved in each cell is determined adaptively.

- **A unified architecture of adaptive bandwidth management:** we develop a unified architecture for wireless bandwidth allocation and management. We identify the problem of the bandwidth adaptation triggered by time-varying wireless channels as well as user mobility. The BS allocates bandwidth to each connection adaptively so as to maximize the aggregate revenue while attempting to meet the adaptation constraints specified by each connection.

1.4 Organization of the Dissertation

The remainder of this dissertation is organized as follows. Chapter 2 considers a unified architecture of a wireless LAN with diverse packet-level QoS provisioning. We first develop a MAC protocol to handle both real-time and non-real-time traffic. Then, the run-time scheduling of real-time and non-real-time traffic, and the admission-control scheme for real-time connection requests are presented. The performance of the proposed protocol is evaluated and discussed through simulations.

Chapter 3 deals with an MC-CDMA uplink system to provide diverse packet-level QoS for heterogeneous traffic according to the application requirements. The communication protocols for both real-time and non-real-time traffic, and run-time control schemes of the BS including the allocation of a power level to uplink accesses, admission control of real-time connections, and scheduling of non-real-time packet-transmission requests are described. Specifically, the error-control schemes and the error performance analysis are presented. Based on both the analytical and simulation results, the performance of the proposed schemes is evaluated and discussed.

Chapter 4 presents bandwidth reservation for hand-offs and admission control for new requests in order to limit the hand-off dropping probability under a pre-specified target value. The users' mobility estimation based on an aggregate history of observations is first developed. Based on this mobility estimation, predictive, adaptive bandwidth reservation and three admission-control schemes are presented. Some existing and related schemes are also considered for a comparative purpose. Based on detailed simulations under various scenarios, we evaluate the performance of the proposed schemes, and compare it with that of others.

Chapter 5 deals with the problem of allocating the scarce wireless bandwidth to each connection adaptively in the dynamic environment with time-varying errors and user mobility. An adaptive bandwidth-allocation mechanism geared toward maximizing the total reward to the service provider is presented. Connection admission control and bandwidth reservation for hand-offs are re-defined from those in Chapter 4 utilizing this adaptive bandwidth-allocation. Using simulations, the performance of the adaptive scheme is evaluated, and compared with that of a non-adaptive scheme to show its superiority.

Chapter 6 summarizes the main contributions of this dissertation, and describe possible future research directions.

CHAPTER 2

WIRELESS LAN WITH QOS GUARANTEES

2.1 Introduction

Real-time communication service deals with the real-time traffic, in which the packet delivery delay is bounded at the cost of some packet losses depending on the time-varying channel condition, while non-real-time communication service is for non-real-time traffic. In this chapter, we consider a unified WLAN architecture, composed of a base station (BS) and a number of mobiles, which provides both real-time and non-real-time communication services. The integrated communication services considered here can be divided into four types: (1) downlink real-time; (2) uplink real-time; (3) downlink non-real-time; and (4) uplink non-real-time services, where the downlink (uplink) is for transmitting BS-to-mobile (mobile-to-BS) traffic. To guarantee the timely delivery of packets, real-time traffic is handled by a connection-oriented service, while non-real-time traffic is handled by a connection-less service. Dynamic Time Division Duplexed (D-TDD) transmission is used as in [15, 1, 88, 81, 36], and hence, the same frequency channel is time-shared for downlink and uplink transmissions under the dynamic access control of the BS.

Due to the different nature of uplink and downlink transmissions, they are treated differently. For the downlink, the BS schedules the transmission of those packets which had already arrived at the BS. On the other hand, the exact status of uplink packets in mobiles is not known to the BS. Thus, mobiles request the BS for permission to transmit their uplink packets, which, upon reception of the requests, schedules uplink transmission permissions. These requests are made on a *per-connection* basis for the real-time service, and on a *per-message* basis for the non-real-time service, where a message is composed of a number of packets. The BS can give a mobile a packet-transmission permission by *polling* the mobile. Polling is one of the most popular wireless access mechanisms when the BS exists [97, 104, 41, 44, 15, 13, 1]. Real-time traffic is given priority over non-real-time traffic for its timely delivery.

The WLAN system should provide (1) timely delivery of real-time packets while minimizing packet losses; (2) virtually error-free transmission of non-real-time packets; (3) good average delay and throughput performance of non-real-time traffic utilizing the bandwidth left unused by real-time traffic; (4) fair usage of a channel for non-real-time traffic among mobiles; and (5) low latencies for non-real-time packet transmissions and real-time connection setup, and in handling hand-off requests. We design a medium access control (MAC) protocol which can support these requirements, the scheduling of uplink and downlink packet transmissions, and the admission control of newly-requested real-time connections in a dynamic environment with location-dependent and bursty errors.

The chapter is organized as follows. Section 2.2 states the specification and assumptions of the wireless network under consideration. In Section 2.3, we describe the MAC protocol to support both real-time and non-real-time traffic. Section 2.4 presents the run-time scheduling of real-time and non-real-time traffic, and the admission-control scheme for real-time connection requests. The proposed protocol is evaluated in Section 2.5. Finally, the chapter concludes with Section 2.6.

2.2 Assumptions and System Specifications

Before presenting our protocols, we state the assumptions and specification of the WLAN under consideration.

2.2.1 Network Specification

The WLAN of our interest is based on an infrastructure that is composed of a wired backbone network and a (possibly large) number of base stations (BSs) which cover the whole geographic area in service. Each BS handles an area, called a *cell*, where all the mobiles in the cell communicate through the BS. In this system, the downlink and uplink in a cell can respectively be thought as the end-most and front-most links of a multi-hop end-to-end connection. Since wireless links usually have much less bandwidth than the wired counterpart, the former might become the bottleneck of the end-to-end communication performance. We will, in this chapter, focus on the uplink and downlink transmissions within a single cell.

Each mobile is assumed to transmit packets with some limited transmission power so

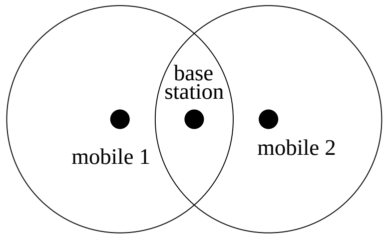


Figure 2.1: The hidden terminal problem.

as to reach the BS only, which is located at the "center" of the cell. So our system differs from peer-to-peer communication systems which do not require any infrastructure, e.g., ad-hoc networks. The lower the transmission power, the better, since (1) mobile devices operate with batteries; and (2) the transmitted signals in a cell are nothing but inter-cell interferences to other cells. In this sense, it is desirable to transmit packets with as low a power as possible, which may, unfortunately, cause the *hidden* terminal problem [2]. In Figure 2.1, the transmission ranges of mobiles 1 and 2 do not allow them to hear each other, but can both be heard by the BS in between. Mobiles 1 and 2 are hidden terminals to each other. So, in a cell, the uplink (mobile-to-BS) is not a broadcast channel while the downlink (BS-to-mobile) is. Hence, mobiles are assumed not to listen directly to one another even if some can, depending on their relative locations. When a mobile wants to send a packet, regardless whether the packet is destined for another mobile in the same cell or a mobile or static node outside the cell, it first sends the packet to its BS. The BS will then forward the packet to its final destination, sometimes via a wired backbone network, i.e., when the packet is destined for a node outside the cell. When the BS transmits packets through the broadcast downlink, all but the destination mobile in the cell ignore them. Note that only the BS can determine if a collision has occurred in the uplink channel. In summary, the system under consideration can be considered as a network with the *star topology* whose center is the BS.

2.2.2 Dynamic Time-Division Duplexed Transmission

Dynamic Time Division Duplexed (D-TDD) transmission is used in the system, i.e., a wireless channel over a frequency slot is time-shared for both downlink and uplink transmissions under the BS's dynamic access control. We could instead use (1) Frequency Division Duplexed (FDD) transmission as in most cellular systems, in which two different frequency channels are allocated for uplink and downlink transmissions; or (2) static TDD in which a portion (usually a half) of each time frame is allocated for the uplink and the remaining portion is for the downlink. Even though these static duplexing transmissions are simpler, and adopted in most currently available cellular systems, D-TDD is shown to offer more efficient link utilization in case of unbalanced and time-varying uplink and downlink traffic [14]. Considering the growing number of applications involved with unbalanced two-way traffic (e.g., web browsing), D-TDD is a very promising and important design choice.

2.2.3 Location-Dependent Errors and Their Control

It is well-known that a wireless channel is inherently unreliable due mainly to noises, interferences, fadings, etc. We use a *hybrid* of Forward Error Correction (FEC) and Automatic Repeat reQuest (ARQ). That is, the receiver tries to correct errors first, and if unsuccessful, a decoding failure is declared, implying the detection of a packet error. A selective-repeat ARQ is used, so the receiver sends an ACK or NAK for each received packet according to the decoding result, and only NAKed packets will be retransmitted by the sender. However, real-time packets can be retransmitted only for a limited time due to their timing constraints or deadlines, which should be met by appropriately scheduling packet transmissions. Wireless channels, in many cases, are known to experience location-dependent, time-varying, and bursty errors, switching back and forth between good and bad states over time [30, 94, 100, 31]. A *channel* is defined between each mobile and the BS, and hence, mobile i's channel represents the one between this mobile and the BS. Channel probe and transmission deferment are adopted to handle the above-mentioned types of errors, i.e., before transmitting each real-time packet over a channel, the channel condition is probed through a hand-shaking mechanism, and if the condition is bad, the transmission is deferred, and a different packet is scheduled for transmission. This in turn reduces the need of retransmissions significantly. Since the prediction of channel condition/state cannot be perfect, a packet transmission may fail. Erroneously-transmitted or deferred real-time packets must be scheduled for (re)-transmission before their deadline in a best-effort manner so that they won't compromise the other undeferred packets' deadline guarantees. We assume that certain channel coding is used for the receiver to detect and correct errors in packets it received. However, the MAC protocol should provide the means of channel probing and feedback from the receiver to the sender for possible retransmissions.

Throughout the rest of this chapter, we will ignore the packet-propagation delay, since it is usually small relative to the other delay components like queueing and transmission delays in a cell.[1] Finally, all packets, like ATM cells, are assumed to be of the same fixed size. So,

[1]A cell in this chapter refers to a *micro-cell*, which has coverage of a few hundred meters, or a *pico-cell*,

	Real-Time	Non-Real-Time
Connection	Connection-oriented	Connection-less
Request	Per-connection based	Per-measage based
Admission Test	Yes	No
Scheduling	Deadline-driven	Round-robin-variant

Table 2.1: Comparison of real-time and non-real-time communication services in the proposed protocol.

a long message is fragmented into a number of packets of identical size at the transmitter. With a selective-repeat ARQ scheme, fragmentation could be effective. Otherwise, a long message received in error could result in the retransmission of the entire message, wasting a significant amount of network resources.

2.3 Protocol Description

This section describes the proposed protocol that supports both real-time and non-real-time communication services. In this protocol, due to their different QoS requirements, real-time and non-real-time packets are treated differently. Table 2.1 summarizes how the two communication services are differentiated and supported. Throughout this and the next sections, we give a detailed account of each entry in the table. We will first outline the protocol, and then present in the next section the details of both the scheduling and admission-control schemes.

2.3.1 Downlink vs. Uplink

A BS has full control of the transmission of all downlink and uplink packets within its cell. For the downlink, the BS just schedules the transmission of the packets which had already arrived at the BS. In contrast, the exact number of pending uplink packets in each mobile is not known to the BS. Mobiles request the BS's permission to transmit uplink packets, and upon reception of these requests, the BS schedules uplink packet transmissions. These requests are made on a per-connection basis for the real-time service, and on a per-message[2] basis for the non-real-time service as explained below. The BS gives a mobile packet-transmission permission by *polling* the mobile, i.e., checking if it has a packet

which covers small indoor areas [78].

[2] A message, in this chapter, is referred to as a number of non-real-time packets, which arrive at the same time.

to transmit. That is, the BS schedules the order of both polling and downlink packet transmissions.

2.3.2 Real-Time vs. Non-Real-Time

The real-time communication service is provided through connection-oriented communication. Real-time traffic like audio and video usually arrives regularly or periodically and a real-time session/connection usually lasts for a long time, as compared to randomly-arriving non-real-time traffic. Moreover, it is desired to guarantee the requested delivery-delay bound for *each* real-time packet, and hence, we need to set up a real-time connection. For a connection to be admitted, a request should be made to the BS with the connection specification including the requested delay bound. Then, the BS performs an admission test to decide if it is possible to guarantee the requested delay bound without violating the existing connections' guarantees. At run-time, the BS performs priority-based scheduling on packets to guarantee the required delay bounds for all admitted real-time connections provided their channels stay in good condition. A connection is defined to be *uni-directional*, i.e., downlink or uplink. For the sake of notational simplicity, it is assumed that an *active* mobile has only one connection even though it is usually expected to have at least two connections (i.e., one for uplink and the other for downlink) at a time. Active mobile i with downlink (uplink) real-time connection i is called downlink (uplink) real-time mobile i.

Non-real-time communication service, on the other hand, is connection-less (or best-effort). Whenever a non-real-time packet arrives at the BS, its transmission is scheduled by the BS. If an uplink message arrives, or is generated at a mobile, the mobile should request the BS's permission to transmit the message; the BS then schedules the transmission permission for that mobile. Some form of scheduling is also needed to give each mobile fair access to the wireless link. Use of the wireless link is prioritized; when there is a pending schedule to poll mobiles for uplink real-time packets or there are "eligible" (to be defined later) downlink real-time packets, the BS will continue to poll mobiles for uplink real-time packets or transmit downlink real-time packets. However, whenever there is no such a polling schedule nor any downlink real-time packet to transmit, the BS will transmit downlink non-real-time packets or poll mobiles for the transmission of uplink non-real-time packets. As will be described later, the prioritized channel usage becomes more complicated when location-dependent errors must be handled efficiently.

2.3.3 Slotted Channels

A frequency channel is slotted along the time axis. There are three types of slots: *packet-transmission slot*, *transmission-request slot*, and *control mini-slot*. For simplicity, it is assumed that both packet-transmission and transmission-request slots have the same size T_s while the control mini-slot has the size T_{ms} $(= T_s/K)$, where K is an even number much larger than 1. First, a (fixed-size) downlink or uplink packet is transmitted in a packet-transmission slot. Within a packet, the following information can be piggybacked:

- Information within a downlink packet to give a mobile permission to transmit uplink packets;
- ACK/NAK for a previously-received packet within a downlink or uplink packet;
- A message-transmission or connection-setup request by a mobile within an uplink packet.

Second, the transmission-request slot is used by mobiles to convey a request to the BS. Specifically, the usage is three-fold:

- A non-real-time message transmission can be requested with (mobile ID, Class II-A/B ID, the number of packets in the message);
- A real-time connection-setup request can be made with (mobile ID, connection specification);
- A hand-off request can be made by a mobile, which is communicating with an adjacent BS in the overlapping region of two cells, with (previous cell ID, mobile ID, connection specification).

Finally, using the control mini-slot, a control packet is transmitted by the BS or by a mobile. A control packet can be used for the following cases:

- For the BS to poll a mobile for transmission of an uplink packet;
- For the BS to issue a transmission-request slot by announcing that the next slot is a transmission-request slot;
- For the BS to announce the admission-test result for a newly-requested or handed-off real-time connection;

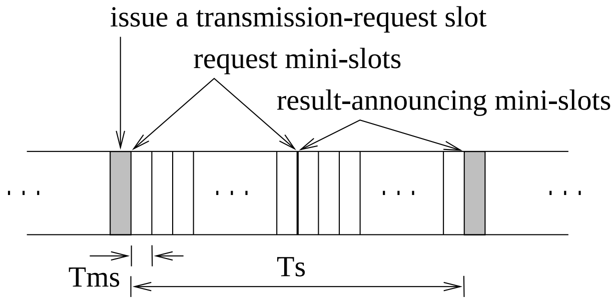


Figure 2.2: The structure of a transmission-request slot.

- For a real-time mobile to return the transmission permission to the BS without transmitting any packet;
- For the BS or a mobile to send an ACK/NAK for each received packet;
- For the BS to send or receive a channel-probing control packet to predict the channel condition a mobile is experiencing.

More than one of the above can be piggybacked in a control packet, and only the last three cases use uplink control packets.

2.3.4 Transmission Request Access

As shown in Figure 2.2, a transmission-request slot is divided into two parts: (1) the first half is the set of $K/2$ *request mini-slots* used by mobiles; and (2) the second half is the set of $K/2$ *result-announcing mini-slots* corresponding to the previous request mini-slots. Then, the first K_{ho} request mini-slots are reserved for real-time hand-off requests. The request mini-slots are used by a slotted ALOHA-like random access protocol: when the BS issues a transmission-request slot, each mobile with pending requests decides whether to make a request with probability p_r or not (i.e., in a p_r-persistent manner). If it decided to send the request, the mobile randomly chooses one of the last $(K/2 - K_{ho})$ request mini-slots, then sends it in that chosen mini-slot. One difference with real-time hand-off requests is that they are made through one of the first K_{ho} request mini-slots. The result of each of $K/2$ mini-slots can be success,[3] or collision, or empty/unused. Using each of the next $K/2$ downlink mini-slots, the result of the corresponding request mini-slot is announced.

The reason why K_{ho} mini-slots are reserved for real-time hand-off requests is because without this reservation, the hand-off requests, which should be made in a timely manner to

[3]Due to the *capture effects* [2], a request can be transmitted successfully even in the presence of concurrent requests from other mobiles.

guarantee the connection's deadline, might be swamped by many requests for non-real-time traffic transmissions. If any of K_{ho} reserved mini-slots results in collision, all request mini-slots in the following transmission-request slot are dedicated to real-time hand-off requests (i.e., $K_{ho} = K/2$) so that the collided hand-off requests can be made successfully with a much higher probability. The initial value of K_{ho} can be adaptively set according to the hand-off rates. For example, if there was a collision in a reserved request mini-slot, the size can be increased by one (and this new value is used after the request slot with $K_{ho} = K/2$ as described above), then if there is no such collision for a certain time period, it can be decremented by one.

The request access probability p_r is determined by each mobile independently of others. For the first attempt of a message-transmission request, p_r is set to 1. If the request becomes unsuccessful, then p_r is decreased according to

$$p_r := \frac{p_r}{p_r + 1}, \tag{2.1}$$

thus following the sequence $1, 1/2, 1/3, \cdots$ (i.e., the "harmonic" backoff) [16]. Each unsuccessful request should be retransmitted p_r-persistently until it is successfully transmitted. Note that there are two ways to transmit a request: one through a transmission-request slot, and the other by piggybacking it in an uplink packet. Which of the two ways should be used depends on the request type. First, for a non-real-time message-transmission request, if a mobile has not yet transmitted all of its pending uplink packets of which transmission requests had already been made to the BS, then, upon arrival of another message, the mobile need not use a contention-based transmission-request slot. Instead, the request can be piggybacked in an uplink packet. Second, for a real-time connection-setup request, the mobile transmits the request using either of two ways, whichever becomes available first. Third, a hand-off request can be made only through a transmission-request slot since the BS is unaware of the existence of that mobile. In this case, the mobile will access transmission-request slots in a 1-persistent manner. Due to the limited packet size, one may not be able to piggyback all pending requests in a single uplink packet. In such a case, priority is given in the order of (1) real-time connection-setup request; (2) class II-A message-transmission request; and (3) class II-B message-transmission request.

2.3.5 Error-Handling Mechanisms

To handle location-dependent, time-varying, and bursty channel errors, the channel state can be predicted via channel probing before a packet is transmitted. That is, before polling a mobile or transmitting a downlink packet to the mobile, the BS transmits a probing control packet to the mobile, which then returns the control packet to the BS. If the BS does not receive the probing control packet correctly from the mobile, the channel is considered bad, and the polling or transmission is deferred. The channel condition is estimated before each real-time packet transmission in order to reduce the need for retransmitting real-time packets, since retransmission can be really harmful in meeting deadlines while it is optional for the non-real-time case (to reduce its significant overhead).

The ACK/NAK is sent from the receiver upon packet reception.[4] The most desirable way to send an ACK/NAK is to piggyback it in another packet. ACK/NAK for an uplink packet can always be piggybacked since after each uplink packet transmission, a downlink packet (or a control packet) will always be transmitted from the BS. Note that a downlink (control) packet and the piggybacked ACK/NAK may have different destination mobiles. The packet transmitted after a downlink packet can not be a regular uplink packet from the destination mobile of the previous downlink packet, because the destination mobile transmits an ACK/NAK via an uplink control packet right after it received a downlink packet. If the destination mobile of a downlink packet does not know the packet's destination due to an error in the packet, then the mobile won't transmit NAK either. If the BS does not receive an ACK/NAK within T_{ms} after transmitting a downlink packet, the packet will be assumed lost and hence retransmitted later. In summary, (1) ACK/NAK for each packet is transmitted right after its reception; (2) ACK/NAK for an uplink packet is always piggybacked in a downlink (control) packet; and (3) after a downlink packet transmission, a control packet containing ACK/NAK is transmitted from the destination mobile unless an uplink packet transmission from that mobile is scheduled next.

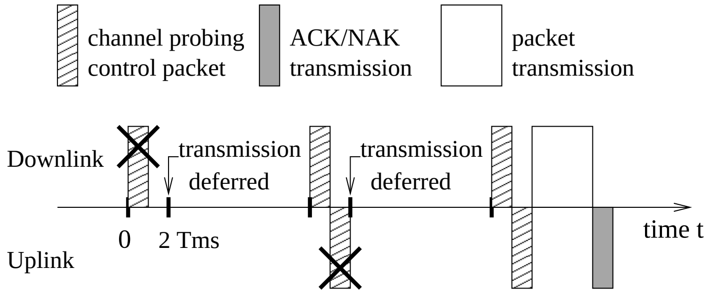
When a mobile is polled for a real-time (non-real-time) packet transmission, if it does not have any real-time (non-real-time) uplink packet to transmit,[5] the mobile transmits an uplink control packet, saying that it has no packets to transmit. So, if the BS does not receive any (control) packet within the mini-slot period T_{ms}, the polling control packet had not been received by the mobile due to an error, implying that the mobile's channel is bad, and hence the polling is deferred. The issue of handling this deferment is addressed next.

2.3.6 Real-Time Communication Service

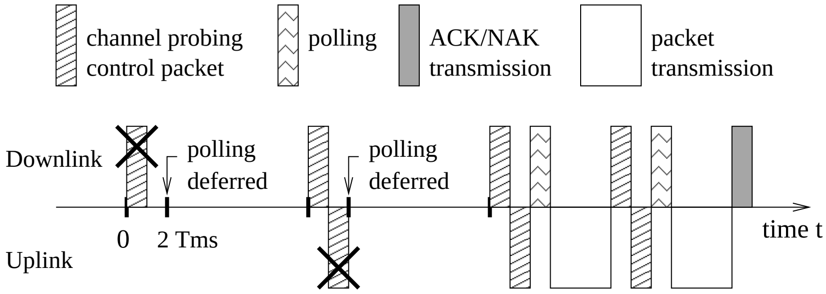
Real-time connection i is specified by a triplet (M_i, T_i, D_i), where

[4]Actually, right after determining if the received packet can be corrected by the channel decoder.

[5]Even though this should not happen with the non-real-time case if everything went correctly because the polling is based on transmission requests from mobiles



(a) Downlink transmission



(b) Uplink transmission via polling

Figure 2.3: Real-time communication slot structures with channel probing.

- (M_i, T_i): M_i is the maximum number of packets that can arrive in an interval of length T_i;
- D_i: the packet delivery delay (i.e., from the mobile to the BS, or from the BS to the mobile, hence a fraction of the target end-to-end delay) bound for real-time packets of connection i, which has the following relationship with T_i:

$$D_i \geq D_i^{min} = \begin{cases} T_i, & \text{for downlink,} \\ 2T_i, & \text{for uplink.} \end{cases} \tag{2.2}$$

As will be clear later, the BS will admit new connections and schedule packet-transmissions and channel-polling based on the minimum delay bound D_i^{min} so that each packet's delay is bounded by D_i^{min} as long as the channel condition continues to be good. Once a packet transmission is deferred or fails, the BS will attempt to deliver the packet within D_i in a best-effort manner. Note that the larger D_i, the more likely packets will be delivered in time under a time-varying channel condition.

To set up a real-time connection, the source node (which is a mobile if it is an uplink connection) will request the BS to set up the connection with its specification triplet. The BS will then perform the admission test given in Section 2.4.2. Depending on the test result, the new connection will be admitted or rejected. This test result is conveyed to the mobile through a downlink control packet. If it is admitted, a connection is set up: if it is an uplink connection, the BS starts to schedule the polling for that mobile, or if it is a downlink connection, it starts to schedule downlink real-time packets arriving at the BS. Even though it was described in the context of a new connection request, the same procedure can be used for a hand-off connection request. For an uplink connection, the BS just polls mobiles assuming the contracted traffic characteristics because the BS does not know the exact status of a mobile's pending real-time packets. The control packet for polling a mobile contains the mobile ID. When a mobile is polled, it can transmit one pending real-time packet through the uplink.

Figure 2.3 shows how the channel probing and deferment work for both downlink and uplink transmissions, where the label 'X' represents loss of a control packet. The BS will attempt to transmit each deferred polling or downlink packet later so that its delivery bound can be met if the channel condition improves before the deadline expires without compromising other undeferred packet transmissions as described in Section 2.4.1. The channel-state probing can be used adaptively; that is, a real-time connection can be serviced without this process initially, but later if this connection is determined to suffer from time-varying errors, this process can be invoked. Especially, a mobile which stays close to the BS will be experiencing a very good channel, so it will not need this process at all. Note that this channel probing imposes bandwidth overhead; $2T_{ms}$ is consumed for each channel probing. As shown in Figure 2.3, the transmission of a real-time packet (both downlink and uplink) consumes $3T_{ms} + T_s$. Note that the ACK/NAK of an uplink packet can always be piggybacked in a downlink (control) packet, so it was not counted.

2.3.7 Non-Real-Time Communication Service

The BS schedules the transmission of downlink non-real-time packets or the polling for uplink non-real-time packet-transmission permissions with lower priority than the scheduled real-time traffic. Important concerns about non-real-time communication service include:

- Virtually error-free transmission achieved by our retransmission-based error control mechanism;
- (Long-term) fair access to the link of mobiles: a round-robin-based packet/polling scheduling is used for fairness;
- Maximizing aggregate throughput: our non-real-time scheduling is geared toward re-

duction of control packet overheads associated with ACKs/NAKs and polling. Moreover, channel probing is not used as long as a mobile's channel continues to be good.

2.4 Run-Time Scheduling and Admission Control

As mentioned earlier, D-TDD requires uplink and downlink real-time packets to be multiplexed and scheduled over the same frequency channel. We need two different packet/polling scheduling policies for real-time and non-real-time services due to their distinct requirements. To set up a real-time connection, the BS performs an admission test to check if it can guarantee the timely delivery of every real-time packet from the new channel without compromising other existing guarantees.

2.4.1 Real-Time Traffic Scheduling

The conventional first-in-first-out (FIFO) scheduling is known to be unable to bound the packet-delivery delay. So, we adopt a non-preemptive earliest-deadline-first (EDF) scheduling algorithm — the earlier the deadline the higher the priority. Liu and Layland [62] proved that the deadline-driven scheduling is optimal among all dynamic-priority scheduling policies when the deadline of each task is equal to the end of its period. Our real-time scheduling and admission control are based on the real-time channel protocol in [51].

Downlink Packet Scheduling

The *logical arrival time* for the n-th packet, arrived at the BS at time $t_i(n)$, of real-time connection i is defined as:

$$l_i(n) = \begin{cases} t_i(n) & \text{for } 1 \le n \le M_i, \\ \max\{l_i((\lfloor n/M_i \rfloor - 1)M_i + 1) + T_i, \;\; t_i(n)\} & \text{for } n > M_i. \end{cases} \tag{2.3}$$

The *logical deadline* this packet is then $l_i(n) + D_i^{min}$. A packet with $t_i(n) < l_i(n)$ (i.e., arrived too early) is not transmitted until it becomes 'current' since its immediate transmission might result in buffer overflow at the destination mobile. By using this logical deadline, one can protect well-behaving connections from misbehaving ones. For example, much more than M_i packets could arrive during T_i from a misbehaving connection with the contract (M_i, T_i, D_i), which can cause violation of the bounded-delivery guarantees of packets belonging to other well-behaving connections if their logical deadlines are not enforced. Note that D_i^{min} is added to the logical arrival time instead of D_i in the calculation of the logical deadline. In this way, the BS attempts to deliver each real-time packet within its minimum delay bound as long as the destination mobile's channel remains in good condition. A deferred or erroneously-transmitted packet will be (re)-transmitted when the channel is predicted and its actual deadline D_i can be met.

When the n-th packet, $p_i(n)$, of real-time connection i, C_i, arrives at the BS from the wired network at time $t_i(n)$, it is placed in a connection-specific FIFO queue $Q[i]$, and its logical deadline is calculated. Then, the packet's connection ID is fed into queue **R** (meaning 'ready') if $t_i(n) \geq l_i(n)$, and the entries of this queue are sorted according to the corresponding packets' deadlines. On the other hand, if $t_i(n) < l_i(n)$, this connection ID will be held in another queue **H** (meaning 'hold'), and will later be fed into **R** at time $t = l_i(n)$ (i.e., when it becomes current). When **R** is selected for service, the connection, say C_j, whose ID is at the head of **R**, is served, and then the connection ID is removed from **R**. To serve C_j, the BS first (i) drops all those packets in queue $Q[j]$ whose deadlines cannot be met, (ii) probes the destination mobile's channel, and finally, (iii) transmits the packet at the head of queue $Q[j]$ if the channel condition is predicted to be good; else defers its transmission and places the connection ID in a FIFO queue **D** (meaning 'deferred').

If the packet transmission upon prediction of a good channel condition results in a NAK, the connection ID is placed into a FIFO queue **B** (meaning 'back-logged'). Both queues **D** and **B** are served similarly to **R**. For convenience, the process of serving queue **X** is referred to as a "queue **X** service," where **X** is **R** or **D** or **B**. Note that the (re)-transmission of an once-deferred or unsuccessfully-tried packet should not cause other undeferred packets to miss their deadlines. To achieve this, we use a book-keeping counter, called the *credit counter* (CC), which records the time both queues **D** and **B** can be served continually without delaying queue **R** service. The CC is updated as follows:

- $CC := CC + 5T_{ms} + T_s$, if a queue **R** service results in a deferment;
- $CC := CC + 2T_{ms}$, if a queue **R** service results in a packet transmission;
- $CC := CC - 2T_{ms}$, if a channel is probed for a queue **D** or **B** service;
- $CC := CC - (T_{ms} + T_s)$, if a packet is transmitted via a queue **D** or **B** service;
- $CC := CC - T_{tr}$, if the link is used for non-real-time traffic or transmission-requests for the duration of T_{tr};
- $CC := 0$, if $CC < 0$ after any of the above updates.

As will be described later in Section 2.4.2, $5T_{ms} + T_s$ is assumed necessary for each packet transmission during the admission-control phase, because packet transmission without deferment takes $3T_{ms} + T_s$ and additional $2T_{ms}$ is reserved for one extra channel probing for each packet. This is why $5T_{ms} + T_s$ ($2T_{ms}$) is added when a queue **R** service results in a deferment (packet transmission).

When $CC \geq 3T_{ms} + T_s$, queue **D** is given priority over queue **R**. Then, **B** is given next priority after **D**. Basically, the channel usage is prioritized in the order of **D**, **B**, and **R** if $CC \geq 3T_{ms} + T_s$, and **R**, **D**, and **B** if $CC < 3T_{ms} + T_s$. Whenever the wireless link becomes free (after completing an on-going transmission), the BS will choose the non-empty highest-priority queue for the next channel usage. When queue **D** is chosen, the connection specified by the D_{index}-th entry of **D** is served. Unless the queue **D** service results in a deferment, the D_{index}-th entry is removed, and all the following entries are shifted toward the head within **D**, i.e., the $(i+1)$-th entry becomes the i-th entry for all $i \geq D_{index}$. The index D_{index} is updated as follows, with the initial value $D_{index} = 0$:

- $D_{index} := 0$, if **D** becomes empty;
- $D_{index} := D_{index} + 1$, when a queue **D** service results in a deferment;
- $D_{index} := 1$, when (1) an entry in queue **R** is fed into the empty queue **D**, or (2) the last entry in queue **D** is served, or (3) CC becomes less than $3T_{ms} + T_s$.

In fact, if $D_{index} = 1$, there is another condition — that is, the queue **D** service flag $D_{flag} = 1$ — to serve queue **D**. Otherwise, the link is used to serve the queue with the next priority. The value of D_{flag} is updated as follows, with the initial value $D_{flag} = 0$:

- $D_{flag} := 1$, if any packet (other than control packets) is transmitted over the channel;
- $D_{flag} := 0$, when an entry in queue **R** is fed into the empty queue **D**, or a queue **D** service results in a deferment when $D_{index} = 1$.

Note that by using D_{flag}, once a packet transmission is deferred, at least one packet is transmitted before serving this packet again. This is reasonable since trying to continuously serve a deferred packet can waste bandwidth significantly due to continuous channel probes without its successful transmission. The same rule is used for queue **B** by replacing D_{index} and D_{flag} with B_{index} and B_{flag}, respectively.

Note that the entries in real-time queues **R**, **D**, and **B** specify connection IDs, not packet IDs, and the packet with the smallest deadline in each connection — irrespective of which queue specifies this connection for a service — is transmitted. In this way, we can maximize the chance to deliver packets before their deadlines. Figure 2.6 summarizes how to determine which queue to serve, including the non-real-time queues that will be covered later.

Polling Order

As mentioned earlier, the polling order for uplink real-time mobiles is determined based on the contracted traffic characteristics as follows:

1. The BS generates a polling request for uplink real-time connection i with (M_i, T_i, D_i), periodically once every T_i.

2. The request generated at time t_i is sorted in queue $\mathbf{R}$ according to the deadline $t_i + D'_i$ along with downlink connection IDs, where $D'_i = D_i^{min} - T_i = T_i$. Conceptually, a polling request for connection i is considered as a packet of size $M_i(3T_{ms}+T_s)$ including the channel probing overhead.

3. When the request is at the head of queue $\mathbf{R}$, the BS will serve[6] uplink real-time mobile i up to M_i times consecutively until the channel for this mobile is predicted to be bad, or this mobile does not have any more pending packets to transmit.

The deadline-driven scheduling plus the admission control described in Section 2.4.2 will ensure that the BS can start probing the channel for uplink connection i as late as $D'_i - M_i(3T_{ms}+T_s)$ after generation of the last polling request. Then, for the k-th polling request generation time G_k (where $G_{l+1} - G_l = T_i$ for all l), all of M_i packets generated during the interval of $[G_{k-1},\ G_k]$ will be transmitted by time G_{k+1} as long as the channel condition continues to be good. So, the worst-case delay will be $G_{k+1} - G_{k-1}$, which is $2T_i$ or D_i^{min}.

If the BS sends a channel probing control packet to a mobile when this mobile does not have any pending real-time packet, the mobile will specify that it does not have any real-time packet via the returning control packet so that the BS need not poll the mobile. On the other hand, if a mobile completes the transmission of all of its pending real-time packets after being polled N_i times (where $N_i < M_i$), the BS will be informed of this via the N_i-th packet so that the BS need not poll the mobile again. When polling for connection i in queue $\mathbf{R}$ is deferred after polling it N_i times, where $0 \le N_i < M_i$, the polling request is queued up in $\mathbf{D}$ with $P_i = M_i - N_i$, which is the number of polls to be taken, as well as the connection ID. When this request is to be served later, mobile i will be polled up to P_i times consecutively. The credit counter is updated with polling as follows:

[6] By 'serving' an uplink real-time mobile, we imply that this mobile's channel is probed, and the mobile is polled if the channel is predicted to be good; otherwise, the polling is deferred.

- $CC := CC + 2T_{ms}N_i + (3T_{ms} + T_s) + (M_i - N_i - 1)(5T_{ms} + T_s)$, if polling for connection i in queue **R** is deferred after it is polled N_i times;
- $CC := CC - 2T_{ms} + M_i(5T_{ms} + T_s)$, if mobile i had no pending packet to transmit;
- $CC := CC + 2T_{ms}N_i + (M_i - N_i)(5T_{ms} + T_s)$, if mobile i completed the transmission of its pending packets after being polled N_i times (where $0 < N_i \leq M_i$), i.e., it had N_i pending packets;

The other rules associated with serving a polling request are the same as those for real-time downlink packet transmissions. For each erroneously-transmitted uplink packet, a polling request is queued up in **B**.

Transmission-Request Slot Scheduling

To achieve low latencies for message transmissions or connection setup/hand-off requests, it is necessary to issue transmission-request slots regularly and frequently. When it is desirable to have a transmission-request slot once every T_{req} time units, this slot is issued with the same scheduling used for polling with $T_i = T_{req}$, $M_i = 1$, and $D_i = D_i^{min} = 2T_{req}$. Note that no channel probe is made before issuing a transmission-request slot since there is no specific destination mobile in this case. For the rest of this chapter, we "imagine" there exists a virtual uplink connection with $(M_i, T_i, D_i) = (1, T_{req}, 2T_{req})$, which is equivalent to transmission-request slots. However, the scheduling discussed so far is valid only when the system is busy, i.e., there always exist some downlink packets or uplink pollings scheduled continuously. If not, the BS basically has nothing to do. Then, the BS continues to issue transmission-request slots, thus reducing the transmission-request access latency as much as possible. With the above scheduling, during the system busy period, a transmission-request slot will appear, on average, once every T_{req} time units, and the time span between two consecutive transmission-request slots is bounded by $2T_{req}$.

Note that the packet scheduling itself cannot guarantee a bounded delivery delay. A proper admission control should be performed to determine if a set of connections can be guaranteed to have their required delivery-delay bounds using the above scheduling mechanisms.

2.4.2 Admission Test

We now consider the admission test for a new connection request. If the other communicating party of a mobile is outside of the cell, the entire communication path will be divided into one or two wireless links and a route within the wired network. The admission test should consider end-to-end communication. However, here we consider the admission test for a wireless link only since the transmission over the wired links can be handled by other known schemes like the one in [51]. (Admission tests for the wired-network part have been studied by many other researchers, and depend on the scheduling policy used in the intermediate nodes.) For a new connection to be established, the results from admission tests on all components (on the communication path) should be positive.

As mentioned already, the admission control is applied so that the delay of each packet in connection i for all i is bounded by D_i^{min} as long as mobile i's channel stays in good condition. Let's consider a set of connections $\{C_i = (M_i, T_i, D_i) \mid i = 1, 2, \cdots, N_r\}$. The admission test is composed of two phases, i.e., a bandwidth test phase, and a delay-bound test phase. For the newly-requested connection *new*, the bandwidth test is to check if the sum of reserved bandwidth for each connection including overheads satisfies the following condition:

$$(T_s + 5T_{ms}) \left(\sum_{i=1}^{N_r} M_i/T_i + M_{new}/T_{new} \right) \leq 1 - \Delta_r, \qquad (2.4)$$

where Δ_r is the portion of the link capacity reserved for real-time packet retransmissions and non-real-time traffic. Note that $(5T_{ms}+T_s)$ instead of $(3T_{ms}+T_s)$ is needed for a packet transmission to reserve additional $2T_{ms}$ for an extra channel probing. The larger Δ_r, the more packets will make their deadlines even under bad channel conditions. The value of Δ_r can be adapted depending on the channel conditions of the existing connections and the observed packet-dropping probability.

If a new connection does not satisfy the first condition, it is rejected while it is considered for the second phase delay-bound test only if it satisfies the first condition. The delay-bound test is set up as follows. First, suppose the elements of set $\{C_i\}$ are arranged in ascending order of D_i', where

$$D_i' = \begin{cases} D_i^{min}, & \text{if downlink,} \\ D_i^{min} - T_i, & \text{if uplink.} \end{cases} \qquad (2.5)$$

In fact, $D_i' = T_i$ for both uplink and downlink packets. Note that when a packet (or polling request) from each of N_r connections arrives at the same time, the packet (or polling request) from a smaller-numbered connection will have higher priority. To establish a real-time connection, the BS needs to check its schedulability with the connection's worst-case delay. We can derive an admission test similar to that in [51]. For downlink connection i, M_i packets are assumed to arrive periodically at the beginning of each virtual packet-arrival period T_i, which represents the worst case in terms of the delivery delay of these packets.

Then, the last of these M_i packets should finish the transmission within D_i' as long as the channel continues to be good. If this works correctly, uplink packets are also transmitted by the minimum delay bound as explained in Section 2.4.1.

Now, for the schedulability test, we consider the transmission time of the last of M_i packets of downlink connection i. The worst-case packet-delivery delay occurs when its arrival coincides with all other higher-priority packet arrivals. This "critical" instant is denoted as time $t = 0$. The transmission of the packet can be delayed by all instances of higher-priority packets (and pollings) at $t = 0$ and by the subsequent instances of these packets (and polling requests). Arrival times of these instances within the delay bound D_i' are given as:

$$A_i = \{D_i'\} \bigcup \{kT_j \mid j = 1, 2, \cdots, i-1; k = 1, 2, \cdots, \lfloor (D_i'/T_j) \rfloor\}. \tag{2.6}$$

The total time required for transmitting this packet and others with priority equal to, or higher than, this packet (or polling) is given by:

$$W_i(t) = T_{max\_poll} + M_i(5T_{ms} + T_s) + \sum_{j=1}^{i-1} \{M_j(5T_{ms} + T_s)\}\lceil t/T_j \rceil. \tag{2.7}$$

Note that the first and second terms of Eq. (2.7) denote, respectively, an upper bound of the time to complete in-progress packet transmission or polling and the time to transmit M_i packets of real-time connection i. The third term denotes the time to transmit all packets with the same or higher priority. Note that $(5T_{ms} + T_s)$ instead of $(3T_{ms} + T_s)$ was again assumed necessary for a packet transmission. In fact, T_{max_poll} is affected by uplink connections since the time to complete an in-progress downlink packet transmission will be just $3T_{ms} + T_s$. An uplink real-time mobile i will be polled up to M_i times consecutively once it is polled:

$$T_{max\_poll} = \max\{2T_s, \max_{i \in \mathbf{S}_u}\{M_i(3T_{ms} + T_s)\}\}, \tag{2.8}$$

where $\mathbf{S}_u$ is the set of uplink connections' indices. By definition, T_{max_poll} will be at least $2T_s$ even when there is no uplink real-time connection. This follows from the maximum possible non-real-time traffic transmission without interrupting real-time packet transmissions as explained in Section 2.4.3. A set of connections $\{C_i\}$ is said to be *schedulable* (i.e., all packet minimum delivery-delay constraints will be met as long as the channel conditions continue to be good) if the following condition holds:

$$\begin{aligned} W_i(t) \le t, \qquad & \text{for some } t \in A_i, \\ & \text{for all } i = 1, 2, \cdots, N_r, \end{aligned} \tag{2.9}$$

where N_r is the number of connections, and A_i is given as in Eq. (2.6). From this schedulability condition, we now derive the second phase delay-bound test of a new connection when there already exists a set of connections $\{C_i\}$ in the system. Note that the schedulability condition in Eq. (2.9) already holds for the set of existing connections $\{C_i\}$.

1. Upon arrival of a new connection request, the BS inserts the newly-requested connection new in the set of connections $\{C_i\}$ according to the ascending order of D'_{new}.

2. If new is a downlink connection, the admission test is to check if the following condition holds:

$$W_i(t) \leq t, \qquad \text{for some } t \in A_i. \\ \text{for } i = new, new+1, \cdots, N_r, \tag{2.10}$$

3. If new is an uplink connection, T_{max_poll} is re-calculated using Eq. (2.8).

   - If T_{max_poll} hasn't changed, the admission test is given by Eq. (2.10);
   - If T_{max_poll} has changed, the admission test is given by Eq. (2.9).

Note that if T_{max_poll} hasn't changed, the performance guarantees of connections with smaller D'_i's are not affected by the admission of connection new.

2.4.3 Non-Real-Time Traffic Scheduling

The transmission of non-real-time traffic is based on a modified round-robin scheduling policy with a set of two prioritized queues. Using the round-robin scheduling, it is possible to give each mobile fair channel access as long as each mobile's channel condition continues to be good. When there are M_o mobiles in the cell, the BS is equipped with the following two service queues (in that order of priority): (1) a class II-A round-robin queue (with M_o entries), called queue **RR.A**, and (2) a class II-B round-robin queue (with M_o entries), called queue **RR.B**. The i-th entry of a round-robin queue, corresponding to mobile i, is composed of two parallel queues for the downlink and uplink, respectively. The uplink queue has a number of uplink packets requested for transmission permissions, which is the sum of the number of packets in transmission permission-requested messages. On the other hand, the downlink queue buffers the downlink packets in a FIFO manner which had arrived at the BS. Figure 2.4 shows a round-robin queue for M_o mobiles in the cell.

We first describe how the round-robin queues work assuming that there is no channel error, then address how to handle channel errors. For the i-th entry's turn, up to two packets

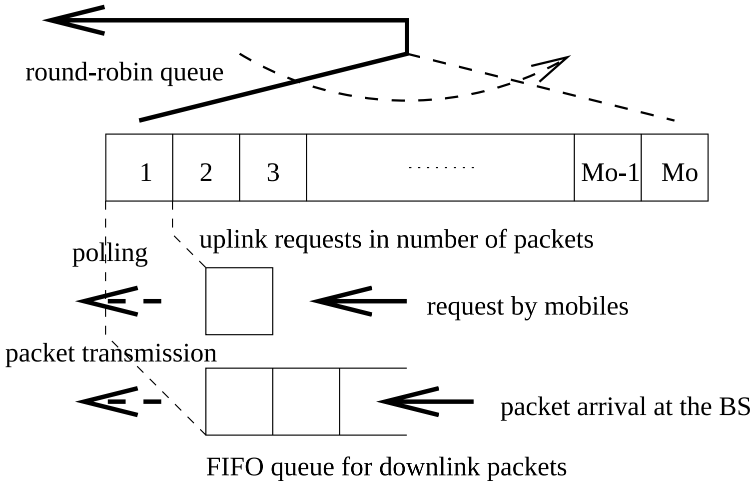


Figure 2.4: A round-robin queue for non-real-time communications.

for mobile i are served. To reduce the ACK/NAK-transmission and polling overheads, it is desirable to sequentially serve one downlink and one uplink packets. In this case, a downlink packet destined for mobile i is transmitted first with up to two different pieces of piggybacked information: (1) ACK/NAK for a previously-transmitted uplink packet if the previous packet was uplink; and (2) the transmission permission for mobile i to transmit an uplink packet (i.e., a polling to mobile i). Then, mobile i will transmit an uplink packet with ACK/NAK for the previously-received downlink packet. Note that through this transmission order, two mini-slots were saved, i.e., for one downlink control packet to poll mobile i, and for one uplink control packet for the ACK from mobile i. If the i-th entry does not have both downlink and uplink requests, up to two downlink or uplink packets for mobile i can be transmitted consecutively using two mini-slots for polling or ACK transmissions. An uplink packet and a downlink packet can also be transmitted sequentially if a downlink packet arrives during the transmission of an uplink packet.

An example of this non-real-time scheduling is shown in Figure 2.5, where a square and a small rectangle represent a packet transmission slot and a mini-slot, respectively. The notation used is: $U_{i.j}$ is the j-th uplink packet transmission of mobile i; $D_{i.j}$ is the j-th downlink packet transmission to mobile i; $POLL\_M_i$ is a polling to mobile i; $ACK\_\{x\}$ is the ACK/NAK for the previous packet transmission x; M_i represents the transmission from mobile i, and '+' represents piggybacking of the following information in the same (control) packet. Four different cases are shown in chronological order: (1) an uplink packet from, and a downlink packet sequentially to, mobile 0; (2) two uplink packets from mobile 1; (3) a downlink packet to, and an uplink packet sequentially from, mobile 2; and (4) two downlink

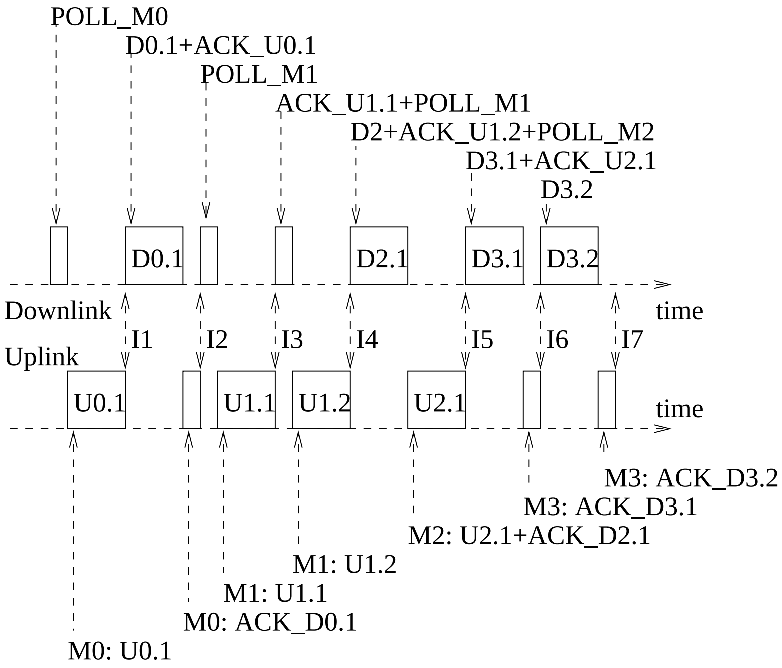


Figure 2.5: An example of non-real-time transmissions.

packets to mobile 1. Note that two control packets are used to transmit two packets from/to mobiles 0, 1, and 3. However, no control packet is used for mobile 2 to receive one downlink and transmit one uplink packet.

Note that real-time traffic or higher-priority non-real-time traffic (such as class II-A traffic if the traffic considered is class II-B) can take over the channel according to the priority rule when the wireless link becomes free after (1) the completion of the ACK transmission for a non-real-time downlink packet, or (2) the completion of a non-real-time uplink packet transmission. For the latter case, the BS needs to piggyback the ACK/NAK within the following (control) downlink packet. So, the BS can switch to the real-time traffic service or a higher-priority non-real-time traffic service at the marked (with 'I_n') moments in Figure 2.5. The worst-case delay before switching to the real-time traffic service is shown in the figure, i.e., from I_4 to I_5. To interrupt at I_5, there is a delay of up to $2T_s$ time units. This is why T_{max_poll} should be at least $2T_s$ in Eq. (2.8). Even though it was not shown in the figure, the request for an uplink message transmission can also be piggybacked in an uplink packet.

Now, let's assume that there was a NAK associated with a packet transmission from/to mobile i. Then, the entry for this mobile in the particular round-robin queue is marked "back-logged" indicating that the mobile's channel condition is bad. For this entry's next

whenever (the link becomes free) {
  if ($C \geq 3T_{ms} + T_s$ and ($D_{index} > 1$ or ($D_{index} = 1$ and $D_{flag} = 1$)))
    serve queue **D**;
  else if ($C \geq 3T_{ms} + T_s$ and ($B_{index} > 1$ or ($B_{index} = 1$ and $B_{flag} = 1$)))
    serve queue **B**;
  else if (queue **R** is non-empty)
    serve queue **R**;
  else if ($D_{index} > 1$ or ($D_{index} = 1$ and $D_{flag} = 1$))
    serve queue **D**;
  else if ($B_{index} > 1$ or ($B_{index} = 1$ and $B_{flag} = 1$))
    serve queue **B**;
  else if (queue **RR.A** is non-empty and $RR^{A}_{flag} = 1$)
    serve queue **RR.A**;
  else if (queue **RR.B** is non-empty and $RR^{B}_{flag} = 1$)
    serve queue **RR.B**;
  else issue a transmission request slot;
}

Figure 2.6: A pseudo-code to determine which queue to serve when the wireless link becomes free.

service turn, the mobile's channel is probed first. If the channel is predicted to be good, the mobile's packets are served, and the entry is marked "active." Otherwise, the entry stays back-logged without the mobile being served at all. For the turn of the i-th entry, which is back-logged, the i-th *non-real-time compensation counter* $NCC.(*)[i]$ is increased by the number of packets to be served if the entry was active, where $(*)$ is A (B) if the queue is for class II-A (class II-B). This counter is increased when an entry becomes back-logged as well. For example, if uplink class II-A packet $U_{0.1}$ in Figure 2.5 is erroneously transmitted, the 0-th entry of queue **RR.A** is marked back-logged, and $NCC.A[0]$ is increased by two since both $U_{0.1}$ and $D_{0.1}$ could be transmitted if the channel was good. For the turn of the i-th entry which becomes active, as many as $NCC.(*)[i] + 2$ packets of mobile i are served continually. Then, $NCC.(*)[i]$ is reset to zero. In this way, we can achieve long-term fairness among mobiles. Note also that before serving the first entry in a round-robin queue, if all the entries are marked back-logged, the round-robin queue service flag $RR^{(*)}_{flag}$ is set to zero, where $(*)$ is again A (B) if the queue is for class II-A (class II-B). $RR^{(*)}_{flag}$'s are set

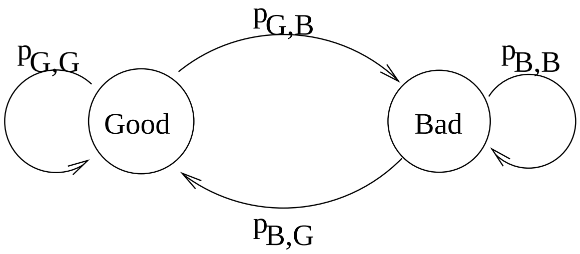


Figure 2.7: Two-state Markov chain modeled channel.

to 1 whenever a packet is transmitted over the channel. A round-robin queue can be served only when its flag is 1. This way we can reduce the overhead associated with the channel probing for non-real-time traffic while being able to serve class II-B traffic when all mobiles with class II-A traffic are experiencing bad channel conditions. Note that the channel is probed for back-logged connections only in order to reduce unnecessary bandwidth waste for channel probing.

An ACK/NAK can be lost due to errors. Because the feedback should be received right after the packet transmission, the sender assumes, in case of ACK/NAK loss, that the packet transmission was not successful and re-transmits the previous packet later. The receiver, in turn, knows that the previous ACK/NAK was lost when it receives a duplicate packet. If the receiver is the BS, the BS needs to update a round-robin queue's entry by increasing the number of uplink packet requests by one since it received the same packet again. Finally, recall that the above-mentioned scheduling is applied whenever (1) all the real-time queues **R**, **D**, and **B** are empty, or (2) queue **R** is empty, $D_{flag} = 0$, $D_{index} = 1$, $B_{flag} = 0$, and $B_{index} = 1$. Figure 2.6 provides a pseudo-code to determine which queue to be served whenever the wireless link becomes free.

2.5 Experimental Results and Discussion

This section evaluates and discusses the performance of the proposed scheme. We first state the assumptions and the system specification used in the evaluation study.

2.5.1 Assumptions and System Specification

We assume that there are N_u $(= 10)$ mobiles in the cell, and the duration T_{ms} of a mini-slot is used as a basic *time unit*. Only one request can be transmitted in a transmission-request slot or piggybacked in an uplink packet by a mobile. The wireless channel is modeled as follows:

C1. The channel between the BS and a mobile is modelled by a Markov chain as shown in Figure 2.7. The channels of two different mobiles are assumed to be independent.

C2. A transition between the good and bad states can happen at mini-slot boundaries with the transition probability $p_{G,B}$ ($p_{B,G}$) from the good to bad states (from bad to good states).

C3. For a given average time duration T_G (T_B) at the good (bad) state, $p_{G,B} = 1/T_G$ ($p_{B,G} = 1/T_B$). We assume that $T_G = 2000$, and $T_B = 100$.

C4. Each request could be received in error (1) when it is corrupted due to channel errors, or (2) when it collides with another request within the same request mini-slot.

C5. If a packet transmission duration overlaps with a time in the bad state, the packet is received in error. The packet is received correctly otherwise.

C6. Every packet received in error is assumed to be detected by the channel decoder.

The real-time traffic of interest is specified as follows:

R1. Four types of real-time connections are considered with the connection specification (M_i, T_i, D_i): (1) type-1 downlink $(1, 200, 300)$; (2) type-1 uplink $(1, 200, 500)$; (3) type-2 downlink $(1, 500, 600)$; and (4) type-2 uplink $(1, 500, 1100)$. Note that type-1 (or 2) downlink and uplink connections have the same M_i and T_i.

R2. Real-time connections arrive according to a Poisson process with rate λ_{rt}. A half of them are hand-off connections which arrive from adjacent cells.

R3. The lifetime of each connection i is distributed geometrically with mean T_{life} $(= 50T_i)$.

Finally, the assumptions on non-real-time traffic are as follows:

N1. Downlink (uplink) messages arrive according to a Poisson process with rate λ_d (λ_u) (messages/mini-slot). The overall message-arrival rate λ_{nrt} is defined to be $\lambda_{nrt} = \lambda_d + \lambda_u$.

N2. The downlink class II-A (II-B) messages, destined for each of N_u mobiles, arrive independently at the BS with rate $0.9\lambda_d/N_u$ ($0.1\lambda_d/N_u$).

N3. The uplink class II-A (II-B) messages arrive independently with rate $0.9\lambda_u/N_u$ ($0.1\lambda_u/N_u$) at each of N_u mobiles.

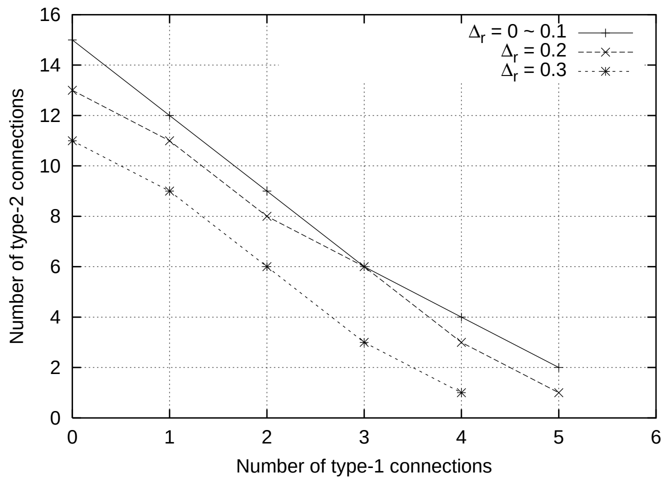


Figure 2.8: The admission region for real-time connections: $K = 20$ and $T_{req} = 200$.

N4. The number of packets in a class II-A (II-B) message is geometrically-distributed with average 2 (18).

A frequently-used parameter, the *offered load*, is defined as the overall expected number of the packets to be transmitted within a slot. For the overall non-real-time message-arrival rate λ_{nrt} (messages/mini-slot), the offered load from non-real-time traffic, L_{nrt}, is given by

$$\begin{aligned} L_{nrt} &= (0.9 \cdot 2 + 0.1 \cdot 18)K\lambda_{nrt}, \\ &= 3.6K\lambda_{nrt}, \quad \text{(packets/slot)}, \end{aligned} \tag{2.11}$$

of which a half is for class II-A and the other half is for class II-B. On the other hand, the offered load from real-time traffic, L_{rt}, is given by

$$\begin{aligned} L_{rt} &= \lambda_{rt} T_{life} E\left\{\frac{M_i T_s}{T_i}\right\}, \\ &= \lambda_{rt} T_{life} \sum_{j=1}^{2} \alpha_j \frac{\hat{M}_j T_s}{\hat{T}_j}, \quad \text{(packets/slot)}, \end{aligned} \tag{2.12}$$

where $E\{\cdot\}$ represents the averaging operation, α_j is the fraction of type-j (both downlink and uplink) connections out of the total real-time connection arrivals, $\hat{M}_j$ is the M_i of type-j connection i, and $\hat{T}_j$ is the T_i of type-j connection i.

2.5.2 Interworking of Real-Time Connections with Non-Real-Time Traffic

Throughout this subsection, the following parameters are used: $K = 20$ and $T_{req} = 200$, so there is 10 ($= K/2$) request mini-slots within a transmission-request slot. Three request

Real-time arrival rate (λ_{rt})	0.0001	0.0005	0.001
Real-time offered load (L_{rt})	0.10	0.50	1.00
Real-time throughput	0.099	0.359	0.439
Connection-blocking probability	0.0013	0.273	0.563
Average packet delay	164.55	181.76	185.09
Maximum packet delay	500	500	500
Packet-dropping probability	0.0031	0.0040	0.0042

Table 2.2: Performance of real-time traffic when non-real-time uplink traffic arrives with $\lambda_u = 0.0005$: $K = 20$ and $T_{req} = 200$.

mini-slots are reserved for hand-off requests, i.e., $K_{ho} = 3$. We first consider the admission region of type-1 and 2 connections depending on the reserved link capacity portion Δ_r for real-time packet retransmissions and non-real-time traffic. Using Eqs. (2.4) and (2.9), each pair of the numbers of type-1 and 2 connections is tested; a pair (n_1, n_2) represents a set of n_1 type-1 connections and n_2 type-2 connections. Figure 2.8 shows the curves representing the admission regions for three different Δ_r values. For each line, a set of connections for each pair under or on a curve can be admitted into the system. For example, 6 type-1 uplink connections and 2 type-2 uplink connections can co-exist in the system at a time when $\Delta_r = 0 - 0.1$. We observe that the larger Δ_r, the smaller the admission region as it should be. The admission regions for $\Delta_r = 0$ and $\Delta_r = 0.1$ are the same is due to the effect of T_{max_poll} in Eq. (2.7) for the delay-bound test. Note that the admission regions of both downlink and uplink connections are the same since both downlink and uplink connections of the same type have the same M_i and T_i.

Now, let's consider the performance of real-time traffic. For further study, we assume that $\Delta_r = 0$, and only type-1 uplink connections were generated, where each of them is a constant bit rate (CBR) connection, i.e., M_i packets are generated every T_i time units. We learned from the previous admission-region figure that up to 5 type-1 uplink connections can be admitted at a time. The reason why we consider only uplink connections is to examine the access latency of uplink connection requests, which is a very important performance measure. In the simulation, non-real-time uplink traffic arrives with rate $\lambda_u = 0.001$. Note that the performance of real-time traffic is almost unaffected by non-real-time traffic because real-time traffic is given priority over non-real-time traffic. (Some small effect can appear due to non-preemptive transmission of non-real-time traffic, and packet deferments of real-time traffic.) Table 2.2 shows the real-time offered load L_{rt}, real-time traffic throughput,

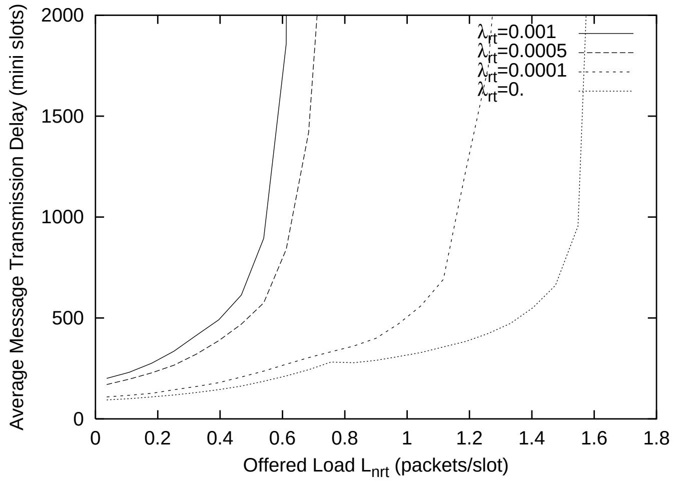


Figure 2.9: Average uplink class II-A message-transmission delay vs. non-real-time offered load: for different real-time connection-arrival rates λ_{rt}, $K = 20$ and $T_{req} = 200$.

connection-blocking probability, average and maximum packet delays, and packet-dropping probability (i.e., packet drops due to the deadline expirations) for three different arrival rates λ_{rt} of real-time connections. The blocking probability represents both new-connection blocks and hand-off connection drops since a half of the entire real-time connection arrivals are assumed to be hand-offs in our simulation. As the real-time connection arrival rate increases, more connections are blocked. We also observed that the larger λ_{rt}, the larger the average packet delay and the packet-dropping probability, since the more real-time connections exist, the longer a scheduled polling is likely to be delayed. However, the maximum delay is observed to be always 500, which is the delay bound D_i. Note that this delay is bounded at the cost of some packet drops. We can easily expect that the larger Δ_r, the larger the connection-blocking probability, and the smaller packet-dropping probability.

Figure 2.9 shows the average uplink class II-A message-transmission delay for four different real-time traffic-arrival rates. We observed that the larger λ_{rt}, the larger the delay is for a given non-real-time offered load L_{nrt}, because real-time traffic is given priority over non-real-time traffic. Note that a half of non-real-time offered load is from class II-B. This is the reason why the delay of class II-A could not be infinite even for the offered load is more than one. Even though it is not shown, the delay of class II-B is surely infinite for the offered load greater than one. The next subsection will elaborate on the performance of non-real-time communication service.

Next, we consider the request-access latency, which is defined as the time span from the arrival of a request to its successful reception by the BS. Recall that a request access

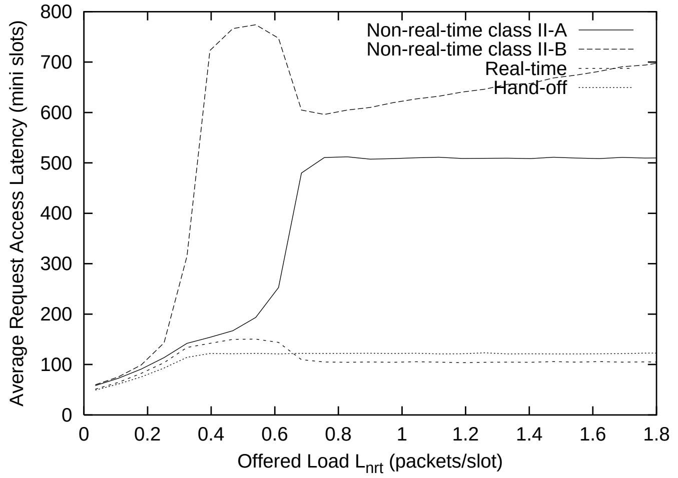


Figure 2.10: Average request-access latency vs. non-real-time offered load: $\lambda_{rt} = 0.001$, $K = 20$, and $T_{req} = 200$.

is made through the transmission request slot or piggybacking for (1) non-real-time (class II-A and II-B) message-transmission requests, (2) a new real-time connection request, and (3) a hand-off request. The request-access latencies for four different cases are plotted in Figure 2.10 when $\lambda_{rt} = 0.001$, and only uplink non-real-time traffic exists. We observe that the request-access latency for real-time and hand-off connections are reasonably low (less than 150) throughout the whole non-real-time offered load examined. Especially, the request-access latency for hand-off connections are saturated by 120 beginning $L_{nrt} \approx 0.4$. Note that the request-access latency of hand-off connections should be kept low because otherwise, the hand-off will not be smooth. To give a detailed account, we divide the entire non-real-time offered loads examined into three regions: (1) lightly-loaded region: $L_{nrt} = 0.0$ to 0.5; (2) moderately-loaded region: $L_{nrt} = 0.5$ to 0.8; and (3) heavily-loaded region: $L_{nrt} = 0.8$ to 1.8.

First, we consider the request-access latency of non-real-time traffic. In the lightly-loaded region, both the latencies of class II-A and II-B traffic increase monotonically. Class II-B traffic's access latency is larger since class II-A's requests are given priority. In the moderately-loaded region, class II-B traffic's access latency starts to decrease due to the effect of piggybacked requests. Class II-A traffic's access latency, on the other hand, continues to increase further due to the effect of piggybacked requests. That is, in this region, as the offered load increases, more requests will be made via piggybacked requests. The piggybacked request-access delay will increase since the time span between two consecutive services of a mobile will increase due to the increasing number of mobiles with pending requests in round-robin queues. In the heavily-loaded region, class II-A request-access latency is saturated at around 500 while class II-B request-access latency increases slowly beginning at around 600. In this region, non-real-time requests are likely to be made only though the piggybacked requests since the non-real-time round-robin service queues in the BS are unlikely to be empty. In this case, a mobile is supposed to have a chance to transmit two uplink packets once every 440 time units when there is no real-time traffic since there are 10 mobiles and 2 packets (plus 2 control packets) can be transmitted at each round-robin turn. So, the request-access latency would be about 220 when there is no real-time traffic. Due to real-time connections, these chances are delayed more, depending on the real-time arrival rate. It is found to be about 500 for class II-A traffic when $\lambda_{rt} = 0.001$. The request access of class II-B traffic is likely to be delayed more since only one request for a traffic class can be piggybacked in a packet.

The access latency of real-time and hand-off connection requests is found to be lower than that of non-real-time traffic over the entire offered load region. This is because real-time connection requests are given priority over non-real-time transmission requests, and K_{ho} (which is 3 in our simulations) request mini-slots are reserved for hand-off requests. For the very lightly-loaded region (say $L_{nrt} \leq 0.2$), the latencies for all types of requests are about the same since almost every request is made through a transmission-request slot and is mostly successful because collisions rarely occur. As the offered load increases, all the latencies increase while the hand-off request latency is saturated. In the moderately-loaded region, the access latency for real-time connections starts to decrease. This is again because some of transmission requests are piggybacked. Then, in the heavily-loaded region, real-time and hand-off request-access latencies are saturated at around 100 and 120, respectively. The real-time request-access latency is smaller than the hand-off request-access latency since a hand-off request can only be made through the transmission-request slot while a new connection request can be made through either a transmission-request slot or piggybacking. They are about 100 since the period of issuing the transmission-request slot $T_{req} = 200$ while they are about the same since the effect of the piggybacked request is negligible. The exact request-access latencies will depend on the real-time connection-arrival rate λ_{rt}, but the overall tendency observed remains the same irrespective of the rate.

2.5.3 Non-Real-Time Communication Service

Here we examine in detail the performance of non-real-time communication service. To understand the non-real-time communication service better, we generated only non-real-

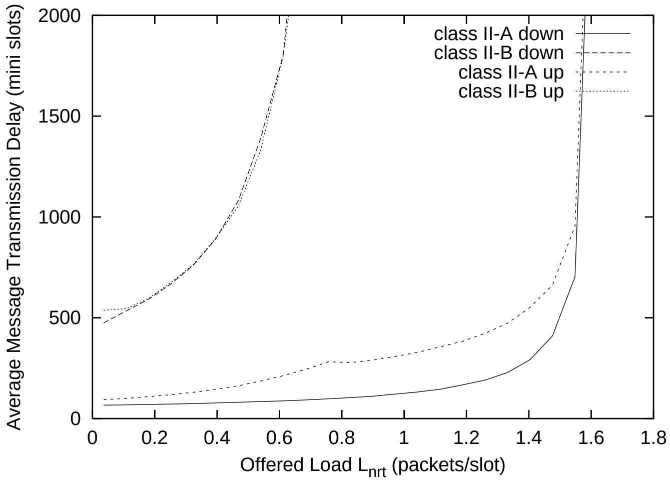


Figure 2.11: Average message-transmission delay vs. non-real-time offered load when downlink or uplink traffic exists: $K = 20$ and $T_{req} = 200$.

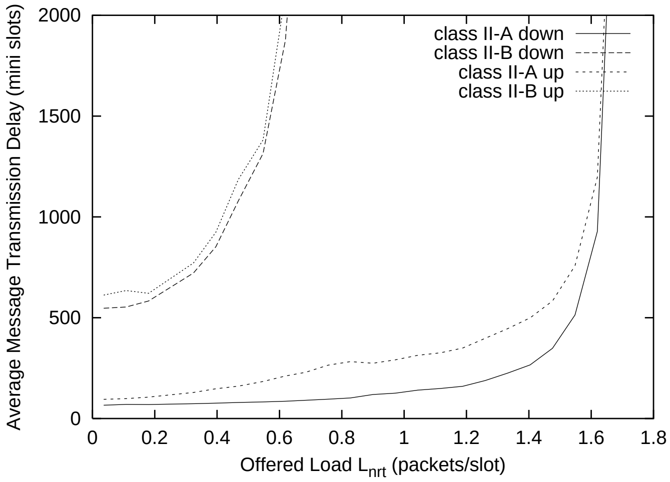


Figure 2.12: Average message-transmission delay vs. non-real-time offered load when downlink and uplink traffic arrive with the same rate, i.e., $\lambda_d = \lambda_u$: $K = 20$ and $T_{req} = 200$.

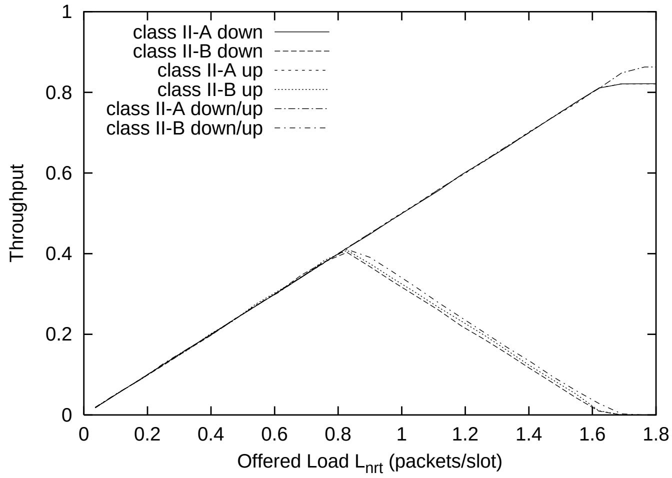


Figure 2.13: Throughput vs. non-real-time offered load: $K = 20$ and $T_{req} = 200$.

time traffic for the simulations, i.e., $\lambda_{rt} = 0.$. Figure 2.11 shows the average message-transmission delays for both class II-A and II-B for two extreme cases: one with downlink traffic arrivals only (marked with 'down') and the other with uplink traffic arrivals only (marked with 'up'). We observe that the delays with downlink traffic arrivals are a bit smaller than those with uplink traffic arrivals. This indicates the effect of the uplink transmission-request access latency. For both cases, class II-B's delay goes to infinity at around $L_{nrt} = 0.7$ while class II-A's delay goes to infinity at around $L_{nrt} = 1.6$. This means that class II-B traffic is rarely transmitted for $L > 0.7$. This is reasonable since a half of the offered load is from class II-B, and the maximum achievable throughput is less than one. The fact that class II-A's delay goes to infinity at around $L_{nrt} = 1.6$ means that the maximum achievable throughput is about 0.8. Figure 2.12 shows the average message-transmission delays for both class II-A and II-B when a half of arrived messages are downlink and the other half are uplink, i.e., $\lambda_d = \lambda_u$. The general tendency is the same as in the previous figure, but the maximum achievable throughput is found to increase since the delay of class II-A is still about 2000 (i.e., within the figure) even for $L_{nrt} = 1.65$. This shows the effect of sequential transmission of a downlink packet and an uplink packet without any control packet at a mobile's turn in the BS round-robin queue.

These facts can be seen clearer in Figure 2.13. First, beginning at $L_{nrt} = 0.8$, the throughput of class II-B starts to decrease while that of class II-A continues to increase. When the maximum achievable throughput is reached, the throughput is saturated there, i.e., about 0.8 for both only downlink (marked with 'down') and only uplink (marked with 'up') arrival cases. Note that the throughputs for both downlink and uplink arrival case (marked with 'down/up') for a given offered load are larger than, or equal to, those for the other two extreme cases. The maximum achievable throughput for the latter case is found to be about 0.86. The reason why the maximum achievable throughput is less than one is due to the overhead of transmission-request slots, polling mini-slots, ACK/NAK-transmitting mini-slots, and channel probing processes.

So far, we used the value of $T_{req} = 200$, i.e., the average interval between two transmission-request slots is about 200. Figure 2.14 shows the average transmission delays of uplink class II-A messages for three different T_{req} values. Note that the larger T_{req}, the larger the maximum achievable throughput could be. The delay is found to increase abruptly at around $L_{nrt} = 0.8$ for a large T_{req}. This is because the larger T_{req}, the more chance of a mobile having a message to request upon appearance of a transmission-request slot. So, the larger T_{req}, the more likely a request is to collide with others, and the larger the delay eventually. This effect is lessened dramatically after $L_{nrt} = 0.8$ due to the increases of piggybacked requests. Even though it is not shown here, the access latency of real-time connection requests also has a similar abrupt increase at some regions of offered load with large T_{req}. This observation implies that too large a T_{req} is not desirable.

Figure 2.15 shows the effect of K values when $T_{req} = 600$. As K increases, the peak delay at $L_{nrt} = 0.8$ is found, on average, to decrease while the delay increases. A large K means a large packet size and an increased number of request mini-slots in a transmission-request slot. The decreased peak delay is due to the second fact, and the increased average delay is due to the first fact. Note that too large a K is not good even though the peak delay is reduced because a large fixed-size packet is likely to be transmitted with some empty portion, thus lowering the system utilization. From the above two figures, one can conclude that the choice of K and T_{req} is closely related to each other, and hence, they should be considered together so that there is no undesirable peak in the average delay.

2.6 Conclusion

This chapter considered how to support both real-time and non-real-time communication services in a WLAN. We proposed a MAC protocol based on polling mobiles for their uplink accesses with slotted ALOHA-based and piggybacked transmission requests. A variation of EDF scheduling is used for real-time traffic to bound delivery-delays at the cost of some packet losses depending on the channel conditions while a variation of round-robin scheduling is used for non-real-time traffic to achieve error-free transmissions and long-term

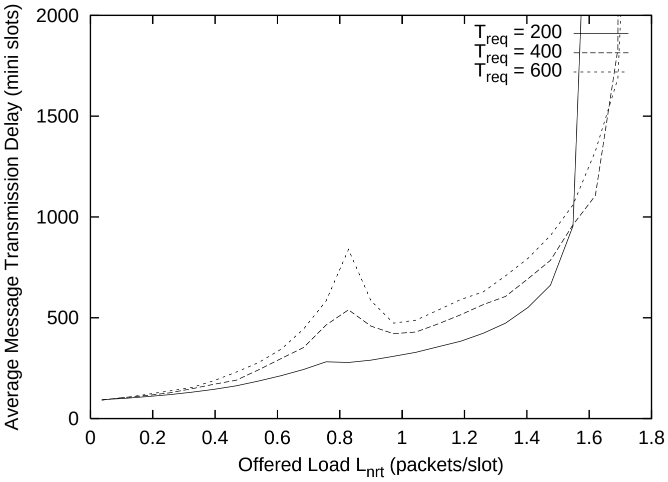


Figure 2.14: Average uplink class II-A message transmission delay vs. non-real-time offered load: with different values of T_{req}: $K = 20$.

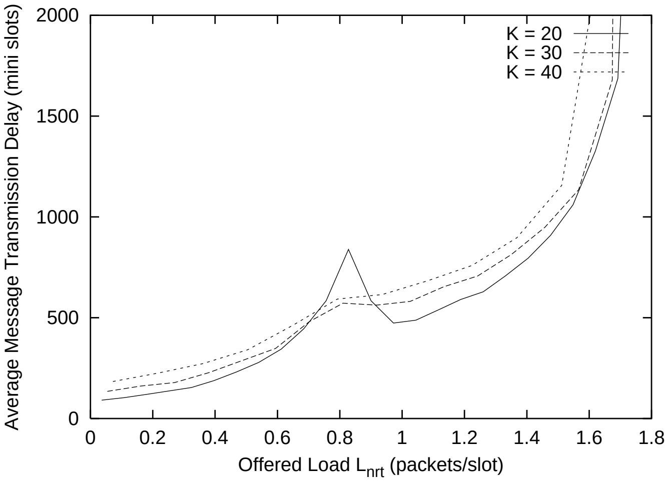


Figure 2.15: Average uplink class II-A message-transmission delay vs. non-real-time offered load: for $T_{req} = 600$ and with different values of K.

fairness among mobiles. Also addressed are the problems of (1) multiplexing downlink and uplink traffic for D-TDD transmission, and (2) handling location-dependent, time-varying, and bursty channel errors. An admission test for each new real-time connection request has been established, determining if the requested packet delivery-delay bound can be guaranteed without violating the existing guarantees.

An extensive evaluation study has shown the proposed scheme to work well and meet the design goals. We first considered the admission regions of real-time connections for four types of connections, and the average performance of real-time traffic. We also examined the inter-working of real-time and non-real-time traffic while emphasizing the access delays of non-real-time messages, real-time connection requests, and hand-off requests. Finally, the performance of non-real-time communication service was evaluated in detail.

CHAPTER 3

A QOS-SUPPORTING CDMA SYSTEM

3.1 Introduction

We consider a multi-code (MC) CDMA uplink system to provide three types of packet-level QoS for heterogeneous traffic according to the application requirements: (1) bounded packet-delivery delay; (2) guaranteed transmission rate; (3) packet error probability (or bit error probability, BER). For this system, we consider heterogeneous traffic and its required error performance more specifically as in Table 3.1. Note that voice and video fall into this real-time traffic, although video requires better error performance and higher rate than voice. Currently, low-rate voice coding schemes like QCELP and VSELP are being used to deliver speech over a wireless link at a rate of 8 Kbps or less [78]. There has also been significant research on low-rate video coding techniques to transmit video at a rate lower than 64 Kbps including the new ITU-T recommendation H.263 [89].

For class-I real-time traffic, connection-oriented transmission is achieved by assigning a set of mobile-oriented *code channels* and a corresponding receiver, like assigning circuits in a circuit-switched network, to each corresponding mobile after it passes the admission test. (One can imagine a pseudo-noise (PN) code as a channel through which information can be transmitted, hence called a *code channel*. Assigning a code channel to a mobile means that the BS gives the mobile permission to use a particular PN code, and gets ready to receive packets by tuning a receiver to the PN code.) On the other hand, class-II non-real-time traffic does not require bounded delays nor guaranteed rates. Moreover, it is likely to be bursty and randomly-arriving, so it is inefficient to assign a mobile-oriented code channel and a receiver to each mobile for class-II transmissions. We, therefore, adopt a transmission-rate request access scheme, similar, in principle, to the Distributed-Queueing Request Update Multiple Access (DQRUMA) [52, 63] and our transmission request-based

	Class I		Class II-A	Class II-B
Examples	voice	video	remote login	e-mail
Delay	bounded		sensitive	tolerable
Rate	guaranteed		not guaranteed	
	0-8 Kbps	0-64 Kbps	8-128Kbps	
BER	$\leq 10^{-3}$	$\leq 10^{-5}$	≈ 0	

Table 3.1: Traffic classification for the proposed CDMA system.

MAC protocol for non-real-time traffic in Chapter 2, for class-II traffic, in which a mobile, upon generation of a class-II message, requests its transmission in a transmission-request mini-slot, according to a link access scheme similar to the slotted ALOHA, and the BS permits the transmission according to some scheduling principle. The transmission-request scheme is also used for setting up new class-I connections. The third QoS related to error performance is supported by the combination of power control, error control codes, and ARQ scheme (only for class II). An FEC scheme is used for class I due to the uncertain delivery delays of ARQ schemes.

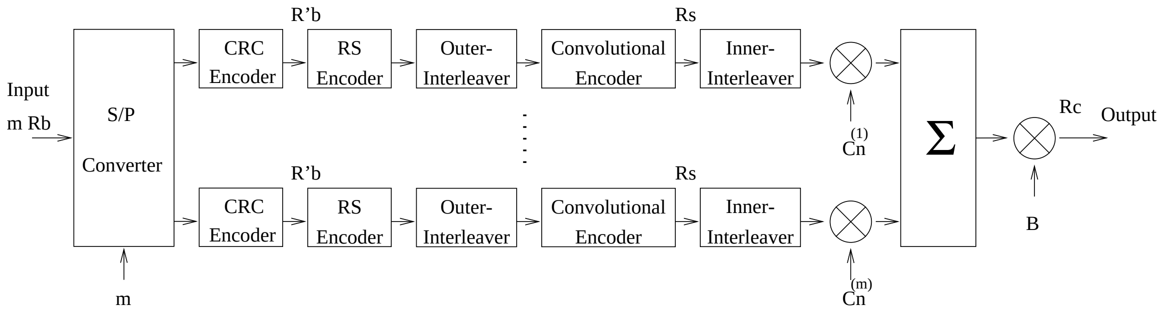
This chapter is organized as follows. Section 3.2 describes the CDMA system specification, the assumed fading model, as well as the traffic models under consideration. The communication protocols for both class I and II traffic are presented in Section 3.3. The error-control schemes and the associated equations are given in Section 3.4. Section 3.5 describes run-time control schemes of the BS including the allocation of a power level to uplink accesses, admission control of class-I connection requests, and scheduling of class-II packet-transmission requests. In Section 3.6, we present both the analytical and simulation results. Finally, the chapter concludes with Section 3.7.

3.2 Modeling and Specifications

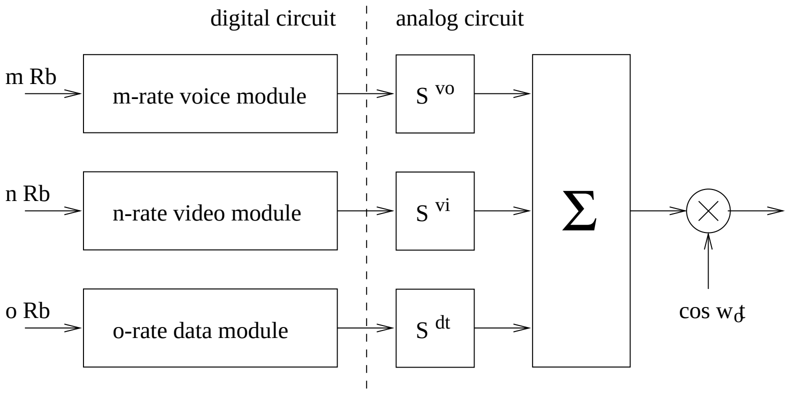
We now describe the specification of the MC-CDMA transmitter which is equipped in each mobile and the Rayleigh fading channel assumed, and also present the class-I traffic models, which are essential to establish the admission test of class-I connections.

3.2.1 MC-CDMA Transmitter

Only the uplink is considered; the downlink can be supported similarly but more easily, thanks to the broadcast nature of the downlink. Figure 3.1 shows the block diagram of



(a) An m-rate transmitter module



(b) The integrated transmitter

Figure 3.1: Block diagram for an MC-CDMA transmitter: (a) an m-rate transmitter module with RS/convolutional concatenated encoders and interleavers, where $R_b' = R_b(Kb)/(Kb - H)$; (b) the integrated transmitter, where $m + n + o \leq M$.

a transmitter equipped in each mobile. The system is based on the MC-CDMA [46, 47] and the Reed-Solomon/convolutional concatenated coding scheme [3, 23]. Each of these components is described below in detail.

A mobile can transmit its packets at integer multiples of the *basic rate* R_b up to $M \cdot R_b$. We assume $R_b = 8$ Kbps, which is equivalent to the rate for a voice transmission, and $M = 16$ to make $M \cdot R_b = 128$ Kbps, the maximum data transmission rate. We start with an m-rate[1] transmitter module, shown in Figure 3.1 (a), which accepts an incoming message stream at rate mR_b. First, the stream is converted into m basic-rate streams. Each basic-rate stream is encoded by cyclic redundancy check (CRC) encoders, and then by (N, K, q) Reed-Solomon (RS) encoders. The RS encoder maps K symbols into N symbols each of which consists of a b-binary sequence, where $b = \log_2 q$. An RS codeword corresponds to

[1] m-rate means 'rate mR_b' throughout this chapter.

a packet, and accommodates $K \cdot b - H$ information bits assuming that an H-bit header (including CRC) is added in each packet. A header is appended to all m basic-rate streams once every $K \cdot b - H$ bits to form packets in the serial-to-parallel converter. Note that the RS encoder's input rate is $R_b' = R_b(Kb)/(Kb - H)$. The output of the RS encoder is interleaved by an outer-interleaver on a symbol-by-symbol basis, then encoded by a rate $1/n$ convolutional code with memory size M_{mem}. The output of the convolutional encoder is interleaved by an inner-interleaver on a bit-by-bit basis. The inner-interleaver renders the fading channel memoryless while the outer-interleaver randomizes burst errors at the output of the convolutional (Viterbi) decoder.

When a mobile enters the cell under consideration,[2] it is assigned a primary pseudo-noise (PN) code of chip rate R_c, which is used exclusively by the mobile as long as it remains within the cell. Each of m streams is multiplied by one of orthogonal codes. The set of orthogonal codes is generated by a sub-code concatenation scheme: when C_n^{PN} is the primary PN code of a mobile n, the spreading codes $\{C_n^{(i)}, i = 1, 2, \cdots, M\}$ are obtained by

$$C_n^{(i)} = C_n^{PN} \times D_i, \quad D_i \perp D_j, \quad i \neq j, \tag{3.1}$$

where D_i's are from a set of Walsh (i.e., orthogonal) codes. Note that $C_n^{(i)} \perp C_n^{(j)}, i \neq j$, is also guaranteed. Using this sub-code concatenation, it was claimed that there was zero interference across the received orthogonal codes so that multi-rate transmission can be achieved with a smaller power than the basic rate transmission [46, 47, 63, 16]. However, this is not the case when the orthogonal codes are transmitted over a multi-path fading channel which is typical in wireless cellular networks.

In fact, multi-rate transmission can still benefit from the sub-code concatenation since, among the various delayed versions of codes, those arriving simultaneously at the BS will not work as interferences to each other. Especially, using the RAKE receiver [83] which can resolve multiple delayed versions of a signal transmitted with a resolution of chip duration T_c $(= 1/R_c)$, this can really happen. Depending on the underlying wireless channel characteristics and the receiver techniques, this benefit could be significant or very slim; for example, if (1) there is a line-of-sight transmission path between a mobile and the BS, and (2) multi-path versions of the transmitted codes are relatively weak, there will be virtually no self-interference. In this chapter, our CDMA system is assumed not to benefit from self-code concatenation due to multi-path fading (in order to err on the safe side). In the system designed under this assumption, higher-rate transmission can, in practice, have better error

[2] Physically, when the user turns on the switch of a handset or the mobile is handed off from an adjacent cell.

performance than lower-rate transmission due to (possibly) lower interferences. After being spread by orthogonal codes, all parallel streams are summed together, then multiplied by a base station code, which is used to indicate which BS the mobile is communicating with.

Each mobile can transmit voice, video, and data traffic at the same time as long as the total transmission rate does not exceed M. As shown in Figure 3.1 (b), for each traffic type, a separate transmitter module is used because each traffic type requires a different transmission power level due to its different BER requirement (to be described later). Note that when transmitter modules are in use, orthogonal codes $C_n^{(i)}$ from the same set of M codes will be used. The blocks S^{vo}, S^{vi}, and S^{dt} represent the analog amplifiers with those values of power level for voice, video, and data traffic, respectively. Note that the blocks before the amplifiers are implemented with a digital circuit, while the others are implemented with an analog circuit. All the amplified outputs of modules are summed together, multiplied by the carrier frequency, then transmitted.

3.2.2 Fading Channel and Power Control

The channel model assumed in this chapter is a frequency non-selective and slowly-fading channel with Rayleigh-distributed envelope statistics. The frequency non-selective channel results in multiplicative distortion of the transmitted signal at the BS. Furthermore, the condition that the channel fades slowly implies that the multiplicative process may be regarded as a constant during at least one signaling interval [83]. In fact, this is one of the simplest among the known fading channel models. Depending on the actual channel model and the receiver techniques used (e.g., with or without a RAKE receiver), actual performance may vary, but the general performance trends, and the system architecture considered in this chapter are still valid irrespective of the assumed channel model and the receiver techniques.

A wireless channel, in reality, is subject to (1) the long-term propagation loss due to the distance from mobiles to the BS and (2) the shadowing effects of buildings and other objects in addition to the above-mentioned short-term fading. We assume that both can be adjusted through power control, which is the case in real CDMA systems [99]. Both open- and closed-loop power-control mechanisms will be used to control the received power at the BS. Time-averaged versions of the received power are used for closed-loop power control, and hence the short-term fading is not mitigated by power control. However, we assume that the *average* received signal-to-interference ratio (and hence, the average received signal power) at the BS can be maintained according to a target value assigned to the traffic type at a given traffic load. How to control the transmission power itself is out of the scope of this chapter. We only design the mechanism to determine and assign the target power level for each traffic type depending on a given traffic load.

3.2.3 Class-I Traffic Modeling

In our proposed scheme, class-I traffic is handled by assigning specific code channels to mobiles similarly to assigning frequency slots in FDMA, time slots in TDMA, and more generally, circuits in a circuit-switched network. However, the underlying concept is totally different because in CDMA systems, all mobiles share the entire frequency and time slots. Depending on their traffic characteristics, we can admit many more mobiles simultaneously into the system than the physical link capacity without violating the required BER because the CDMA system is interference-limited. In a sense, the CDMA system achieves statistical multiplexing. However, to achieve the required error performance, we need to control the admission of class-I traffic which, in turn, requires appropriate traffic modeling. Class-II traffic does not affect the admission control of class-I traffic because the transmission of each class-II packet is permitted by the BS, depending on the available bandwidth left unused by class-I traffic in our protocol.

It is well-known that voice traffic is characterized by an "on-off" model since a speech signal is in either *talk-spurt* or *silent* mode during the conversation of a mobile [8]. Each voice mobile is modeled by a two-state Markov chain where one state represents ON (transmitting at rate R_b) and the other represents OFF (idle). The utilization ρ_{vo} (i.e., fraction of time during which the voice mobile is active) is assumed to be 0.5. Given the number K_{vo} of voice mobiles, the number K_{vo}^a of active voice mobiles at current time can be computed by a binomial distribution:

$$P_{K_{vo}^a}(k) = \binom{K_{vo}}{k}(0.5)^{K_{vo}}. \tag{3.2}$$

Since there does not exist any proper model for low-rate video coding, we adopt a very simple model to represent the variable bit rate (VBR) characteristics of video traffic: each video mobile is modeled by a three-state Markov chain where in the first state, the mobile transmits packets at the rate of $4R_b$; in the second, at the rate $6R_b$; and in the third, at the rate $8R_b$ ($=64$ Kbps). The mobile will reside in each of the three states with probability 0.25, 0.5, and 0.25, respectively. Given the number K_{vi} of video mobiles, the number $K_{vi}^a(j)$ of j-rate video mobiles is given by a 3rd-order multinomial distribution:

$$P_{K_{vi}^a(4),K_{vi}^a(6),K_{vi}^a(8)}(k_4, k_6, k_8) =$$

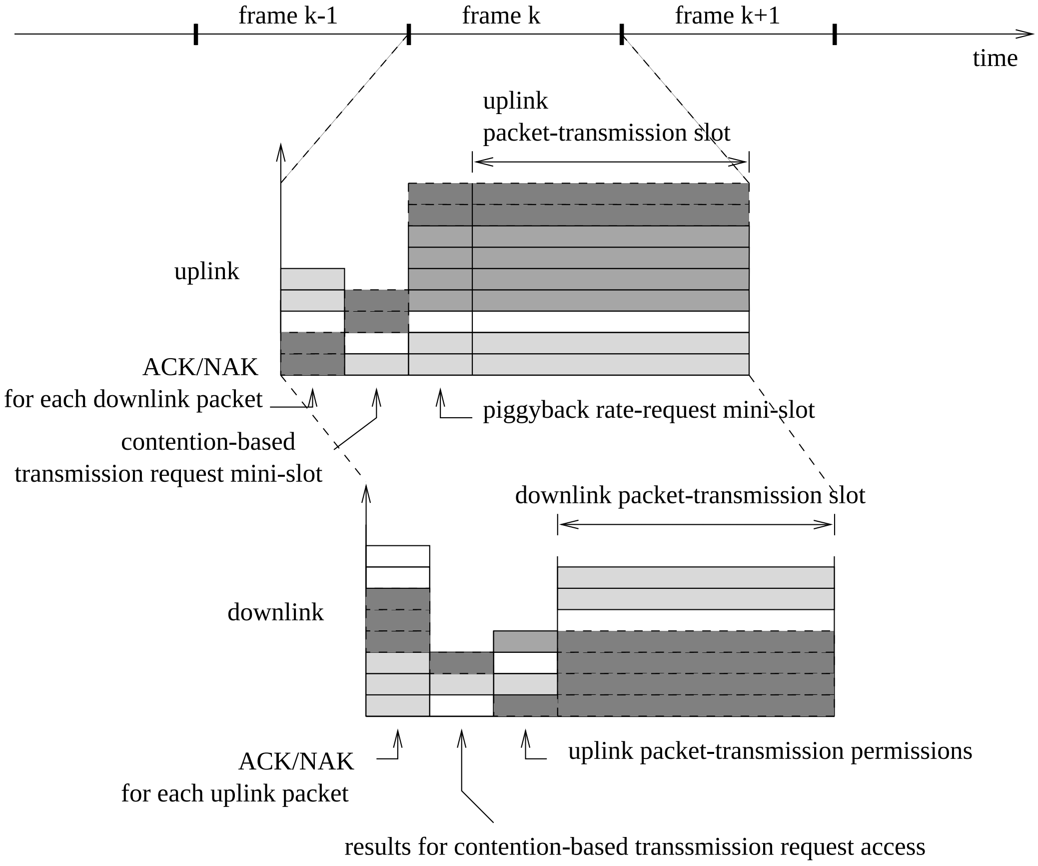


Figure 3.2: The uplink and downlink frames along the time-axis.

$$\binom{K_{vi}}{k_4, k_6, k_8}(0.5)^{k_6}(0.25)^{k_4+k_8}, \tag{3.3}$$

where $k_4 + k_6 + k_8 = K_{vi}$. For other video traffic models, this distribution can still be obtained if the rate distribution is given for a model used.

3.3 Protocol Description

Connection-oriented transmission is achieved by assigning a set of mobile-oriented code channels for each class-I mobile. On the other hand, whenever a mobile has class-II messages to transmit, it requests code channels via a BS-oriented transmission-request code channel (or a piggyback rate-request code channel), then, according to the BS's scheduling policy, the transmission is scheduled and granted. Note that for a mobile to send a packet through a code channel, the BS should be ready to receive it by tuning a receiver to the corresponding PN code. As shown in Figure 3.2, our scheme is based on the uplink and downlink frame formats, which are drawn basically from [63]. The uplink time-axis is divided into frames which consist of a mini-slot for ACK/NAK, a contention-based transmission-request mini-slot, a piggyback rate-request mini-slot, and a packet-transmission slot. On the other hand, the downlink time-axis is divided into frames which consist of a mini-slot for ACK/NAK transmission, a mini-slot for result announcements of contention-based transmission-requests, a mini-slot for uplink packet-transmission permissions, and a slot for downlink packet transmission. In the figure, the rectangles arranged vertically in a slot or mini-slot correspond to different packet or transmission request or ACK/NAK transmissions (with different PN codes in most cases except for the contention-based transmission request mini-slot.) Basically, rectangles with the same shade represent transmissions from (to) the same mobile if they are for uplink (downlink) except for the contention-based transmission request mini-slot. The role of each slot and mini-slot will be explained below in detail.

3.3.1 Contention-Based Transmission-Request Accesses

A set of BS-oriented code channels in each contention-based transmission-request mini-slot are used for class-II transmission-request access: a finite number, say, N_{rcvr}^{rqst}, of receivers (and hence code channels) are dedicated for mobiles' transmission-request accesses. An ALOHA-like access protocol is used for the request accesses. This contention-based request is basically used by a mobile which is not permitted to transmit any packet during the next frame and falls into one of the following three cases: (1) when the mobile wants to establish a new class-I connection; (2) when it generated new class-II messages after transmitting all messages for which it previously requested; and (3) when it wishes to transmit class-II messages at a higher rate than the rate it previously requested due to the generation of new messages. Such a mobile waits for the next contention-based transmission request mini-slot then decides whether to send the request or not with probability p_r (i.e., in a p_r-persistent manner). If it decided to send, the mobile chooses one of N_{rcvr}^{rqst} code channels, and transmits the request via the code channel in the mini-slot. Each request includes the relevant information such as mobile ID, class type (voice, video, class II-A, or class II-B), and the required transmission rate, up to MR_b (or equivalently, the number of packets, up to M packets, to be transmitted in the next frame) if it is for class-II traffic. The packets to be retransmitted by the ARQ scheme will not be included in the calculation of the required transmission rate.

A request can be received erroneously by the BS for the following three reasons. First, the total interference level exceeds the link capacity[3] and many of the requests in the

[3] Due to the CDMA's soft capacity characteristic, the request might be received correctly even under an

mini-slot may be received with error. Second, the request collided with another request transmitted in the same request code channel, and only the collided requests are received with error.[4] Third, a request may be received erroneously due to random noise and interference. In Figure 3.2, we see there are four transmission requests in frame k. Two upper requests with the same shade are found to have used the same code channel, so will collide with each other. Only (lower) two requests out of four can be successful. Only successful requests are broadcast to the mobiles through the downlink channel before the next contention-based transmission request mini-slot. The value of p_r is determined by each mobile in a distributed fashion. For the first attempt of a message transmission request, p_r is set to 1. Upon failure of a request, however, the value of p_r is decreased to

$$p_r := \frac{p_r}{p_r + 1}, \tag{3.4}$$

following $1, 1/2, 1/3, \cdots$ (i.e., the "harmonic" backoff) [52]. Each failed request should be retransmitted in a p_r-persistent manner until it is successfully received, and a mobile with a failed request is said to be *backlogged.*

Note that the choice of the slotted ALOHA with the harmonic backoff for the contention-based request access is not mandatory. We could use other access techniques such as the Binary Stack Algorithm [11, 52] or the slotted ALOHA with other backoffs such as the binary exponential backoff as used for the IEEE 802.3 Ethernet. Depending on the request access method, actual performance will vary. However, in our system, this contention-based access is used for request access only. Moreover, the BS can control the number N_{rcvr}^{rqst} of receivers for request access depending on the observed request access delay so that the delay would not be very large. So, the actual effect of choosing a request access method could be kept minimal. We adopted the slotted ALOHA with the harmonic backoff for simplicity.

3.3.2 Collision-Free Piggyback Class-II Rate Requests

When a mobile is scheduled and permitted to transmit class-II packets in the packet-transmission slot of the subsequent frame, it is supposed to update the required transmission rate (for the next scheduled transmission-slot of that mobile) through the piggyback rate-request mini-slot. These piggyback requests are collision-free because, during the piggyback mini-slots, mobiles use code channels dedicated to the mobiles in the subsequent transmission slot. Through the piggyback mini-slot, the currently-transmitting mobile can request

over-threshold situation.

[4] Transmission by different users on the same code could not collide if there is a relative delay of a few chips between the signals arriving at the receiver if PN codes with good auto-correlation property are used.

a transmission rate including the rate of newly-generated messages without involving the contention-based request accesses, thus reducing the average request access delay. A class-I mobile can also request the transmission of newly-generated class-II messages through the piggyback rate-request mini-slot, so class-I mobiles do not need the contention-based transmission-request accesses. A mobile which has voice and/or video connections can request the transmission of class-II data up to rate $15R_b$ (with a voice connection), $8R_b$ (with a video connection), and $7R_b$ (with both voice and video connections).

Each piggyback request includes such information as the class type (voice, video, class II-A or II-B) and the requested transmission rate. As shown in Figure 3.2, as many piggyback requests as the number of packet transmissions permitted for a mobile during a frame can be transmitted within a piggyback mini-slot. However, up to four piggyback requests will be transmitted: one for each class type.

3.3.3 Class-I Connection Establishment

The required QoS for class-I traffic is the bounded delivery delay of packet transmission. To this end, the BS will set up a class-I connection for each class-I transmission request by assigning a set of mobile-oriented code channels to the corresponding mobile. An important matter here is that the class-I connection request should be delivered to the BS as fast as possible. In our system, there are two ways to transmit requests to the BS, and both class I and II share these. To reduce the access delay of a class-I connection request, (1) a class-I connection is requested using either of two ways whichever becomes available to the mobile earlier. Note that a mobile already in the BS queue cannot make a new class-II transmission request using the contention-based request access. (2) Priority is given to class-I connection requests over class-II transmission requests. For example, a mobile which has both requests and not yet in the BS queue will transmit the class-I connection request only during a transmission-request mini-slot.

Upon receiving a request to establish a class-I connection in a mini-slot, the BS performs an admission test on the request according to the link capacity. The admission test checks if the total required bandwidth for class-I traffic (or *equivalent bandwidth* BW_{eq} in Section 3.5) is less than, or equal to, the link capacity C. If the test result is positive, the BS admits the mobile via a transmission-permission mini-slot, and assigns a receiver to the corresponding mobile while the receiver tunes to the mobile's code channels using the mobile's primary code and the corresponding orthogonal codes, then the mobile continues to transmit packets whenever it wants. If the test result is negative, the BS declares *connection blocking*, and notifies it to the mobile.

An admitted mobile will transmit its packets in each packet-transmission slot with the power level controlled by the BS, depending on the traffic condition. The transmission rate (or the number of packets transmitted in a slot) can vary according to the amount of traffic generated. Because each class-I mobile can transmit packets at a rate up to the contracted value (i.e., R_b for voice and $8R_b$ for video) after its admission into the system, there is no queueing delay, so the delivery delay is bounded, i.e., the first QoS is guaranteed.

3.3.4 Class-II Message Transmission

Each of successful transmission/rate requests is queued in one of two base station service queues, i.e., one for class II-A and the other for class II-B. The BS serves (i.e., gives the transmission permission to) the requests according to a transmission scheduling policy with the bandwidth available for class-II, BW_{dt} $(= C - BW_{eq})$, which depends on the admitted pair (K_{vo}, K_{vi}) of voice and video mobiles. The relation between BW_{dt} and the transmission scheduling of class-II traffic is detailed in Section 3.5.3. Transmission permissions for the subsequent frame are announced to mobiles in the middle of each slot via the downlink as shown in Figure 3.2. After sending the transmission permissions, the BS assigns a number of receivers, so that they tune to the scheduled mobiles' code channels. Each permission includes such information as mobile ID and transmission rate.

Since class-II traffic uses a selective-repeat ARQ scheme, the feedback from the BS should be announced. For each transmission of packets, ACK (acknowledge) or NAK (non-acknowledge) is fed back. NAKed packets are retransmitted with priority over other packets in the next frame during which the mobile is permitted to send packets; if a mobile has NAKed packets, only after retransmitting all of them, it will transmit other newly-generated packets. An erroneous packet transmitted in the i-th frame can be retransmitted as early as in the $(i+2)$-th frame because the corresponding NAK is fed back to the mobile during the $(i+1)$-th frame.

3.4 Error Performance

As described in Section 3.2, the RS/convolutional concatenated coding is adopted to protect data from the errors caused by the Rayleigh fading and additive white Gaussian noise (AWGN). A dual-mode RS decoder is used; for class-I traffic, an error-correction-only decoder is used while for class-II, an error-correction-and-detection decoder is used for retransmitting corrupted packets with selective-repeat ARQ.

3.4.1 Demodulator and Convolutional Decoder

The bit error probability at the output of the convolutional decoder can be upper-bounded [83] by:

$$P_b \leq \sum_{d=d_{free}}^{\infty} b_d P_d, \tag{3.5}$$

where d_{free} is the free distance of the convolutional code, b_d is the total number of nonzero information bits on all weight-d paths in the trellis diagram of the convolutional code, and P_d is the pairwise error probability, which is, the probability that an incorrect path at distance d from the correct path being chosen by the Viterbi decoder. When the binary phase shift keying (BPSK) demodulator and soft decision decoding is applied, $P_d(\gamma_{o,i}, i = 1, \cdots, d)$ for a given set of the signal-to-noise ratio (SIR) $\gamma_{o,i}$ of the i-th error symbol in the incorrect path is represented by

$$P_d(\gamma_{o,i}, i = 1, \cdots, d) = Q\left(\sqrt{2\sum_{i=1}^{d} \gamma_{o,i}}\right) \leq \frac{1}{2}\prod_{i=1}^{d} e^{-\gamma_{o,i}}. \tag{3.6}$$

By averaging Eq. (3.6) over $\gamma_{o,i}$'s which are exponentially distributed in a Rayleigh fading environment, we obtain

$$\begin{aligned} P_d &\leq \frac{1}{2}E\left[\prod_{i=1}^{d} e^{-\gamma_{o,i}}\right], && (3.7)\\ &= \frac{1}{2}\left(E\left[e^{-\gamma_o}\right]\right)^d, && (3.8)\\ &= \frac{1}{2}\left(\frac{1}{1+\overline{\gamma}_o}\right)^d, && (3.9) \end{aligned}$$

where Eq. (3.8) is obtained since $\gamma_{o,i}$'s are independent due to the ideal inner-(de)interleaver, and Eq. (3.9) is obtained by using the pdf of exponentially-distributed γ_o with the average SIR $\overline{\gamma}_o$. Now, when the BPSK demodulator, an ideal inner-(de)interleaver, and soft decision convolutional decoding are used, the upper bound of the average bit error probability at the output of the convolutional decoder is given by

$$P_b \leq \frac{1}{2}\sum_{d=d_{free}}^{\infty} b_d \left(\frac{1}{1+\overline{\gamma}_o}\right)^d. \tag{3.10}$$

3.4.2 Reed-Solomon and CRC Decoders

We adopt a bounded distance RS decoder of the (N, K, q) RS code over GF(q) with the error correction capability t $(= \lfloor (N-K)/2 \rfloor)$. That is, the decoder looks for a codeword within distance t of the received word; if there is such a codeword, the decoder finds it, and if not, the decoder declares "decoding failure." A q-ary symbol is mapped to b bits, so $q = 2^b$. If P_s denotes the symbol error rate at the input of the RS decoder, then the probability P_T of the total error of the RS decoder which includes both decoding error (i.e., decoding into a wrong codeword) and decoding failure (i.e., inability to decode) is given by

$$P_T = \sum_{i=t+1}^{N} \binom{N}{i} P_s^i (1 - P_s)^{N-i}, \tag{3.11}$$

where $t = \lfloor (n-k)/2 \rfloor$ is the RS error-correction capability, and P_s can be upper-bounded by

$$P_s \leq bP_b, \tag{3.12}$$

with the bit error probability P_b of the Viterbi decoder given in Eq. (3.10). Now, the probability P_E of RS decoding error and the probability P_F of RS decoding failure are given [69] by:

$$P_E \leq P_T \sum_{i=0}^{t} \binom{N}{i} (2^b - 1)^{i-(N-K)}, \tag{3.13}$$

$$P_F = P_T - P_E. \tag{3.14}$$

When a decoding error happens with probability P_E, the errors in the received word are not detected by the RS decoder. In most cases, these undetected errors are detected by the CRC decoder, which is located after the RS decoder. Through two-level error detection by a combination of the RS decoder in the error-correction-and-detection mode and the CRC decoder, virtually all uncorrectable errors can be detected.

A detected error in a class-II packet will be informed to the mobile via the downlink ACK/NAK-announcing mini-slot, and will trigger the retransmission of the packet by the selective-repeat ARQ. Note that the packet retransmission probability P_R can be approximated by the total error probability, i.e.,

$$P_R \approx P_T, \tag{3.15}$$

and, the bit error probability at the output of the CRC decoder for class-II traffic will be virtually zero, i.e.,

$$P_{eb,II} \approx 0. \tag{3.16}$$

Now, when the RS decoder is used for class-I traffic, the decoder will be switched to the error-correction-only mode, in which, instead of declaring a decoding failure, the decoder will just bypass the received word to the output of the decoder. (The CRC decoder will do the same.) Without loss of the generality, we derive the probability $P_{eb,I}^1$ that the first bit of the received word is in error at the output of the RS decoder in the error-correction-only mode in order to obtain the bit error probability $P_{eb,I}$ at the output of the CRC decoder for class-I traffic as follows:

$$\begin{aligned}
P_{eb,I} &= P_{eb,I}^1, \\
&= Pr(\text{1st bit error } \& \ i \geq t+1 \text{ errors out of } N \text{ symbols}), \\
&= Pr(\text{1st bit error}) \cdot Pr(i \geq t+1 \text{ errors out of } N \text{ symbols}|\text{1st bit error}), \\
&= P_b \cdot Pr(i \geq t \text{ errors out of the last } N-1 \text{ symbols}), \\
&= P_b \cdot \sum_{i \geq t}^{N-1} \binom{N-1}{i} P_s^i (1-P_s)^{N-1-i}, \\
&= P_b \cdot \sum_{i \geq t+1}^{N} \frac{i}{N} \binom{N}{i} P_s^{i-1} (1-P_s)^{N-i}.
\end{aligned} \tag{3.17}$$

3.5 Run-Time Control Schemes

At run-time, the BS exercises the following three types of control: (1) allocation of a power level for each packet transmission; (2) admission control of each newly-requested class-I connection; and (3) class-II traffic transmission-permission control. We use a *unit of bandwidth* or a BU which is the amount of bandwidth to support an active voice mobile.

3.5.1 Power-Level Allocation

Let's consider how to set the target SIR for each traffic class while meeting different error-performance requirements for different traffic classes. Given modulation and coding schemes, the BER of class-I traffic is determined directly from the received average SIR as shown in Eq. (3.17). So, for each class-I traffic, a *target* SIR of the convolutionally-coded symbol received is assigned according to its required BER. On the other hand, a virtually error-free class-II communication is achieved via error detection and retransmission basically irrespective of the received average SIR. However, we can determine the optimum target SIR of class-II traffic which maximizes the aggregate throughput of class-II traffic. How to determine the target SIRs will be presented in Section 3.6.1.

Now, we assume that there are K_{vo} voice mobiles with target SIR $\hat{\gamma}_o^{vo}$ and K_{vi} video mobiles with target SIR $\hat{\gamma}_o^{vi}$. In the next frame, $K_{dt,k}^a$ k-rate data mobiles, $k = 1, 2, \cdots, M$ (= 16), with target SIR $\hat{\gamma}_o^{dt}$ are scheduled to transmit packets. Note that a single mobile can simultaneously have all of voice, video, and data traffic. Interference from the other mobiles can be approximated by a Gaussian noise with a zero mean. This approximation was used first in [84], and widely accepted. When random PN codes are used, the required average energy of convolutionally-coded symbol at the receiver for a voice packet can be obtained as:

$$\overline{E}_s^{vo} = \frac{1.5PGN_o}{(\hat{\gamma}_o^{vo} + 1.5PG)/\hat{\gamma}_o^{vo} - I_{est}}, \tag{3.18}$$

where I_{est} is the total estimated interference level during the subsequent slot as a function of K_{vo}, K_{vi}, and $K_{dt,k}^{a}$ (to be defined later), N_o is the single-side power density of white Gaussian noise, and PG is the processing gain of the CDMA system. The processing gain is defined as $PG = R_c/R_s$, where R_c is the chip rate of the PN sequence and R_s is the convolutionally-coded symbol rate (as shown in Figure 3.1). The analysis of DS/CDMA system similar to that of an MC-CDMA system can be found in [63] which did not consider traffic heterogeneity or classification.

For a mobile which will transmit voice at i-rate, video at j-rate, and data at k-rate, where $i+j+k \leq M$, during the subsequent slot, the corresponding average symbol energies (i.e., $\overline{E}_s^{vi}$ for video, and $\overline{E}_s^{dt}$ for data) relative to $\overline{E}_s^{vo}$ can be given by

$$\begin{aligned} \overline{E}_s^{vi} &= \beta_{vi}\overline{E}_s^{vo}, \\ \overline{E}_s^{dt} &= \beta_{dt}\overline{E}_s^{vo}, \end{aligned} \tag{3.19}$$

where the relative energy ratios are

$$\begin{aligned} \beta_{vi} &= \frac{\hat{\gamma}_o^{vi}}{\hat{\gamma}_o^{vo}}, \\ \beta_{dt} &= \frac{\hat{\gamma}_o^{dt}}{\hat{\gamma}_o^{vo}}. \end{aligned} \tag{3.20}$$

Now, the value of the total estimated interference level I_{est} during the subsequent slot is given by

$$I_{est} = BW_{eq} + \beta_{dt}\sum_{k=1}^{M} kK_{dt,k}^{a}, \tag{3.21}$$

where BW_{eq} is the equivalent bandwidth which was reserved for class-I traffic, and $K_{dt,k}^{a}$ is the number of k-rate data mobiles scheduled. BW_{eq} is determined for a given pair (K_{vo}, K_{vi}) of admitted voice and video mobiles during the admission control phase, as described in the next subsection. Note that for simplicity the expression of I_{est} does not account for the inter-cell interference. By considering the inter-cell interference, some additive terms will be included in the equation, depending on the traffic load in adjacent cells. The power level to be calculated is for the next frame, so the BS does not know the actual rates of class-I connections in that frame. Hence, the BS just assumes that BW_{eq} is always consumed by class-I connections. Note that Eqs. (3.18) and (3.19) represent the average symbol energies, not the instantaneous symbol energies. Depending on the distance between the BS and a mobile, and the channel condition, the instantaneous symbol energies for a given slot will vary. The BS actually controls the transmission power of each mobile's each traffic class according to these average symbol energies using combined open-loop and closed-loop power control techniques [99]. We do not consider how to control the transmission power in detail according to the allocated power level because it is beyond the scope of this chapter.

In Eq. (3.18), the following condition should hold:

$$I_{est} \leq \frac{\hat{\gamma}_o^{vo} + 1.5PG}{\hat{\gamma}_o^{vo}} - \Delta_1 = C, \tag{3.22}$$

where Δ_1 is called the *link capacity reservation factor* as it determines the bandwidth that should remain unallocated, and the value C represents the link capacity. Note that (i) if $\Delta_1 = 0$ then $\frac{E_s^{vo}}{N_o}$ could be infinite when Eq. (3.22) is satisfied with the equality, and (ii) using Δ_1, we can bound the maximum transmission energy. The required power levels of a voice mobile for the subsequent frame calculated using Eq. (3.18) is announced to the mobiles through the downlink, and the mobiles will calculate their corresponding powers using Eq. (3.19).

3.5.2 Admission Control of Class-I Connections

Before admitting each class-I connection, the corresponding mobile needs to pass the admission test, determining if the new connection can be established without affecting the QoS of the existing connections. The admission test can be derived easily by modifying Eqs. (3.21) and (3.22) as:

$$BW_{eq} \leq \frac{\hat{\gamma}_o^{vo} + 1.5PG}{\hat{\gamma}_o^{vo}} - \Delta_1 - \Delta_2 = C - \Delta_2, \tag{3.23}$$

where Δ_2 is the link capacity reservation factor for class-II traffic which is a design or dynamic parameter that depends on the underlying traffic condition. BW_{eq} is a function of the pair (K_{vo}, K_{vi}) including the newly-requested connection.

Suppose there are K_{vo}^a active voice mobiles and $K_{vi,j}^a$ j-rate video mobiles, such that $\sum_{j=0}^{8} K_{vi,j}^a = K_{vi}$ in a given slot. Assuming that all the available bandwidth BW_{dt} ($= C - BW_{eq}$) for class-II traffic is used by class-II traffic for a specific value of BW_{eq}, average

SIRs for voice and video mobiles, respectively, are given by:

$$\begin{aligned}\overline{\gamma}_o^{vo} &= \left[\frac{N_o}{\overline{E}_s^{vo}} + \frac{K_{vo}^a - 1 + \sum_{j=1}^{8} \beta_{vi} j K_{vi,j}^a + BW_{dt}}{1.5PG}\right]^{-1}, \\ \overline{\gamma}_o^{vi} &= \frac{1.5PG}{\left(\frac{1.5PG}{\overline{\gamma}_o^{vo}} + 1\right) / \beta_{vi} - 1}. \end{aligned} \tag{3.24}$$

The BERs for each state $(K_{vo}^a, K_{vi,j}^a, j = 0, \cdots, 8)$ can be obtained by using the equations in Section 3.4. Because we can calculate the distributions of active voice mobiles and actual video rates from Eqs. (3.2) and (3.3), the average BERs can be obtained for a given set of $(K_{vo}, K_{vi}, BW_{eq})$. Using an iterative search, the minimum BW_{eq} satisfying the admission condition in Eq. (3.23) and all of the required BERs can be obtained for a given pair (K_{vo}, K_{vi}). If there does not exist such a BW_{eq}, the corresponding numbers of voice and video mobiles cannot be admitted into the system. Because it is computationally expensive to calculate BW_{eq} for each pair of voice and video mobiles, one can calculate it off-line and store the results in the BS to be used at run-time.

3.5.3 Class-II Transmission Permission Control

Class-II traffic transmission-permission is controlled under a round-robin scheduling policy with a set of four prioritized queues. When there are M_o mobiles in the cell, the BS is equipped with the following four queues (in that order of priority): (1) a first-in-first-out (FIFO) queue for class II-A retransmissions; (2) a class II-A round-robin queue (with M_o entities, the number of mobiles in the cell); (3) a FIFO queue for class II-B retransmissions; and (4) a class II-B round-robin queue (with M_o entities). The i-th entity of a round-robin queue, corresponding to mobile i, contains its requested transmission rate $R_{req}[i]$, and is served in a round-robin fashion. If a number of transmitted packets by a mobile results in an error, the mobile ID and the number of the packets (R_{ret}) will be placed at one of two FIFO queues, depending on the packets' priority, for retransmission. Figure 3.3 shows a pair of FIFO and round-robin queues for M_o mobiles in the cell.

First, the permitted transmission-rate index $R_{per}[i]$ for mobile i, the maximum transmission-rate index $R_{max}[i]$ for mobile i, and the residual bandwidth BW_{res} are defined. For the subsequent frame, transmission-permissions are scheduled recursively as follows.

S1. With the value BW_{dt} of the available bandwidth for class-II traffic, the residual bandwidth $BW_{res} := BW_{dt}$, and for every i, $R_{per}[i] := 0$. For every mobile i, the maximum

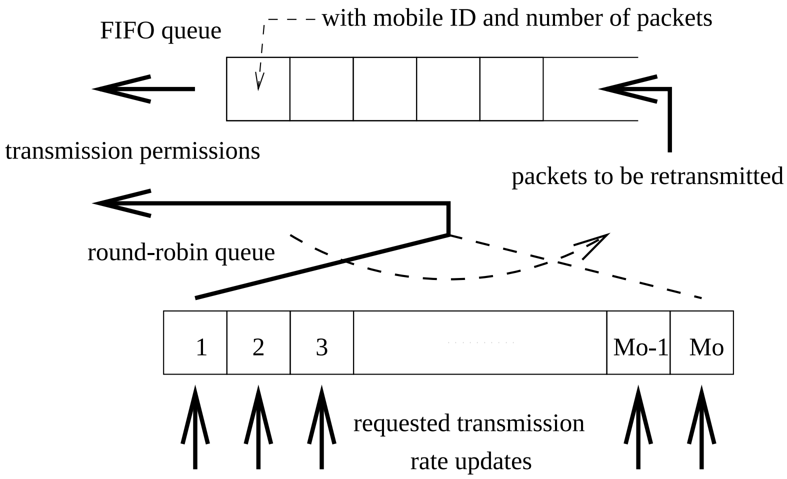


Figure 3.3: A FIFO queue for retransmission and a round-robin queue.

transmission-rate index is assigned as follows:

$$R_{max}[i] := \begin{cases} M, & \text{if mobile } i \text{ has no class-I connection,} \\ M-1, & \text{if mobile } i \text{ has a voice connection,} \\ M-8, & \text{if mobile } i \text{ has a video connection,} \\ M-9, & \text{if mobile } i \text{ has both voice and video connections.} \end{cases} \tag{3.25}$$

S2. For the next step, a mobile is selected according to the order of the class II-A FIFO queue, the class II-A round-robin queue, the class II-B FIFO queue, and the class II-B round-robin queue, where the requested transmission rate is determined by R_{ret} for FIFO queues and $R_{req}[i]$ for round-robin queues, respectively. When mobile i is selected, $R_{per}[i]$ should be less than $R_{max}[i]$. Otherwise, mobile i is skipped, and the next mobile will be selected.

S3. The available rate R_{ava} using the residual bandwidth BW_{res} is calculated as:

$$R_{ava} = BW_{res}/\beta_{dt}. \tag{3.26}$$

S4. For an m-rate transmission-request of mobile i, $R_{tmp} := R_{per}[i]$. Check if $m \leq R_{ava}$, then the transmission-permission rate is set accordingly (i.e., if yes, then $R_{per}[i] := m + R_{tmp}$, else $R_{per}[i] := \lfloor R_{ava} \rfloor + R_{tmp}$). Next, if $R_{per}[i] > R_{max}[i]$, then $R_{per}[i] := R_{max}[i]$. Then, the residual bandwidth is adjusted as:

$$BW_{res} := BW_{res} - (R_{per}[i] - R_{tmp}) \cdot \beta_{dt}. \tag{3.27}$$

S5. If $BW_{res} \geq \beta_{dt}$, then go to Step 3 after selecting the next mobile using Step 2, else go to the next step.

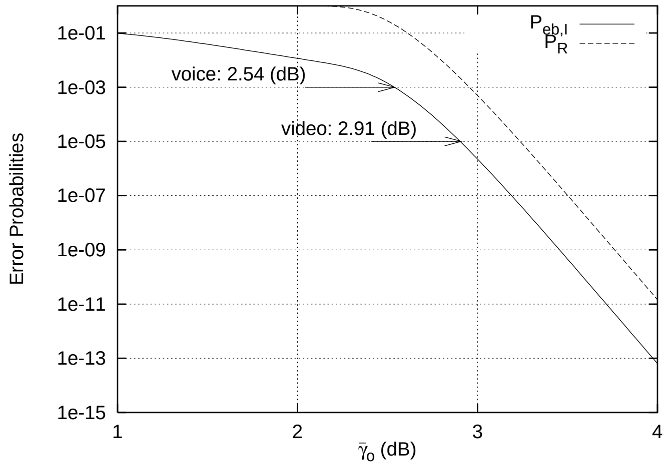


Figure 3.4: Error probabilities vs. average SIR $\overline{\gamma}_o$ for the RS/convolutional concatenated coding system.

S6. The value of $R_{per}[i]$ is the transmission-rate to be permitted for mobile i, so the values of $K^a_{dt,k}$, $k = 1, 2, \cdots, M$, can be calculated. Then, calculate the voice-only mobile's power level using Eqs. (3.18) and (3.21). The transmission permission and power level are announced to mobiles through the downlink.

3.6 Numerical Results and Discussions

To derive numerical results, we used the best rate-1/2 convolutional code with $d_{free} = 10$ and $(256, 240, 256)$ extended RS code with the error-correction capability $t = 8$. The values of $\{b_d\}$ in Eq. (3.10) are obtained from [24]. Notice that the total code rate is $1/2 \cdot 240/256 = 15/32 \approx 0.47$. The PN codes with the processing gain $PG = 128$ and the link capacity reservation factor $\Delta_1 = 1$ are used.

3.6.1 Error Performance and Target SIRs

In Figure 3.4, we plotted the following two error probabilities as the average SIR $\overline{\gamma}_o$ increases:

Bit error probability $P_{eb,I}$ at the output of the CRC decoder when the error-correction-only mode RS decoder for class-I traffic is used;

Packet retransmission probability P_R when the error-correction-and-detection mode RS decoder and CRC decoder for class-II traffic is used.

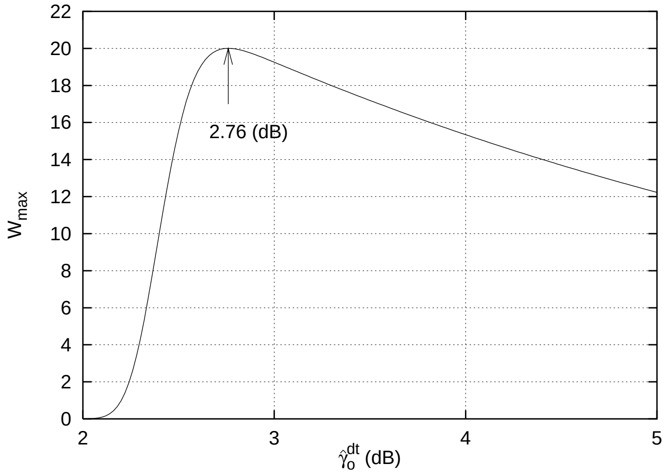


Figure 3.5: Maximum total throughput at peak traffic W_{max} vs. class-II traffic target SIR $\hat{\gamma}_o^{dt}$ with $BW_{dt} = 0.2C$.

According to each BER requirement in Table 3.1, we can determine the target SIR for each class-I traffic type. For class-I voice and video, we obtain $\hat{\gamma}_o^{vo} = 2.54$(dB) and $\hat{\gamma}_o^{vi} = 2.91$(dB), respectively, from the $P_{eb,I}$ curve because the error-correction-only scheme is used in the RS decoder. In [16], we reported results on the same CDMA system (with a different power allocation) without considering fading. The target SIRs for voice and video traffic were found to be -0.3 and -0.12, respectively, under that condition. About 3 (dB) additional power was found to be necessary to attain the same error performance for each type of traffic in the fading environment. Note the link capacity $C \approx 107$ (BUs) is determined from Eq. (3.22).

Now, we study the relationship between the power level and the throughput for class-II traffic to determine the target SIR for class-II traffic. Assume that there are only the mobiles with data to be transmitted at the basic rate, and the ideal selective-repeat ARQ is used. We obtain the maximum total throughput at peak traffic for class II, which is defined as the expected total number of packets successfully received at the receiver during one slot in the presence of more class-II traffic demands than the capacity available for class II:

$$W_{max} = (1 - P_R)K_{dt,1}^*, \tag{3.28}$$

where $K_{dt,1}^*$ is the maximum number of mobiles which can simultaneously transmit data. For a specific value of BW_{dt}, $K_{dt,1}^*$ can be expressed, by modifying Eqs. (3.21) and (3.22), as follows:

$$K_{dt,1}^* = \lfloor BW_{dt}/\beta_{dt} \rfloor. \tag{3.29}$$

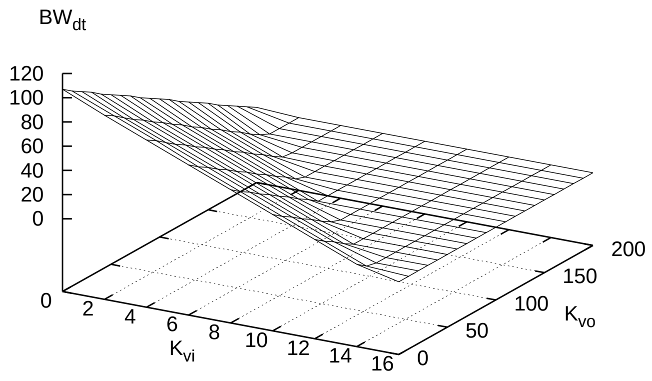


Figure 3.6: Admission regions for the pairs (K_{vo}, K_{vi}).

There is a tradeoff between $\hat{\gamma}_o^{dt}$ and $K^*_{dt,1}$, i.e., for a low $\hat{\gamma}_o^{dt}$, $K^*_{dt,1}$ would be high, and vice versa. But, because P_R would be high for low $\hat{\gamma}_o^{dt}$, there might exist an optimum $\hat{\gamma}_o^{dt}$ that maximizes W_{max}. In Figure 3.5, we find the optimum $\hat{\gamma}_o^{dt}$ to be 2.76 (dB). We can easily see that the result will be the same for different transmission rates and BW_{dt} values.

3.6.2 Admission Region for Class-I Connections

Using three target SIRs and the procedure in Section 3.5.2, we obtained the admission region shown in Figure 3.6, where the link capacity $C \approx 107$. The surface represents the bandwidth, BW_{dt}, which can be used for class II, given the admitted voice and video mobiles (K_{vo}, K_{vi}). For a given capacity reservation factor Δ_2 for class II, the pairs (K_{vo}, K_{vi}) with $BW_{dt} \geq \Delta_2$ can be admitted to the system. From the numerical results, we found that a voice mobile consumes roughly 0.53 BU while a video mobile consumes 7.07 BUs, and we can use the following equation for the admission test:

$$\lceil 0.53K_{vo} + 7.07K_{vi} \rceil \leq C - \Delta_2. \tag{3.30}$$

It can reduce the BS's storage cost at the expense of some possible connection blocks due to the inaccuracy caused by using the ceiling function in the above equation.

3.6.3 Performance of Class-II Traffic Transmission

Finally, let's consider class-II performance in terms of the average message transmission delay. A message transmission delay is defined by the time duration from the generation of a message to the transmission of its last packet. We used the following assumptions and parameters for the simulations:

A1. There are 50 mobiles which generate only class-II traffic throughout the simulation.

A2. We neglect the noise and fading effects in the request transmissions for simplicity; each transmission request is received correctly unless it collides with another request in the same code channel.

A3. Throughout the simulation, the value of BW_{dt} is kept constant; the class-I connections are not changed.

A4. The number of packets in a class II-A (II-B) message is geometrically distributed with average 2 (18).

A5. For a total message generation rate λ (messages/frame), class II-A (II-B) messages are generated independently in each of 50 mobiles according to a Poisson process with rate $0.9\lambda/50$ ($0.1\lambda/50$).

Note that the offered load, defined by the overall expected number of the packets to be transmitted within a frame, for the overall message-generation rate λ (messages/frame) is $(0.9 \cdot 2 + 0.1 \cdot 18)\lambda = 3.6\lambda$ (packets/frame), in which a half of the messages is for class II-A and the other half is for class II-B.

A message-transmission delay consists of three components: (1) a request access delay, i.e., from the generation of a message to the time at which the mobile notifies the required transmission rate to the BS through a contention-based transmission-request mini-slot or a contention-free piggyback rate-request mini-slot. Physically, a mobile requests the required transmission rate, not the transmission of each message, but from an analytical point of view, we can imagine that a successful transmission of a rate request means the successful transmission requests for all those messages that have already been generated; (2) a queueing delay of the request in the BS queues, or equivalently, the delay from the successful rate request after the generation to the transmission of the first packet of the message; and (3) a transmission delay, i.e., the time from the transmission of the first packet to the completion of the last packet transmission.

We first consider the request access delay of the protocol. Figure 3.7 plots the average request access delay as the offered load increases for both class II-A and II-B with the transmission-request access channel number $N_{rcvr}^{rqst} = 15$, 20, and 25. The average request access delay is always greater than, or equal to, 0.5 because message-generation times are

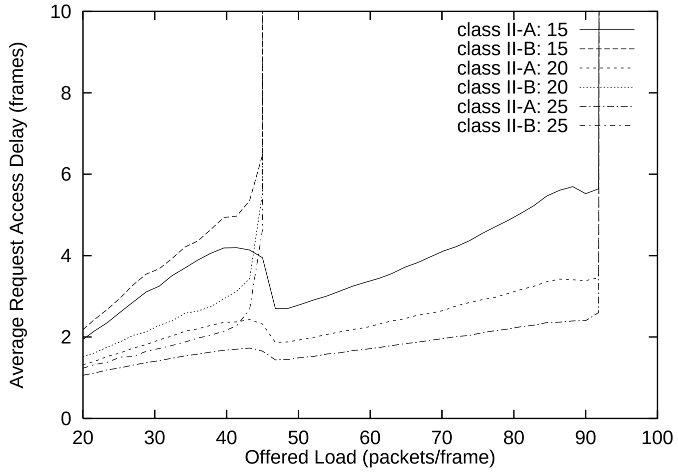


Figure 3.7: Average request access delay vs. offered load with $BW_{dt} = 50$.

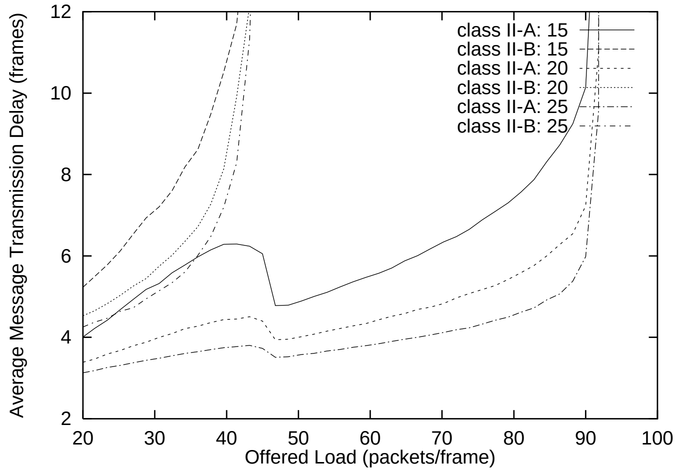


Figure 3.8: Average message transmission delay vs. offered load with $BW_{dt} = 50$.

uniformly distributed within a frame time. Because a request for class II-A is given priority over that for class II-B, the request access delays reflect this fact. We observe that for the marginal offered load, i.e., near 47 for class II-B, and around 95 for class II-A, the delays approach infinity, because at these loads, the existing messages are rarely served, so the mobiles can rarely request newly-generated messages. The reason why the marginal load for class II-A is twice that for class II-B is somewhat obvious because each half of the offered load is from each subclass. We observe that the larger N_{rcvr}^{rqst}, the lower the average request access delay, since the larger N_{rcvr}^{rqst}, the less likely a request results in a collision. One can see that the delay for class II-A increases until around the offered load of 45 is reached, then decreases somewhat abruptly, and starts to increase slowly. The reason why it decreases in the middle is that the role of the piggyback rate-request becomes effective as the number of packet transmissions within a frame increases. However, as the offered load increases further, the number of transmitted packets in a frame also increases, then the total number of mobiles which transmit in a frame will decrease. So, the chance for a mobile to request through a piggyback rate-request mini-slot decreases, so the request access delay increases. The delay eventually goes to infinity at the marginal load. A similar phenomenon was observed from the non-real-time communication performance of our WLAN system in the previous chapter, and also reported for another MC-CDMA system in [63]. It is observed that this behavior of the request access delay showing a peak is almost negligible for the case with a large N_{rcvr}^{rqst}, e.g., $N_{rcvr}^{rqst} = 25$ in the figure, since there will rarely be request collisions in this case.

From the above figure, we observed that the larger N_{rcvr}^{rqst}, the better in terms of the request access delay. However, the physically important measure is the message transmission delay, not the request access delay. We only considered the latter to understand the system better. Figure 3.8 shows the average message transmission delay vs. the offered load from the same simulation as in the previous figure. We observe a similar tendency and dependency on the value of N_{rcvr}^{rqst} to those from the previous figure. Note that the average delay should be at least 2.5 because the average request access delay is at least 0.5, the queueing delay is at least 1, and the transmission delay is also at least 1. We observe that especially for class II-A in the moderately-loaded region (i.e., offered load < 100), the message-transmission delay appears as the request access delay shifted by 2 frames. This is because there will be no more than one queueing delay for class II-A in this case. Note that for class II-B in this region, the message-transmission delay is more than a shifted version of the request access delay because class II-B gets less chance to be transmitted.

Next, we examine the effect of different BW_{dt} values. Figure 3.9 plots the average delay as the offered load increases for $BW_{dt} = 50$ and 107 with $N_{rcvr}^{rqst} = 20$. $BW_{dt} = 107$ corresponds to no class-I connections, and $BW_{dt} = 50$ corresponds to about a half of the total bandwidth is used by class-I connections, e.g., 94 voice mobiles. We see that for $BW_{dt} = 50$, the delay of class II-B becomes infinite at around 50 of offered load and that of class II-A becomes infinite around 100 of the offered load, similar to the request access delay cases. For $BW_{dt} = 107$, the offered loads at which the delays become infinite are observed to be almost doubled, because the bandwidth for class-II traffic is doubled.

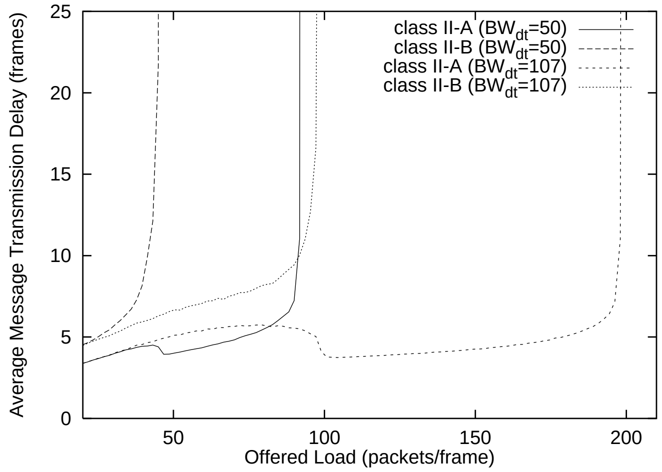


Figure 3.9: Average message transmission delay vs. offered load.

3.7 Conclusion

In this chapter, we studied a CDMA uplink system to provide diverse packet-level QoS guarantees for heterogeneous traffic, where the MC-CDMA is used to support the variable bit rates. To satisfy the pre-defined BER QoS metrics for each traffic class, we used a concatenated RS/convolutional code and developed a new scheme for allocating mobiles power levels. Connection-oriented transmission is achieved by assigning mobile-oriented code channels for class-I traffic where each corresponding mobile needs to pass an admission test. We calculated an admission region for pairs of voice and video mobiles. For class-II traffic, best-effort transmissions are supported: whenever each class-II mobile has messages to transmit, the mobile requests code channels via a transmission-request code channel, then the transmission is granted according to the underlying BS scheduling policy. We derived the target SIR by maximizing the aggregate throughput of class-II traffic and evaluated class-II transmission performance.

We are planning to extend our system to include RAKE receivers in order to combat multipath fadings more efficiently. The performance of this extension under various multipath fading channel environments will be studied. The effect of inter-cell interferences will also be studied. Considering these, the performance values reported in this chapter will change, but the general performance trends are expected to remain the same. We observed these trends by comparing the performances of our system without (in [16]) and with (in this chapter) considering the fading environment. We are also planning to extend our design to support a measurement-based admission control (e.g., [49]) of class-I connections, because it is not generally possible to have *a priori* information on class-I traffic. This admission control will have such a form as: (1) a connection is admitted based on its peak rate, (2) its time-varying transmission rate is observed during runtime, then (3) its equivalent bandwidth is adjusted accordingly.

CHAPTER 4

BANDWIDTH RESERVATION FOR HAND-OFFS

4.1 Introduction

One of the most important connection-level QoS issues is how to control (or reduce) hand-off drops due to lack of available bandwidth in the new cell, since mobile users should be able to continue their on-going sessions, and (2) more frequent hand-offs are expected to occur as a result of the current trend of shrinking cell size. We will consider two connection-level QoS parameters: the probability P_{CB} of blocking new connection requests and the probability P_{HD} of dropping hand-offs. Ideally, we would like to have no hand-off drops so that on-going connections may be preserved as long as the user wants as in a QoS-guaranteed wired network. However, this requires the network to reserve bandwidth in all cells a mobile might pass through; this is not possible in most cases, because the mobile's direction is not known *a priori*. Moreover, this per-connection/mobile reservation will severely under-utilize, and hence quickly deplete, bandwidth, which will, in turn, cause high P_{CB}.

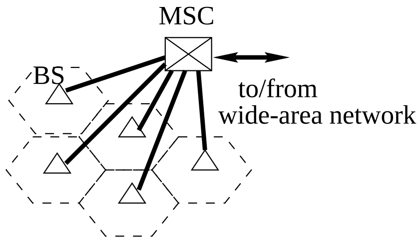
Each cell can, instead, reserve fractional bandwidths of on-going connections in its adjacent cells, and this aggregate reserved bandwidth (of multiple on-going connections) can be used solely for hand-offs, not for new connection requests. The problem is then how much of bandwidth in each cell should be reserved for hand-offs. With this chapter, we present a *predictive* and *adaptive* scheme for bandwidth reservation and admission control that keeps the hand-off dropping probability below a target value $P_{HD,target}$. Since it is practically impossible to completely eliminate hand-off drops, the best one can do is to provide some form of *probabilistic QoS guarantees* by keeping P_{HD} below a pre-specified value. Our scheme is *predictive* as it estimates the directions and hand-off times of on-going connections in adjacent cells, and *adaptive* because it dynamically adjusts the amount of reserved bandwidth according to the estimation results and the observed hand-off dropping events. Mobiles' path/direction information — readily available for certain applications, such as the navigation system of the Intelligent Transportation Systems (ITS) [105] — can also be utilized in our scheme.

For a comparative purpose, five other existing bandwidth reservation schemes — called **STATIC** [45], **NAG** [71], **AG** [95], **BHARG** [66, 64], and **NCBF** — are considered. **STATIC** reserves a portion of link bandwidth permanently for hand-offs, **NAG** are to keep the hand-off dropping probability below a target value as ours, **AG** is to guarantee no hand-off drops through per-connection bandwidth reservation, and **BHARG** & **NCBF** use another type of per-connection bandwidth reservation. These schemes are compared with our proposed schemes quantitatively in terms of (1) hand-off dropping probability, connection-blocking probability, and bandwidth utilization; (2) dependency on the design parameters; (3) dependency on the accuracy of mobility estimation; and (4) complexity.

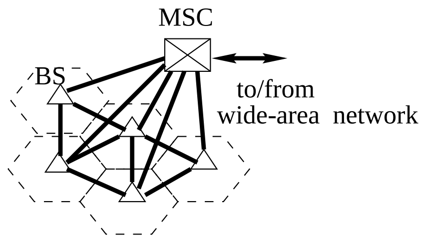
The rest of this chapter is organized as follows. Section 4.2 describes the system specification and states the assumptions to be used. The users' mobility estimation based on an aggregate history of observations is presented in Section 4.3. Section 4.4 describes the proposed predictive, adaptive bandwidth reservation and three admission-control schemes. Section 4.5 describes **NAG** utilizing our mobility estimation, and Section 4.6 presents three per-connection bandwidth reservation schemes, **AG**, **BHARG**, and **NCBF**. Section 4.7 presents and discusses the simulation results of the proposed and other existing schemes under various scenarios. Finally, the chapter concludes with Section 4.8.

4.2 System Model

In this chapter, we assume that each connection requires some specific amount of bandwidth to live with for simplicity. Then, a hand-off attempt could fail due to insufficient bandwidth in the new cell, and in such a case, a *connection hand-off drop* occurs. Here, we preclude (1) delay-insensitive applications, which might tolerate long hand-off delays in case of insufficient bandwidth in the new cell at the time of hand-off; and (2) soft hand-off of the CDMA systems [99], in which a mobile can communicate via two adjacent BSs simultaneously for a while before the actual hand-off takes place. So, the bandwidth reservation for hand-offs considered in this chapter is mostly for real-time connections even though a similar scheme might be adopted for delay-sensitive non-real-time traffic as well. We propose to set aside some bandwidth in each cell for possible hand-offs from its adjacent



(a) Star topology case



(b) Fully-connected case

Figure 4.1: Communication architectures among the MSC and BSs: (a) star topology with the MSC at the center, and (b) fully-connected network.

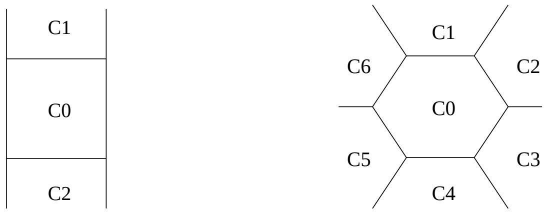
cells. This reserved bandwidth can be used only for hand-offs from adjacent cells, but *not* by newly-requested connections in the cell. Then, a newly-requested connection in a cell requires a very simple admission test:

$$\sum_i b_i + b_{new} \leq C - B_r, \tag{4.1}$$

where C is the wireless link capacity, B_r is the *target reservation bandwidth*, i.e., the required bandwidth to be reserved for hand-offs, b_i is the bandwidth being used by an existing connection i, and b_{new} is the bandwidth required by the newly-requested connection. Upon arrival of a new connection request, B_r is updated predictively and adaptively — before performing the admission test Eq. (4.1) on the request — depending on the traffic status in adjacent cells. Note that B_r is a target, not the actual reserved bandwidth, since a cell may not be able to reserve the target bandwidth, i.e., $\sum_i b_i + B_r > C$. This can happen because a BS can control the admission of only newly-requested connections, not those connections handed off from adjacent cells.

4.2.1 Communication Architecture

Our bandwidth reservation is based on information from adjacent cells such as the number of existing connections and their bandwidth requirements. Thus, it is very important to maintain inter-BS communications. The underlying network topology for BSs can have mainly two possible configurations as shown in Figure 4.1. There is a node called "Mobile Switching Center" (MSC), which covers a number of BSs, and works as a gateway to and from the wide area network. Figure 4.1 (a) shows a star-topology interconnection among the MSC and BSs, in which there are no direct connections among BSs. This is a typical structure found in the currently-deployed cellular networks. In this environment, each BS delivers the information about existing connections in its cell to the MSC. The MSC will



(a) 1-dimensional case (b) 2-dimensional case

Figure 4.2: Indexing of cells.

then determine the target reservation bandwidth in each cell, and accordingly, will perform the admission test for each newly-requested connection in a cell within its coverage. On the other hand, Figure 4.1 (b) shows the case where BSs are fully-connected. In this topology, BSs can communicate directly, not via the MSC, and each BS can determine the target reservation bandwidth, and hence, perform the admission test for each newly-requested connection in its cell.

4.2.2 Cell Indexing

All cells around each cell A are indexed:[1] A with 0, and the others with numbers beginning with 1 as shown in Figure 4.2. Let $C_{i,j}$ be connection j in cell i and $b(C_{i,j})$ be its required bandwidth. For simplicity, we assume that a mobile cannot have multiple connections simultaneously, so by an *active mobile*, we mean a mobile with one existing connection.[2] The cellular system uses a fixed channel allocation (FCA) scheme, and cell i has a wireless link capacity $C(i)$. The unit of bandwidth is BU, which is the required bandwidth to support a voice connection. A connection runs through multiple wired and wireless links, and hence, we need to consider bandwidth reservation on both wireless and wired links for hand-offs. However, we will confine ourselves to reservation of wireless link bandwidth in each cell, because routing and/or re-routing upon hand-off of a connection is beyond the scope of this chapter. Our scheme can be extended easily to include wired link bandwidth reservation by considering the routing and re-routing inside the wired network.

4.3 Mobility Estimation

We probabilistically model mobiles' hand-off behavior and estimate their mobility based on an aggregate history of hand-offs observed in each cell. In order to understand the

[1]This is the cell A's (or its base station's) centric view.

[2]Hence, we will use the terms "connection" and "mobile" inter-changeably throughout this chapter.

rationale behind our mobility estimation, let's consider the usual road traffic as an example:

O1. There are speed limits in most roads, and mobiles' speeds usually are not much higher or lower than the speed limits.

O2. In local roads, traffic signals affect mobiles' movements significantly.

O3. During the rush hours, the speeds of all mobiles in a given geographical area are closely correlated.

O4. In many cases, the direction of a mobile can be predicted from the path the mobile has taken so far.

From the above observations, we expect that the hand-off behavior of a mobile will be probabilistically similar to the mobiles which came from the same previous cell and are now residing in the current cell. Hence, we can predict the next cell of a mobile and estimate its hand-off time by utilizing an aggregate history of observations in each cell. Even though the above observations were made from road traffic, the same method can be used for pedestrians because the speeds of pedestrians won't be that much different among themselves. In a typical outdoor cellular network, there will be both pedestrian and vehicular mobiles while in the indoor case, there are mostly pedestrians or non-moving objects.

The location and direction of active mobiles are, in general, unknown to the underlying wired network (or BS). This is commonly the case when a mobile user is communicating via her/his handset or portable computer. On the other hand, there might be a special case when the wired network knows a mobile's path/direction, i.e., the cells the mobile will traverse in future. When a mobile user has a navigation system, which directs the mobile to a specified destination using the Global Positioning System (GPS) and an ITS navigation system [105], and there is a communication interface between the navigation system and the wired network, the mobile's path information can be made available to the wired network. When this path information is available to the wired network, our scheme will be shown to work better than the case without path information.

Another possibility is to use mobile-specific histories as suggested in [66]. That is, each specific mobile's movement is observed over time, then the mobile's direction in a specific cell can be predicted by utilizing this observation. However, keeping track of each mobile's mobility over time is too costly, and in many cases, mobile-specific histories are not accurate enough to make good predictions. So, we preclude the availability of such information.

4.3.1 Hand-Off Estimation Functions

We now develop a scheme to estimate and predict mobility. This scheme will be executed by the BS of each cell in a distributed manner. For each mobile which moves into an adjacent cell from the current cell 0, the cell 0's BS caches the mobile's quadruplet, $(T_{event}, prev, next, T_{soj})$, called a *hand-off event quadruplet*, where T_{event} is the time when the mobile departed from the current cell, $prev$ is the index of the previous cell the mobile had resided in before entering the current cell, $next$ is the index of the cell the mobile entered after departing from the current cell, and T_{soj} is the sojourn time of the mobile in the current cell, i.e., the time span between the entry into and departure from the current cell. Note that $prev = 0$ means that the departed mobile started its connection in the current cell.

From the cached quadruplets, the BS builds *hand-off estimation function*, which describes the estimated distribution of the next cell and sojourn time of a mobile, depending on the cell the mobile stayed before. One can also imagine that this probabilistic behavior of mobiles, especially in terms of sojourn time, will depend on the time of day, e.g., the sojourn time during rush hours will differ significantly from that during non-rush hours. We assume that the probabilistic behavior will mostly follow a cyclic pattern with the period of one day. A hand-off estimation function, at the current time t_o, is obtained as follows: for a quadruplet $(T_{event}, prev, next, T_{soj})$ such that

$$t_o - T_{int} - nT_{day} \le T_{event} < t_o + T_{int} - nT_{day}, \tag{4.2}$$

where T_{int} is the estimation interval of the function which is a design parameter, T_{day} is the duration of a day, i.e., 24 hours, and n (≥ 0) is an integer,

$$F_{HOE}(t_o, prev, next, T_{soj}) := w_n, \tag{4.3}$$

where $1 \ge w_n \ge w_{n+1}$, and $w_n = 0$ for all $n > N_{win\_days}$. The weight factor w_n is from the fact that the traffic condition in a cell during a specific period of days can vary over time. N_{win_days} is a design parameter so that the quadruplet observed more than $(N_{win\_days} \cdot T_{day} + T_{int})$ ago is determined out-of-date, and not used for the hand-off estimation function. One can easily see that the hand-off estimation functions are affected by the hand-off event quadruplets within the periodic windows of duration $2T_{int}$ as shown in Figure 4.3. Note that the duration $[t_o, t_o + T_{int}]$ is missing in the figure because it represents a future time, which is not meaningful in the definition of a hand-off event quadruplet.

In practice, it is desirable to limit the number of the quadruplets (1) used for the hand-off estimation function and (2) currently not used for the hand-off estimation function, but

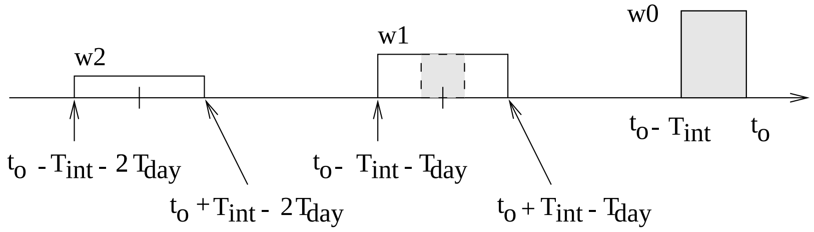


Figure 4.3: An example of periodic windows to obtain hand-off estimation functions with $N_{win\_days} = 2$.

cached for future use, e.g., those with $t_o + T_{int} - T_{day} < T_{event} < t_o - T_{int}$ in Figure 4.3, in order to reduce the memory and computation complexity.[3] We define the *maximum hand-off estimation function size*, N_{quad}, as the maximum number of hand-off event quadruplets used for the hand-off estimation function for each *prev*. This implies that we don't need the quadruplets from previous days if we observed enough during the last T_{int} interval. Up to N_{quad} cached quadruplets are used for the hand-off estimation with the following priority rule. First, the quadruplet which satisfies Eq. (4.2) with a smaller n gets higher priority. Second, among those satisfying Eq. (4.2) with the same n, the quadruplet with a smaller $|T_{event} - nT_{day}|$ gets higher priority. Figure 4.3 shows an example that only the quadruplets with the event times T_{event} within the shaded regions are used for the hand-off estimation function according to the priority rule, implying that the total number of quadruplets within the regions is N_{quad}. In order to reduce the cache memory size, those quadruplets observed at time t', i.e., $T_{event} = t'$, when the hand-off estimation function at time t' doesn't use any quadruplets observed previous days are not cached for future use, because they are unlikely to be used for the hand-off estimation function next day. Note that those quadruplets (1) with $T_{event} < t_o - T_{int} - N_{win\_days}T_{day}$ and (2) not used for the hand-off estimation function during the last $(T_{day} + T_{int})$ can be deleted from the cache memory.

There are other types of periodic and aperiodic patterns to consider for mobility estimation. These will be observed during weekends and holidays, and the mobility patterns will be significantly different from those during weekdays. So, another set of quadruplets will be cached for these special days, and the hand-off estimation functions for weekends, for example, will be built using Eqs. (4.2) and (4.3) by replacing T_{day} and N_{win_days} with $T_{week} = 7$ (days) and N_{win_weeks}, respectively. Figure 4.4 shows an example of footprint of the hand-off estimation function for $prev = 1$ without showing the values of w_n's. The number of

[3]The calculations for the mobility estimation will be dependent on the number of the quadruplets used for the hand-off estimation function as will be shown in the next section.

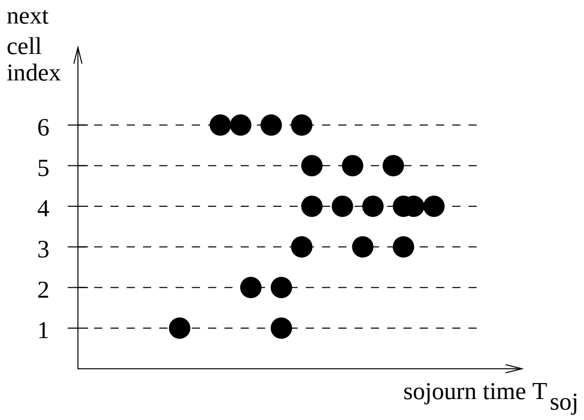


Figure 4.4: An example of the footprint of the hand-off estimation function for $prev = 1$.

dots in the footprint, each of which corresponds to a cached hand-off event quadruplet, will be bounded by N_{quad}. In the hand-off estimation function in a 3-dimensional space, the function is shown to have different heights, depending on the values of w_n's. The example is drawn from the same indexing as shown in Figure 4.2 (b). From the footprint, we observe that cell 4 is the farthest cell from cell 1 (i.e., the previous cell) through cell 0 (i.e., the current cell) among the adjacent cells of cell 0 since the sojourn times before entering cell 4 are generally shown to be among the largest. Note that the hand-off estimation function for a given *prev* can generate a probability mass function for a two-dimensional random vector $(next, T_{soj})$, where *next* is the predicted next cell and T_{soj} is the estimated sojourn time in the current cell. How this hand-off estimation function is used to estimate the user mobility is discussed next.

4.4 Predictive, Adaptive Bandwidth Reservation and Admission Control

We now describe predictive, adaptive bandwidth reservation and admission control to keep the hand-off dropping probability P_{HD} below $P_{HD,target}$ by utilizing the hand-off estimation functions described thus far.

4.4.1 Bandwidth Reservation without Path Information

The scheme considered here is for the general case when the wired network (or BS) does not have the path information of mobiles. The special case when BSs have path information will be treated in the next subsection. Our approach is based on the estimated mobility during the time window $[t_o, t_o+T_{est}]$, where t_o is the current time. We consider the behavior of a mobile in the current cell. The mobility of the active mobile with connection $C_{0,j}$ is estimated with $p_h(C_{0,j} \rightarrow i)$, the probability that $C_{0,j}$ hands off into cell i within T_{est}.

The hand-off probability can be computed using the hand-off estimation function as follows. The BS of a cell keeps track of each active mobile in its cell via the mobile's *extant sojourn time*. The extant sojourn time $T_{ext\_soj}(C_{0,j})$ of connection $C_{0,j}$ is the time elapsed since the active mobile with connection $C_{0,j}$ entered the current cell. Using Bayes' theorem [79], the hand-off probability $p_h(C_{0,j} \rightarrow next)$ at time t_o is calculated by

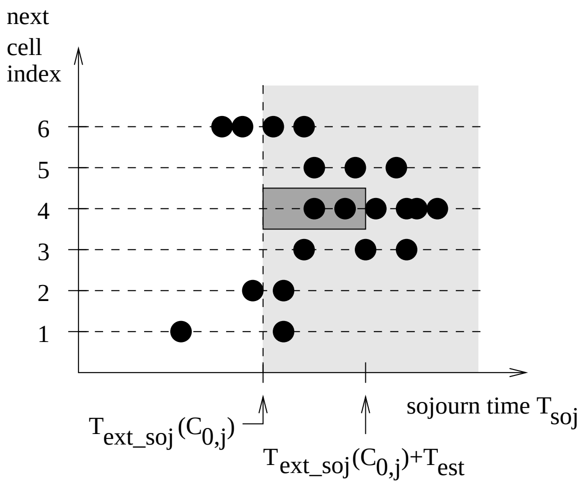
$$p_h(C_{0,j} \rightarrow next) := \begin{cases} \frac{\sum_{T_{ext\_soj}(C_{0,j}) < t_{soj} \leq T_{ext\_soj}(C_{0,j})+T_{est}} F_{HOE}(t_o, prev(C_{0,j}), next, t_{soj})}{\sum_{next' \in \mathbf{A}_0} \sum_{t_{soj} > T_{ext\_soj}(C_{0,j})} F_{HOE}(t_o, prev(C_{0,j}), next', t_{soj})}, \\ \quad \text{if } \sum_{next' \in \mathbf{A}_0} \sum_{t_{soj} > T_{ext\_soj}(C_{0,j})} F_{HOE}(t_o, prev(C_{0,j}), next', t_{soj}) \neq 0, \\ 0, \quad \text{otherwise,} \end{cases} \tag{4.4}$$

in which $prev(C_{0,j})$ is the cell which $C_{0,j}$ resided in before entering the current cell and $\mathbf{A}_i$ is the set of indices of cell i's adjacent cells. The equation represents the expected probability that $C_{0,j}$ hands off into cell $next$ with the sojourn time t_{soj} which is less than, or equal to, $T_{ext\_soj}(C_{0,j}) + T_{est}$ given the condition that $t_{soj} > T_{ext\_soj}(C_{0,j})$, which is the hand-off probability $p_h(C_{0,j} \rightarrow next)$.

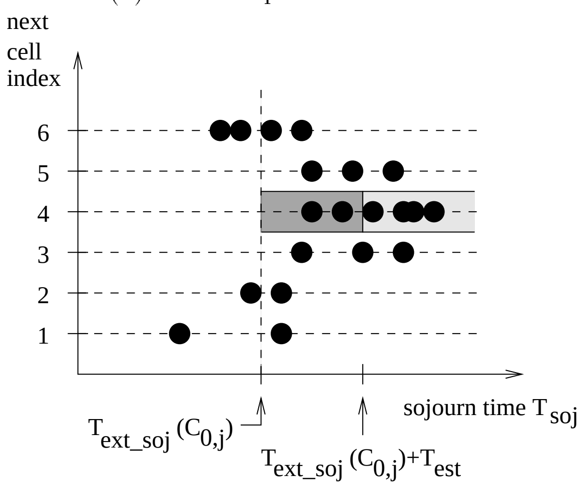
Figure 4.5 (a) shows an example of calculating $p_h(C_{0,j} \rightarrow 4)$, when $C_{0,j}$ entered cell 0 from cell 1, using the footprint of the hand-off estimation function for $prev(C_{0,j}) = 1$, shown in Figure 4.4. In the figure, the values of $F_{HOE}(t_o, 1, next', T_{soj})$ from all points at the right side of the vertical line at $T_{soj} = T_{ext\_soj}(C_{0,j})$ (i.e., in both dark and light shaded regions) are summed to obtain the denominator in Eq. (4.4). Because this value is not zero, the values of $F_{HOE}(t_o, 1, 4, T_{soj})$ from two points in the dark-shaded region are summed to obtain the numerator in Eq. (4.4). Then, we can complete the calculation of $p_h(C_{0,j} \rightarrow 4)$. Note that the mobile with connection $C_{0,j}$ is estimated to be stationary (i.e., non-moving) in cell 0 if there is no hand-off event in the hand-off estimation function with a sojourn time larger than the connection $C_{0,j}$'s extant sojourn time, i.e., the denominator in Eq. (4.4) is zero.

Now, using the probabilities of handing off connections into cell 0 from its adjacent cell i within T_{est} (i.e., hand-off probabilities $p_h(C_{i,j} \rightarrow 0)$), the required bandwidth $B^i_{r,0}$ to be reserved in cell 0 for the expected hand-offs from cell i is obtained as:

$$B^i_{r,0} = \sum_{j \in \mathbf{S}_i} b(C_{i,j}) p_h(C_{i,j} \rightarrow 0), \tag{4.5}$$



(a) without path information



(b) with path information

Figure 4.5: An example of calculating $p_h(C_{0,j} \to next)$ when $prev(C_{0,j}) = 1$ and $next = 4$ using the footprint of $F_{HOE}(t_o, prev(C_{0,j}), next', T_{soj})$.

where $\mathbf{S}_i$ is the set of indices of the connections in cell i and $b(C_{i,j})$ is connection $C_{i,j}$'s bandwidth. Finally, the target reservation bandwidth $B_{r,0}$ in cell 0, which is the aggregate bandwidth to be reserved in cell 0 for the expected hand-offs from adjacent cells within T_{est}, is calculated as:

$$B_{r,0} = \sum_{i \in \mathbf{A}_0} B^i_{r,0}, \tag{4.6}$$

where $\mathbf{A}_i$ is the set of indices of cell i's neighbors.

Note that the target reservation bandwidth is an increasing function of the estimation time T_{est} as $p_h(C_{i,j} \to 0)$ is an increasing function of T_{est}. There might be an optimal value of T_{est} for given traffic/mobility status in the sense of giving the smallest new connection blocking probability while keeping the hand-off dropping probability below the target. In our scheme, the estimation time will be adjusted adaptively in each cell independently of others, depending on the hand-off dropping events in the cell as described in Section 4.4.3. Then, the estimation time T_{est} of cell $next$ (or $T_{est,next}$) will be used in Eq. (4.4). So, when the BS in cell 0 needs to update the value of $B_{r,0}$, the BS will inform the current value of $T_{est,0}$ to the adjacent cells, then the BS in each adjacent cell will calculate the required bandwidth for the expected hand-offs from that cell, i.e., $B_{r,0}^{i}$ for cell i, using Eq. (4.5), and will inform this value to cell 0's BS. Finally, cell 0's BS will calculate $B_{r,0}$ using Eq. (4.6).

4.4.2 Bandwidth Reservation with Path Information

The above-developed scheme is modified to utilize the path information of mobiles, if available. This special case can happen, for example, when a mobile user is driving a car equipped with a navigation system, which guides the user to a specified destination from the current location using the GPS and an ITS navigation system, and the navigation system communicates with the current cell's BS to inform the next cell the user is supposed to move to. The driver may sometimes not follow the direction by the navigation system, because, for example, the driver missed the guided route accidentally. However, we don't consider these accidental cases, and assume that the mobile follows the route which is known to the BS.

In the previous subsection, a BS utilizes the hand-off estimation functions in two ways: one is to predict the next cell a mobile will move to, and the other is to estimate the mobile's sojourn time in the current cell. With path information, the BS knows of the next cell, and hence, the hand-off estimation function is used solely for the second purpose. Now, assume that the mobile with connection $C_{0,j}$ in the current cell will leave for cell $next$, then

$$p_h(C_{0,j} \to next') := 0, \quad \text{if } next' \neq next, \tag{4.7}$$

and Eq. (4.4) is modified to[4]

$$p_h(C_{0,j} \to next) :=$$

[4]Logically, one may think that $p_h(C_{0,j} \to next)$ should be assigned 1 instead of 0 in the 'otherwise' case because the extant sojourn time is larger than the maximum sojourn time in the mobility estimation function, meaning that the mobile will soon move into cell $next$. In practice, the maximum sojourn time might not be so reliable as to represent the real maximum sojourn time since only a finite number of hand-off event quadruplets are used in the mobility estimation function and the possible maximum sojourn time itself is time-varying. Assigning 1.0 to $p_h(C_{0,j} \to next)$ was found to result in frequent over-reservations.

$$
\begin{cases}
\frac{\sum_{T_{ext\_soj}(C_{0,j})<t_{soj}\le T_{ext\_soj}(C_{0,j})+T_{est}\}} F_{HOE}(t_o,prev(C_{0,j}),next,t_{soj})}{\sum_{t_{soj}>T_{ext\_soj}(C_{0,j})} F_{HOE}(t_o,prev(C_{0,j}),next,t_{soj})}, \\
\quad \text{if } \sum_{t_{soj}>T_{ext\_soj}(C_{0,j})} F_{HOE}(t_o, prev(C_{0,j}), next, t_{soj}) \neq 0, \\
0, \quad \text{otherwise.}
\end{cases} \tag{4.8}
$$

Figure 4.5 (b) shows an example of calculating $p_h(C_{0,j} \to 4)$ with path information, when connection $C_{0,j}$ came from cell 1, using the footprint of the hand-off estimation function for $prev(C_{0,j}) = 1$, shown in Figure 4.4. In the figure, the values of $F_{HOE}(t_o, 1, 4, T_{soj})$ from the points in both dark and light shaded regions are summed to obtain the denominator in Eq. (4.8). Because this value is not zero, the values of $F_{HOE}(t_o, 1, 4, T_{soj})$ from two points in the dark shaded region are summed to obtain the numerator in Eq. (4.8). Then, the value of $p_h(C_{0,j} \to 4)$ is obtained. Now, the hand-off probabilities using Eqs. (4.7) and (4.8) are used in Eqs. (4.5) and (4.6) to calculate the target reservation bandwidth. This modified scheme is expected to outperform the previous scheme in terms of estimation accuracy, and hence bandwidth-reservation efficiency.

4.4.3 Control of Mobility Estimation Time Window

Using our scheme, the bandwidth for hand-offs will be over-reserved (under-reserved) if T_{est} is too large (small). There might exist an optimal value of T_{est} for specific traffic load and user mobility, but these parameters in practice vary with time. Moreover, the mobility estimation functions used might not describe mobiles' behavior well, thus resulting in inaccurate mobility estimation even with the optimal T_{est}. We propose an adaptive algorithm for controlling the mobility estimation time window based on the hand-off dropping events in each cell so as to approximate the optimal T_{est} over time. Figure 4.6 shows the algorithm executed by the BS in each cell to adjust the value of T_{est}.

Before running the algorithm, the reference window size w $(= \lceil 1/P_{HD,target} \rceil)$ is determined and assigned to the observation window size w_{obs}. In addition, T_{est} is initialized to T_{start}, a design parameter, and the counts for hand-offs n_H and hand-off drops n_{HD} are reset to 0. As can be found in the pseudocode, w_{obs} is increased or decreased by w, and the constraint $P_{HD} < P_{HD,target}$ can be translated to that to keep the counted number n_{HD} of hand-off drops out of w_{obs} observed hand-offs below w_{obs}/w. During the runtime, whenever there is a hand-off drop after w_{obs}/w drops, $T_{est} := T_{est} + 1$ and $w_{obs} := w_{obs} + w$. On the other hand, when there were less than, or equal to, w_{obs}/w hand-off drops out of w_{obs} observed hand-offs, $T_{est} := T_{est} - 1$ and $w_{obs} := w$. T_{est} is not greater than $T_{soj,max}$ in Figure 4.6, which is the maximum T_{soj} derived from the hand-off estimation functions

```
if (w = ⌈1/P_{HD,target}⌉), then w_{obs} := w;
T_{est} := T_{start};  n_H := 0;  n_{HD} := 0;
while (time increases) {
    if (hand-off into the current cell happens) then {
        n_H := n_H + 1;
        if (it is dropped) then {
            n_{HD} := n_{HD} + 1;
            if (n_{HD} > w_{obs}/w) then {
                w_{obs} := w_{obs} + w;
                if (T_{est} < T_{soj,max}) then T_{est} := T_{est} + 1;
            }
        }
        else if (n_H ≥ w_{obs}) then {
            if (n_{HD} ≤ w_{obs}/w and T_{est} > 1) then
                T_{est} := T_{est} − 1;
            w_{obs} := w;  n_H := 0;  n_{HD} := 0;
        }
    }
}
```

Figure 4.6: A pseudocode of the algorithm to adjust T_{est} in each BS.

in adjacent cells, because any value larger than that is meaningless. We also set the minimum value of T_{est} to 1 since if the value is too small, our scheme will reserve virtually no bandwidth irrespective of the existing connections in adjacent cells.

Given below are some considerations for the design of the estimation time window control algorithm.

C1. When there were more hand-off drops than permitted, the algorithm should start to increase T_{est} quickly because of under-reserved bandwidth; otherwise, there will be continued hand-off drops.

C2. The increment of T_{est} should not be too high. Otherwise, it might result in an over-reaction, hence over-reservation.

C3. Due to over-reaction or decreased traffic load over time, there might be fewer hand-off drops than permitted, so the value of T_{est} should be decreased quickly. Otherwise, the bandwidth will continue to be over-reserved, hence under-utilizing the system.

C4. T_{est} should not be decreased too much. Otherwise, it might result in an over-reaction, hence under-reservation.

There can be many candidate algorithms satisfying the above requirements. Especially, there might be many choices of increment and decrement step sizes, both of which were fixed at 1. We experimented with other choices like additive and multiplicative step sizes: the step size was increased additively $(1, 2, 3, \cdots)$ or multiplicatively $(1, 2, 4, \cdots)$ for consecutive increments and decrements. The main purpose of these choices is a prompt reaction to hand-off drops, i.e., **C1** and **C3**. However, these choices are found to cause over-reactions, and make the reserved bandwidth fluctuate severely between over-reservation and under-reservation. The algorithm presented here is the best one we have found so far.

4.4.4 Admission Control

The admission test after calculating the target reservation bandwidth can be as simple as given in Eq. (4.1). That is,

T1. Check if $\sum_{j \in \mathbf{S}_0} b(C_{0,j}) + b_{new} \leq C(0) - B_{r,0}$.

T2. If the above test is positive, the connection is admitted,

where $C(0)$ and b_{new} are the link capacity of cell 0 and the bandwidth of the newly-requested connection, respectively. This simple admission-control scheme will henceforth be referred to as **AC1**. However, when there is not enough bandwidth left unused by existing connections that can be reserved for hand-offs, it is meaningless to calculate the target reservation bandwidth. If this situation lasts for an extended period due to continued incoming hand-offs, the problem becomes more serious because some of the incoming hand-offs will be continuously dropped due to the unavailability of reserved bandwidth, triggering further increase of T_{est}. This, in turn, requires to reserve more bandwidth that doesn't exist. This situation can happen when adjacent cells accept new connections solely based on the admission test using Eq. (4.1) and those admitted connections continue to be handed off into the current cell even though it doesn't have enough bandwidth.

To handle this problem, the admission test should check available bandwidths of adjacent cells as well as the current cell. More precisely, one must check if there is enough bandwidth in the current and next cell of each newly-requested connection, but, without path information, it is not possible to know the next cell. Thus, depending on whether path information is available or not, we apply a different admission control scheme. First, without path information:

T1. For all $i \in \mathbf{A}_0$, check if $\sum_{j \in \mathbf{S}_i} b(C_{i,j}) \leq C(i) - B_{r,i}$,

T2. Check if $\sum_{j \in \mathbf{S}_0} b(C_{0,j}) + b_{new} \leq C(0) - B_{r,0}$,

T3. If all of the above tests are positive, then the connection is admitted.

We call this scheme **AC2**. Second, with path information, i.e., we know that the new connection will later move to cell $next$:

T1. If $next \neq 0$, then check if $\sum_{j \in \mathbf{S}_{next}} b(C_{next,j}) \leq C(next) - B_{r,next}$,

T2. Check if $\sum_{j \in \mathbf{S}_i} b(C_{i,j}) + b_{new} \leq C(0) - B_{r,0}$,

T3. If the above two tests are positive, then the connection is admitted.

This scheme is referred to as **AC2 w/ PI**. Note that $next$ could be 0, and in this case, the first step can be skipped.

Note that using the above admission test in the absence of path information, the current cell and all of its adjacent cells must calculate $B_{r,i}$ for each new admission request, and this is costly. In fact, the undesirable situation described in the beginning of this subsection is expected to happen only in heavily-loaded networks. So, we present a hybrid scheme which requires only those adjacent cells which "appear" to be unable to reserve the target reservation bandwidth, to calculate the target bandwidth again and participate in the admission

Name	Path Information	Description
AC1	without	Calculation of B_r in the current cell only.
AC2	without	Calculation of B_r in the current cell
AC2 w/ PI	with	and every adjacent cell.
AC3	without	Calculation of B_r in the current cell
AC3 w/ PI	with	and some adjacent cells only.

Table 4.1: Summary of the proposed admission-control schemes.

test. Note that $B_{r,i}$ is a time-varying function, and updated upon admission test. Upon arrival of a new connection request at cell 0, if the current target reservation bandwidth of an adjacent cell i, $B^{curr}_{r,i}$, which was calculated for a previous admission test, is not reserved fully, this cell will re-calculate $B_{r,i}$, and participate in the admission test. First, without path information (**AC3**):

T1. For all $i \in \mathbf{A}_0$ such that $\sum_{j \in \mathbf{S}_i} b(C_{i,j}) + B^{curr}_{r,i} > C(i)$, check if $\sum_{j \in \mathbf{S}_i} b(C_{i,j}) \leq C(i) - B_{r,i}$,

T2. Check if $\sum_{j \in \mathbf{S}_0} b(C_{0,j}) + b_{new} \leq C(0) - B_{r,0}$,

T3. If all the above tests are positive, then the connection is admitted.

Second, with path information, or when the newly-requested connection will later depart to cell *next* (**AC3 w/ PI**):

T1. If $next \neq 0$ and $\sum_{j \in \mathbf{S}_{next}} b(C_{next,j}) + B^{curr}_{r,next} > C(next)$, then check if $\sum_{j \in \mathbf{S}_{next}} b(C_{next,j}) \leq C(next) - B_{r,next}$,

T2. Check if $\sum_{j \in \mathbf{S}_i} b(C_{i,j}) + b_{new} \leq C(0) - B_{r,0}$,

T3. If the above two tests are positive, then the connection is admitted.

Table 4.1 shows the summary of the admission-control schemes described above. These schemes will be comparatively evaluated in the next section.

4.5 A Scheme with Estimation of Incoming and Outgoing Hand-Offs

We now describe the distributed admission control scheme (referred to as **NAG**), originally proposed in [71], which utilizes the cell-specific history-based mobility estimation for a comparative purpose. Described here is more generalized than the original scheme in the sense that heterogeneous connections (in terms of connection bandwidths) are supported. The authors of [66] also presented another generalized version of the original scheme with a number of connection bandwidths. All of the previously-reported performance evaluations were based on exponentially-distributed sojourn times of mobiles in each cell and known connection hand-off/termination rates.

4.5.1 Three State Probabilities

The main difference between our schemes and **NAG** is that our schemes consider incoming hand-offs only while **NAG** considers both incoming and outgoing hand-offs in a cell. **NAG** is also based on the estimated mobility during $[t_0, t_0 + T_{est}]$, in which t_0 is the current time. Like in our schemes, we consider the behavior of a connection in the current cell. After T_{est} time units, connection $C_{0,j}$ can be in one of three different states with the corresponding probabilities shown in parentheses: (1) hand-off into an adjacent cell i ($p_h(C_{0,j} \to i)$); (2) termination after completing the corresponding communication ($p_e(C_{0,j})$); and (3) staying in the current cell ($p_s(C_{0,j})$). We compute the probability of each event by utilizing the mobility estimation.

First, the hand-off probabilities $p_h(C_{0,j} \to i)$ are defined in Eq. (4.4) for our schemes. Next, we consider how to estimate the probability that connection $C_{0,j}$ will terminate within time T_{est}, i.e., $p_e(C_{0,j})$. BSs utilize the average connection lifetime T_{ave_life} of each mobile, which is calculated over time by:

$$T_{ave\_life} := (1-\alpha)T_{ave\_life} + \alpha T_{last\_life}, \tag{4.9}$$

where α (< 1) is a design parameter, and T_{last_life} is the connection lifetime obtained from the last connection of that mobile. We assume that the connection lifetime of $C_{0,j}$ follows an exponential distribution with mean $T_{ave\_life}(C_{0,j})$. In reality, the connection lifetime might not follow an exponential distribution, but this will be most likely dependent on each mobile, not on the cell in which it resides. Hence, this assumption doesn't have significant bearing on the results. Then, the probability is given by

$$p_e(C_{0,j}) = 1 - e^{-T_{est}/T_{ave\_life}(C_{0,j})}. \tag{4.10}$$

Finally, the probability that connection $C_{0,j}$ will stay in the cell for T_{est} time units is given by

$$p_s(C_{0,j}) = (1 - p_e(C_{0,j})) \left(1 - \sum_{i \in \mathbf{A}_0} p_h\left(C_{0,j} \to i\right)\right), \tag{4.11}$$

where $\mathbf{A}_i$ is the set of indices of cell i's neighbors.

We assume that (1) the behavior of each connection is independent of others, and (2) the probability that a mobile hands off more than once during time T_{est} is negligible. Then, the required bandwidth $B_{T_{est},0}$ for handed-off and existing connections in cell 0 during T_{est} will be the sum of the bandwidths of (1) the connections which stay in cell 0 during T_{est} and (2) the connections which hand off into cell 0 from an adjacent cell during T_{est}. Using the Central Limit Theorem [79], this can be approximated to have a Gaussian distribution as:

$$Pr_{B_{T_{est},0}}(k) \approx G(m_{B,0}, \sigma_{B,0}), \tag{4.12}$$

where the mean

$$m_{B,0} = \sum_{i \in \mathbf{A}_0} \sum_{j \in \mathbf{S}_i} b(C_{i,j}) p_h(C_{i,j} \to 0) + \sum_{j \in \mathbf{S}_0} b(C_{0,j}) p_s(C_{0,j}), \tag{4.13}$$

and the variance

$$\begin{aligned} \sigma^2_{B,0} &= \sum_{i \in \mathbf{A}_0} \sum_{j \in \mathbf{S}_i} b^2(C_{i,j}) p_h(C_{i,j} \to 0)(1 - p_h(C_{i,j} \to 0)) \\ &+ \sum_{j \in \mathbf{S}_0} b^2(C_{0,j}) p_s(C_{0,j})(1 - p_s(C_{0,j})). \end{aligned} \tag{4.14}$$

Recall that $b(C_{i,j})$ is the connection $C_{i,j}$'s bandwidth, $\mathbf{S}_i$ is the set of connections' indices in cell i, and $\mathbf{A}_i$ is the set of cell i's neighbors' indices.

4.5.2 Admission Control

To make an admission decision, we define the overload probability after T_{est} in cell i as follows:

$$\begin{aligned} P_{O,i} &= Pr(B_{T_{est},i} > C), \\ &\approx Q\left(\frac{C - m_{B,i}}{\sigma_{B,i}}\right), \end{aligned} \tag{4.15}$$

where C is the link capacity. $m_{B,i}$ and $\sigma_{B,i}$ are obtained from Eqs. (4.13) and (4.14), respectively, after replacing i with k, then replacing 0 with i in the equations. Now, for a new connection request, the admission test is performed as follows:

T1. For all $i \in \mathbf{A}_0 \cup \{0\}$, check if $P_{O,i} \leq P_{HD,target}$,

T2. If all the tests are positive, the connection is admitted.

Note that for this scheme, the specific amount of bandwidth to be reserved is not defined. So, the relation between the value of T_{est} and the bandwidth reserved for hand-offs is not clear. Basically, the larger T_{est}, the larger P_h's and P_e's, hence the smaller P_s's. It is not clear whether m_B and σ_B^2 would increase or decrease as T_{est} increases. There may exist an optimal T_{est} which achieves the smallest P_{CB} while keeping P_{HD} under the target value, but it is not possible to adopt a similar scheme to the mobility estimation window control used for our schemes. We will later evaluate the effect of the value of T_{est} using simulations.

4.6 Per-Connection Bandwidth Reservation Schemes

Now, we describe three admission-control schemes **AG**, **BHARG**, and **NCBF** based on per-connection bandwidth reservation for the comparison with ours.

4.6.1 Control AG: No Hand-Off Drop

This subsection describes an admission-control scheme (referred to as **AG**, meaning "*A*bsolute *G*uarantee") which guarantees no hand-off drop. This is possible by checking the bandwidth in all cells which the mobile requesting a new connection will traverse, then reserving the required bandwidth in each of those cells. So, this admission scheme involves per-connection bandwidth reservation in each cell. This per-connection reservation and the corresponding admission control were proposed in the context of measurement-based admission control in [95].

For this scheme to work, each mobile should inform the wired network (or the corresponding BS) of the *mobility specification* that is composed of the cells the mobile will traverse during the lifetime of the requested connection. It is generally impossible to know a mobile's direction in advance. The navigation systems of Intelligent Transportation Systems (ITS) might be used to predict the mobiles' path/direction accurately, and might be used to predict the mobility specification. The problem is that using a navigation system, it is possible to know the cell to which the corresponding mobile will move next, but we do not know if the mobile's connection will continue when the mobile enters the next cell. So, it is practically impossible to know the exact mobility specification at the time of admission control. But, we describe the admission-control scheme assuming the availability of the mobility specification as in [95].

For the mobility specification M_{sp} of a newly-requested connection, which consists of a set of cells, and its required bandwidth b_{new}, admission control and per-connection bandwidth reservation are as follows:

T1. For each cell i in the mobility specification M_{sp}, check if $\sum_{j \in \mathbf{S}_i} b(C_{i,j}) + b_{new} \leq C - B_{r,i}$,

T2. If all the above tests are positive, for each cell i in the mobility specification M_{sp}, $B_{r,i} := B_{r,i} + b_{new}$, and the connection is admitted,

where $B_{r,i}$ is the sum of all per-connection bandwidths reserved in cell i. Whenever a mobile enters a cell, the cell's reserved bandwidth (for hand-offs) will be decreased:

Upon hand-off of connection $C_{i,j}$ into cell i, $B_{r,i} := B_{r,i} - b(C_{i,j})$.

Note that the cell index i used in this subsection is different from the relative index defined in Section 4.2 and used for the previously-described two schemes. Cell i here should be considered as the i-th cell in the entire cellular system. Through per-connection reservation in the cells within the mobility specification, it is possible to make the hand-off drop probability zero, but we will show how inefficient this scheme is in terms of the bandwidth utilization and the connection-blocking probability.

4.6.2 BHARG: Per-Connection Reservation after Admission

This subsection describes the second admission control scheme based on per-connection reservation referred to as **BHARG**. This scheme doesn't try to limit P_{HD} nor to eliminate hand-off drops, but just reserves each connection's bandwidth in the predicted next cell of the mobile which has an on-going connection. The key aspect of this scheme is how to predict the next cell of a mobile, and it was proposed for indoor mobile computing environments [66, 64]. We assume here that a perfect next-cell estimator, which informs the BS whether a mobile is terminating its connection in the current cell or moving into an adjacent cell with the connection, is available to evaluate the performance of per-connection bandwidth reservation. Admission control and per-connection bandwidth reservation work as follows:

T1. Check if $\sum_{j \in \mathbf{S}_0} b(C_{0,j}) + b_{new} \leq C - B_{r,0}$,

T2. If the above test is positive, for the predicted next cell $next$ of the connection, $B_{r,next} := B_{r,next} + b_{new}$, and the connection is admitted,

where $B_{r,i}$ is the sum of all per-connection bandwidths reserved in cell i. Whenever a mobile enters a cell, the cell's reserved bandwidth (for hand-offs) will be decreased:

Upon hand-off of connection $C_{i,j}$ into cell i, $B_{r,i} := B_{r,i} - b(C_{i,j})$.

Note that the admission test checks for bandwidth availability in the mobile's current cell only. Then, the BS in the predicted next cell of the mobile will try to reserve the mobile's connection bandwidth. However, this is not always possible since bandwidth availability in this next cell was not a condition for admitting the connection. In that sense, $B_{r,i}$ in cell i is not a real reserved bandwidth, but a target reservation bandwidth. Even though this scheme was not aimed for no hand-off drops, it will achieve virtually no hand-off drops as will be shown later, but at a very high cost which is comparable to that of **AG**.

4.6.3 NCBF: Per-Connection Reservation before Admission

This subsection describes the third per-connection reservation scheme referred to as **NCBF** (meaning "*N*ext *C*ell *B*andwidth Reservation *F*irst"). This scheme lies in between the previous two schemes since it predicts the next cell of a mobile requesting a new connection during the admission control phase, and that mobile is admitted only when both the current cell and the predicted next cell of the mobile have enough bandwidth to support the requested connection. The difference between **NCBF** and **AG** is that the former reserves the bandwidth in the next cell only, and the difference between **NCBF** and **BHARG** is that the former won't admit the new connection if the predicted next cell does not have enough bandwidth to support the requested connection. This scheme doesn't try to limit P_{HD} nor to eliminate hand-off drops. The next-cell prediction schemes [66, 64] proposed for **BHARG** are expected to be used for this scheme as well since it is a slight twist of **BHARG**. We again assume that a perfect next-cell estimator, which informs the BS whether a mobile is terminating its connection in the current cell or moving into an adjacent cell with the connection, is available to evaluate the performance of this per-connection bandwidth reservation. Admission control and per-connection bandwidth reservation work as follows:

T1. For $i \in \{0, next\}$ where $next$ is the index of the predicted next cell, check if $\sum_{j \in \mathbf{S}_i} b(C_{i,j}) + b_{new} \le C - B_{r,i}$,

T2. If both of the above tests are positive, the connection is admitted, and $B_{r,next} := B_{r,next} + b_{new}$,

where $B_{r,i}$ is the sum of all per-connection bandwidths reserved in cell i. Whenever a mobile enters a cell, the cell's reserved bandwidth (for hand-offs) will be decreased:

Upon hand-off of connection $C_{i,j}$ into cell i, $B_{r,i} := B_{r,i} - b(C_{i,j})$.

Note that $B_{r,i}$ in cell i for **NCBF** is the real reserved bandwidth (so it differs from that for **BHARG**). Even though this scheme was not aimed to avoid hand-off drops, it will also achieve virtually no hand-off drops, but at a very high cost comparable to that of **AG**. This scheme is claimed to be the best per-connection bandwidth reservation scheme.

4.7 Comparative Performance Evaluation

This section presents and discusses the simulation results of the proposed schemes, and compare them with other existing schemes discussed thus far. We first describe the assumptions and specifications used for the simulation study.

4.7.1 Simulation Assumptions and Specifications

We consider two types of cellular environments for our simulation study. The first one is a one-dimensional cellular system as in Figure 4.2 (a), in which mobiles are traveling along a straight road (e.g., cars on a highway). We make the following assumptions for this environment:

A1. The whole cellular system is composed of 10 linearly-arranged cells, for which the diameter of each cell is 1 km. Cells are numbered from 1 to 10, i.e., cell $< i >$ represents the i-th cell.

A2. Connection requests are generated according to a Poisson process with rate λ (connections/second/cell) in each cell. A newly-generated connection can appear anywhere in the cell with an equal probability.

A3. A connection is either for voice (requiring 1 BU) or for video (requiring 4 BUs) with probabilities R_{vo} and $1 - R_{vo}$, respectively, where the *voice ratio* $R_{vo} \leq 1$.

A4. Mobiles can travel in either of two directions with an equal probability with a speed chosen randomly between SP_{min} and SP_{max} (km/hour). Each mobile will run straight through the road with the chosen speed, i.e., mobiles will never turn around.

A5. Each connection's lifetime is exponentially-distributed with mean 120 (seconds).

A6. Each cell has a fixed link capacity 100 BUs, i.e., $C(i) = C = 100$ for all i.

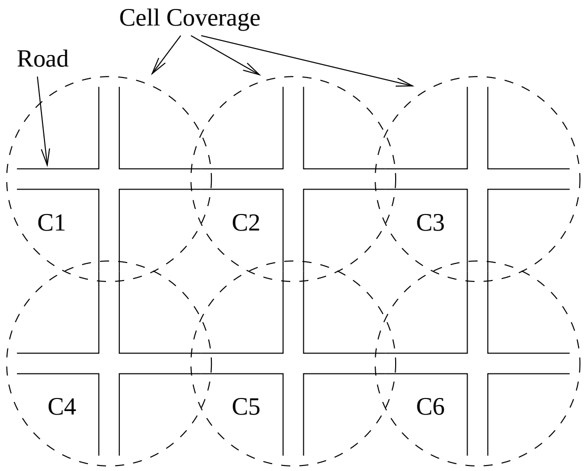


Figure 4.7: A two-dimensional cellular structure.

The second type is two dimensional cellular structure, in which the roads are mapped into a mesh as shown in Figure 4.7. A BS is located at each intersection of two crossing roads. The coverage of each cell is also shown in the figure. This cellular structure can typically be seen in a metropolitan downtown area. For the two-dimensional environment, we make the following assumptions in addition to, or replacing, the assumptions for the above-defined one-dimensional environment.

B1. The whole cellular system is composed of 25 cells arranged as a 5×5 mesh, and the diameter of each cell is 300 m.

B2. At the intersection of two roads (i.e., at the center of each cell), a mobile might continue to go straight, or turn left, right, or around with probabilities 0.55, 0.2, 0.2, and 0.05, respectively.

B3. If a mobile chooses to go straight or turn right at the center of a cell, it might need to stop there with probability 0.5 for a random time from 0 to 30 (seconds) due to a red traffic light. The probability 0.5 roughly represents the fact that one of two crossing roads will see the green light at a time, and the time 30 (seconds) represents the duration of a traffic signal light (e.g., the time from the start to the end of a green light).

B4. If a mobile chooses to turn left or around, it needs to stop there for a random time from 0 to 60 (seconds) due to the traffic signal.

B5. Each cell has a fixed link capacity of 50 BUs.

The rationale behind the assumed mobile's delay at the intersection is that there are four traffic signals at the intersection for mobiles arriving from the four directions. A traffic signal will have the red (for stop), left-turn, green (for going straight and turning right) lights in order, then back to the red light. The period from a red to the next red is $60 + \epsilon$ seconds in which the red light will last for 30 seconds, then the turn-left light will turn on for a very short time ϵ, then, finally, the green light will last for 30 seconds.

Note that the fixed capacity assumption is not necessarily true in practice. For example, CDMA systems have a softer notion of capacity, in which the capacity depends on the target interference level. This target interference level is affected by the desired error performance of the system, which can be negotiable in some case. Each simulation run starts without any pre-memorized hand-off event quadruplets. As simulations are run, quadruplets will be collected, and will affect the hand-off estimation functions $F_{HOE}(t, prev, next, T_{soj})$. Under the above assumptions, the border cells (i.e., cells <1> and <10>) will face fewer mobiles because there are no mobiles entering from the outside of the cellular system. Then, cells near the center (such as cells $< 5 >$ and $< 6 >$) will be more crowded by mobiles than those near the borders. This uneven traffic load can affect the performance evaluation of our proposed schemes, hence making it difficult to comprehend their operations correctly. So, we connected two border cells, i.e., cells <1> to <10>, artificially so that the whole cellular system forms a ring architecture as was assumed in [71] (unless stated otherwise). For the same reason, two end roads in the two-dimensional structure are also connected. For example, in Figure 4.7, the left-most (upper-most) road in cell $C1$ is connected to the right-most (lower-most) road in cell $C3$ ($C4$).

The parameters used include: $P_{HD,target} = 0.01$, $T_{start} = 1$ (second), $N_{quad} = 100$, $T_{int} = 1$ (hour), $N_{win\_days} = 1$, and $w_0 = w_1 = 1$. A frequently-used measure is the *offered load* per cell, L, which is defined as connection generation rate $\times$ connections' bandwidth $\times$ average connection lifetime, i.e.,

$$L = (1 \cdot R_{vo} + 4 \cdot (R_{vo} - 1)) \cdot \lambda \cdot 120, \tag{4.16}$$

with the above-described assumptions. The physical meaning of the offered load per cell is the total bandwidth required on average to support all existing connections in a cell.

We considered a range of the offered load up to 300. Generally, the desirable range of the offered load is less than, or equal to, the link capacity, 100 BUs, of each cell. It is undesirable to keep a cell over-loaded (i.e., the offered load is > 100) for an extended period of time, and in such a case, the cell must be split into multiple cells to increase the total system capacity. However, cells can get over-loaded temporarily. Suppose a mobile user's connection request is blocked once. Then, s/he is expected in most cases to continue to request a connection establishment until it is successful or s/he gives up. This likely behavior of mobile users will affect the offered load. Near the offered load $= 100$, P_{CB} will be around, or larger than, 0.1 in most cases, due to some reserved bandwidth for hand-offs, and in such a situation, if each connection-blocked user attempts to make a connection about 5 times, then the offered load will increase to about 150 in a very short time. Likewise, there might be some cases with the offered load of 300. This possible situation can be interpreted as a positive-feedback effect for increase in the offered load. We consider the large values of offered load such as 300, since even for these large offered loads, our goal to keep P_{HD} below a target value should be achieved.

In the following, we consider stationary traffic/mobility in one-dimensional environment with constant new connection generation rate λ and mobile speed range $[SP_{min}, SP_{max}]$ unless stated otherwise. Two cases of user mobility are considered: high user mobility with $SP_{min} = 80$ & $SP_{max} = 120$, and low user mobility with $SP_{min} = 40$ & $SP_{max} = 60$. For the stationary case, $T_{int} = \infty$ is used since the speed range and the offered load do not vary during each simulation run; so, $N_{days\_win} = 1$ is meaningless.

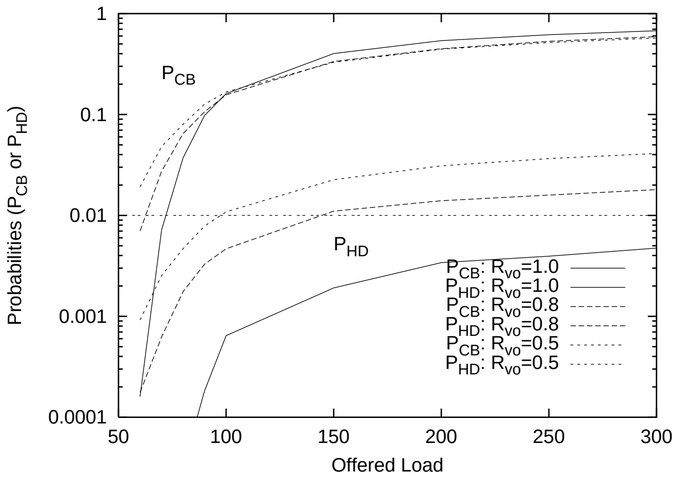
4.7.2 Static Reservation

First, we consider the performance of the static reservation scheme, called **STATIC**, as a reference (for comparison). Figure 4.8 plotted P_{CB} and P_{HD} as the offered load increases for (a) high user mobility and (b) low user mobility when $B_r = 10$, i.e., 10 BUs are reserved permanently for hand-offs in each cell. Three different values of the voice ratio R_{vo} are examined: $R_{vo} = 1.0, 0.8$, and 0.5. The performance of this static scheme, in terms of both probabilities, is found to depend heavily on the voice ratio, user mobility, and offered load. Examples are:

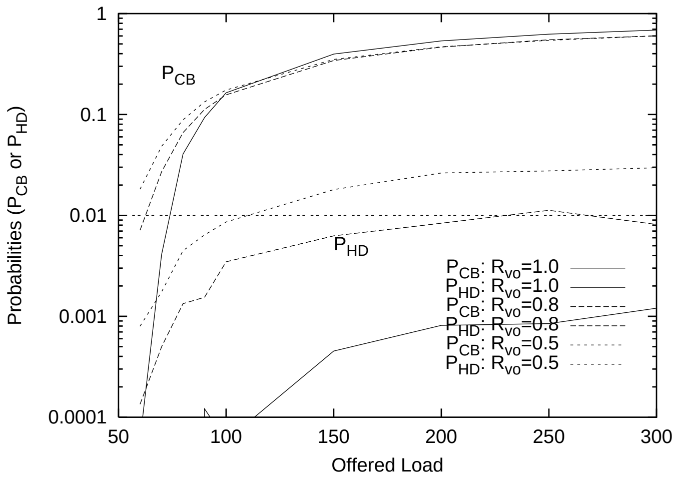
1. Static reservation of 10 BUs suffices to achieve the design goal for $R_{vo} = 1.0$, but is not enough for $R_{vo} = 0.5$;

2. For $R_{vo} = 0.8$, 10-BU reservation seems enough for low user mobility as shown in Figure 4.8 (b), but not enough for high user mobility as shown in Figure 4.8 (a); and

3. For $R_{vo} = 0.8$ and high user mobility, 10-BU reservation seems not enough for a highly over-loaded case (i.e., $L > 150$), but enough for the other case (i.e., $L < 150$). Moreover, for $R_{vo} = 1.0$, 10-BU reservation seems more than enough (i.e., over-reserved) for the under-loaded case (i.e., $L < 100$) since the observed P_{HD} value



(a) high user mobility



(b) low user mobility

Figure 4.8: P_{CB} and P_{HD} vs. offered load: static reservation with $B_r = 10$ BUs.

is too small (< 0.001 for high user mobility, and < 0.0001 for low user mobility), compared to $P_{HD,target} = 0.01$.

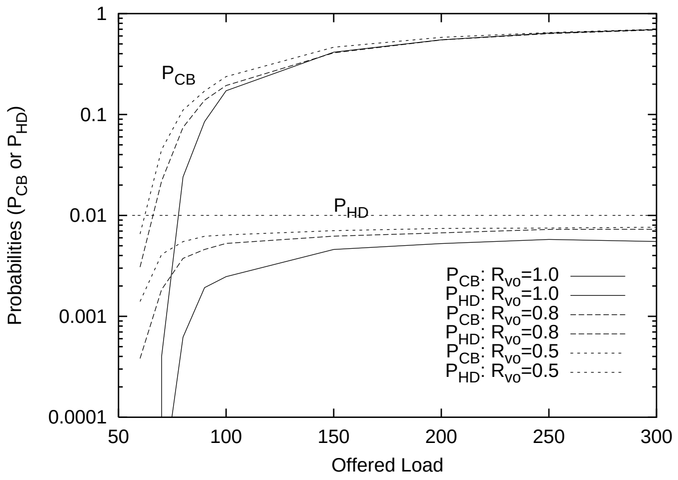
The voice ratio, mobile user speed, and offered load could in reality be any value and can even fluctuate. Hence, our goal cannot be achieved with static reservation, necessitating some form of adaptive reservation.

4.7.3 Adaptive Reservation without Path Information

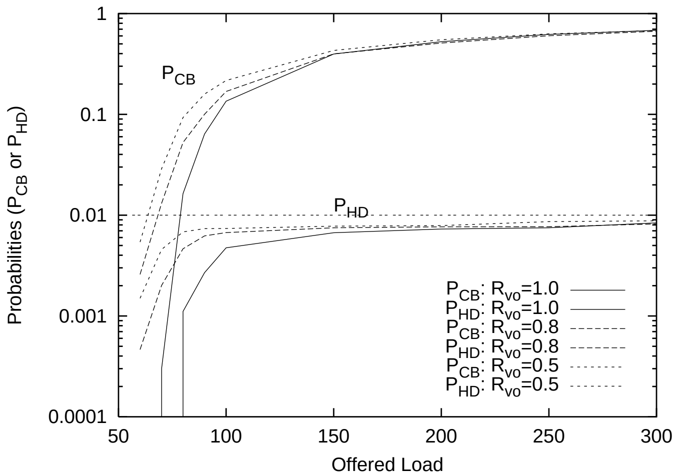
We first consider the performance of **AC3**, which is claimed to be the best without path information among the three alternatives. Figure 4.9 shows P_{CB} and P_{HD} as the offered load increases for (a) high user mobility and (b) low user mobility. For the entire range of the offered load we examined, P_{HD} is observed to be less than, or equal to, our target $P_{HD,target}$ ($= 0.01$) irrespective of user mobility and voice ratio. Moreover, for given user mobility and voice ratio, the difference between P_{CB} and P_{HD} in the plot (of log scale) is getting smaller as the offered load decreases. This means that, as the offered load decreases, the BSs reserve less bandwidth. This is desirable as long as P_{HD} stays below the target value as shown in the graphs.

Adaptive reservation patterns while varying the offered load are plotted in Figure 4.10 with the average target reservation bandwidth B_r in each cell and the average bandwidth B_u used by the existing connecitons in each cell. As the offered load increases, B_r in a cell increases monotonically, meaning that the target reservation bandwidth is controlled based on the offered load. The target reservation bandwidth gets saturated at the over-loaded region, because for the entire over-loaded region, regardless of the exact offered load value, the number of establishable connections will be limited by the link capacity. Our adaptive scheme reserves the bandwidth depending on the existing connections in adjacent cells, and hence the amount of the target reservation bandwidth will be almost the same for the entire over-loaded region.

We also observe that the target reservation bandwidth increases as the voice ratio R_{vo} decreases since the more video connections exist, the more bandwidth is needed. The average bandwidth used B_u is inversely proportional to the average target reservation bandwidth B_r since the reserved bandwidth can be used for handed-offs only. The reason why the sum of B_u and B_r is less than the capacity, 100, is that in **AC3**, the reserved bandwidths in adjacent cells are also checked for the admission test when these cells are suspected to have been over-loaded. By comparing two user mobility cases, we observe that, for similar offered load and voice ratio, the high-mobility case reserves more bandwidth than the low-mobility

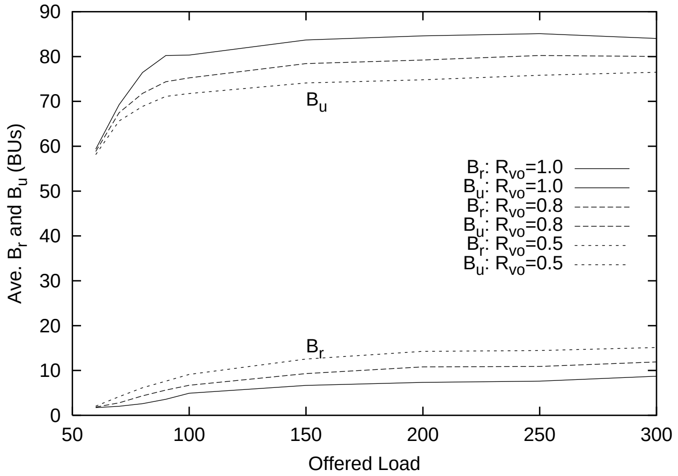


(a) high user mobility

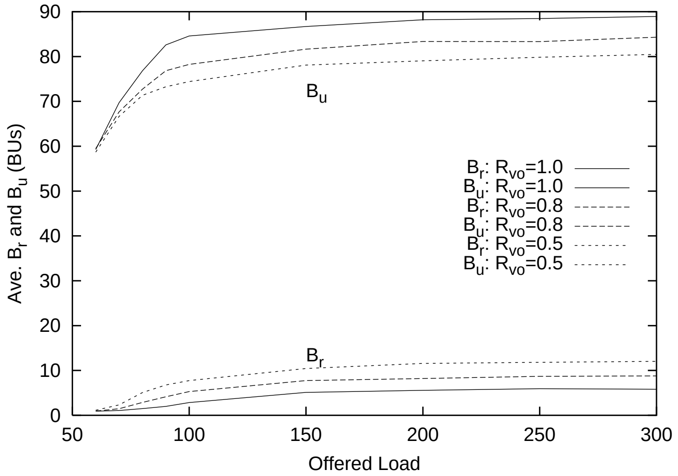


(b) low user mobility

Figure 4.9: P_{CB} and P_{HD} vs. offered load: **AC3**.



(a) high user mobility



(b) low user mobility

Figure 4.10: Average target reservation bandwidth B_r and average bandwidth used B_u vs. offered load: **AC3**.

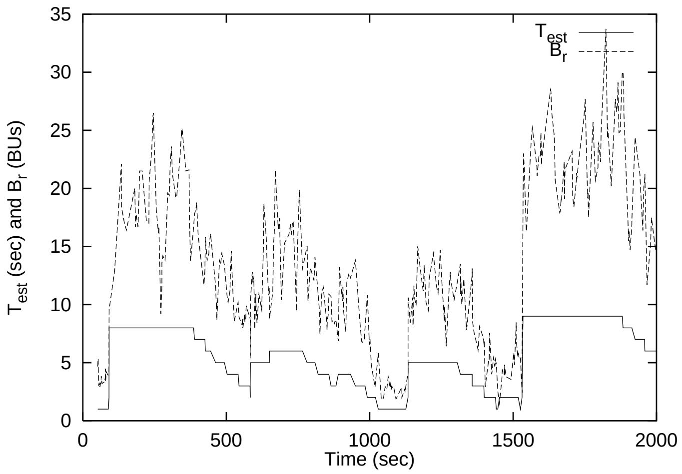
case. For the low-mobility case, the chance of hand-offs would be smaller, and hence less bandwidth needs to be reserved.

Next, let's consider the detailed operation of our scheme in each cell. Figure 4.11 shows T_{est} and B_r, starting from the beginning of a simulation run (i.e., $t = 0$) for the offered load = 300 and $R_{vo} = 1.0$ with high user mobility in (a) cell <5> and (b) cell <6>. The values of T_{est} were observed to go up and down as time passes. Note that an increase of T_{est} by one corresponds to a connection's hand-off drop. The target reservation bandwidth fluctuates between over-reservation and under-reservation, depending on the value of T_{est}. The value of T_{est} seldom stays at a possible optimum value without fluctuation for the following two reasons: (1) hand-offs could be bursty, so when there are a number of hand-off drops, it is difficult to determine whether it is due to the insufficient reserved bandwidth or bursty hand-offs; and (2) the effectiveness of the reserved bandwidth is determined some time later; that is, whether the currently-reserved bandwidth is enough or not can be determined only after some mobiles enter the cell. We also observed the fluctuations of B_r even with a temporarily-constant T_{est} as the value of B_r depends on the number and type of connections in the adjacent cells and their extant sojourn times.

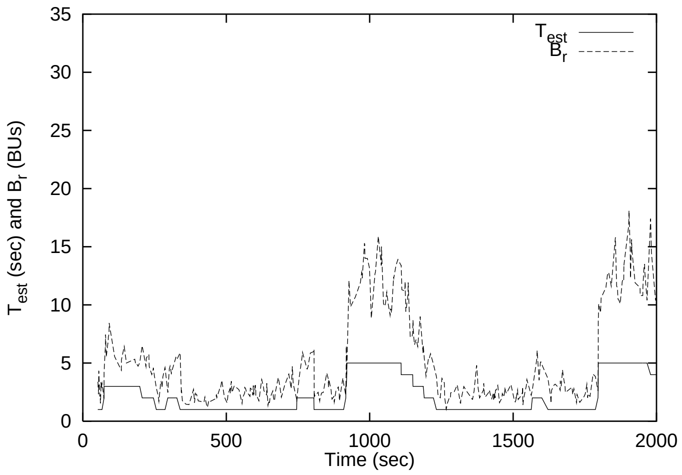
Figure 4.12 plotted P_{HD} for cells <5> and <6> while increasing time, obtained from the same simulation run used for Figure 4.11. Note that the increase of P_{HD} corresponds to hand-off drops. By comparing it with Figure 4.11, we can also observe that the increasing moments of P_{HD} and T_{est} coincide exactly as they should be. P_{HD} peaks over the target value $P_{HD,target}$ (= 0.01) sometimes, but eventually goes below 0.01. Near the starting point, i.e., $t = 0$ (sec), our scheme seems not working well because the simulation starts without any pre-memorized hand-off event quadruplets $(T_{event}, prev, next, T_{soj})$ and with $T_{est} = T_{start} = 1$ (sec). As time goes on, the chance of peaks over the target value is low because (1) hand-off event quadruplets are observed and used for the hand-off estimation functions; (2) T_{est} is adapted; and (3) the effect of some more hand-off drops out of a large number of hand-offs is minor due to an averaging effect.

4.7.4 Comparison among Alternatives without Path Information

We now comparatively evaluate the performance of three difference schemes: **AC1**, **AC2**, and **AC3**. Figure 4.13 plots P_{CB} and P_{HD}. First, in terms of P_{CB}, three schemes work almost the same even though **AC1** has the smallest P_{CB} — with small differences – for the entire offered loads we examined. On the other hand, in terms of P_{HD}, **AC2** and **AC3** work almost the same, and **AC1** is worse. Our goal is not achieved in a highly



(a) cell <5>



(b) cell <6>

Figure 4.11: T_{est} and B_r vs. time when the offered load is 300 and $R_{vo} = 1.0$ for high user mobility: **AC3**.

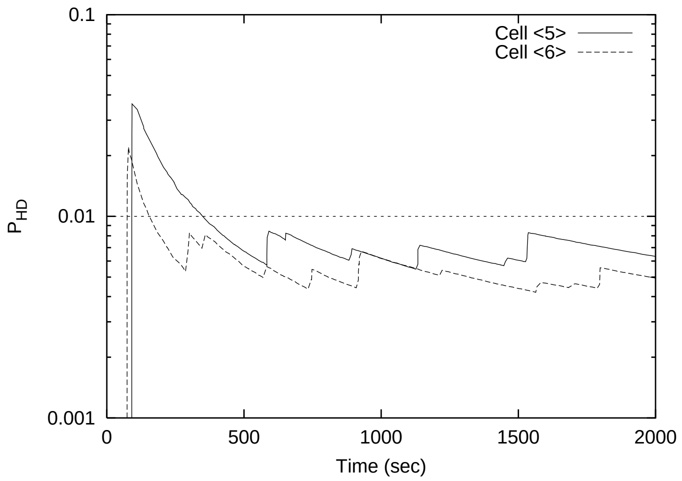
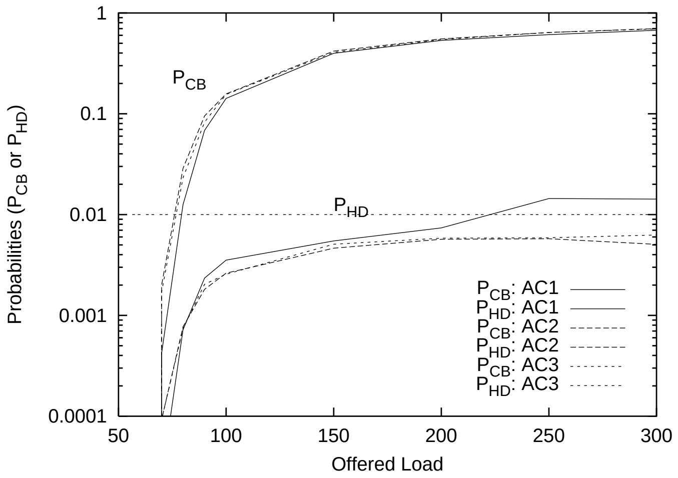


Figure 4.12: P_{HD} at cells <5> and <6> vs. time when the offered load is 300 and $R_{vo} = 1.0$ for high user mobility: **AC3**.

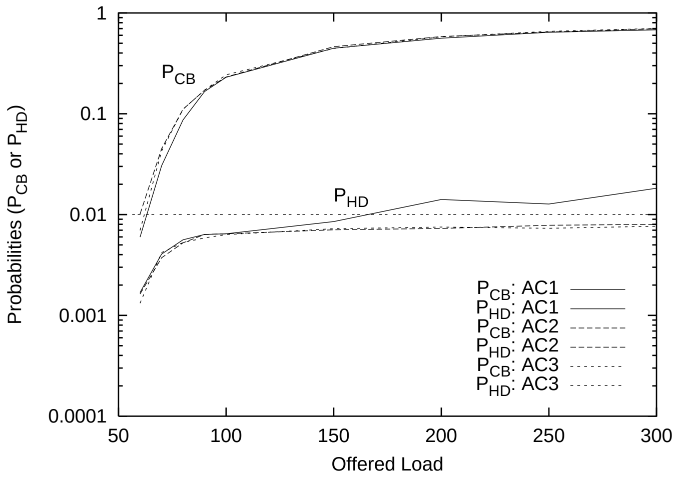
over-loaded region (say, $L > 150$) for **AC1**. P_{HD} does not exceed 0.02 even at the offered load of 300, which is good because this small violation ratio might be tolerable in most practical applications.

Now, we consider the complexity of these schemes measured in average number of B_r calculations for the admission test of a new connection request ($= N_{calc}$). Note that, to calculate B_r in a cell, its BS needs to communicate with BSs in all adjacent cells. Figure 4.14 shows that N_{calc} for **AC1** is 1, irrespective of the offered load because only the BS of the cell in which the new connection was requested has to calculate B_r while $N_{calc} = 3$ for **AC2** because BSs in all adjacent cells are required to calculate B_r. For **AC3**, which is a hybrid of **AC1** and **AC2**, N_{calc} =1 for low offered load, but it starts to increase at about $L = 80$. However, the value is observed to be less than 1.5 in all of our simulations, i.e., less than a half of that of **AC2**. The complexity increase could be larger for two-dimensional cellular structures. Because **AC3** works almost the same as **AC2** in terms of P_{CB} while keeping P_{HD} below the target with a lower complexity according to our simulation results, we conclude that **AC3** is a better choice than **AC2**.

Now, we compare **AC1** with **AC3** by examining each cell when the system is over-loaded. Table 4.2 shows the state of each cell at the end of simulations when the offered load is 300 and $R_{vo} = 1.0$ for high user mobility with (a) **AC1** and (b) **AC3**. The first column represents the cell number, the second is P_{CB}, the third is P_{HD}, the fourth is the value of T_{est}, the fifth is the value of B_r, and the sixth is the value of B_u, all at the end of the simulations. From Table 4.2 (b), **AC3** is found to work similar throughout all cells in

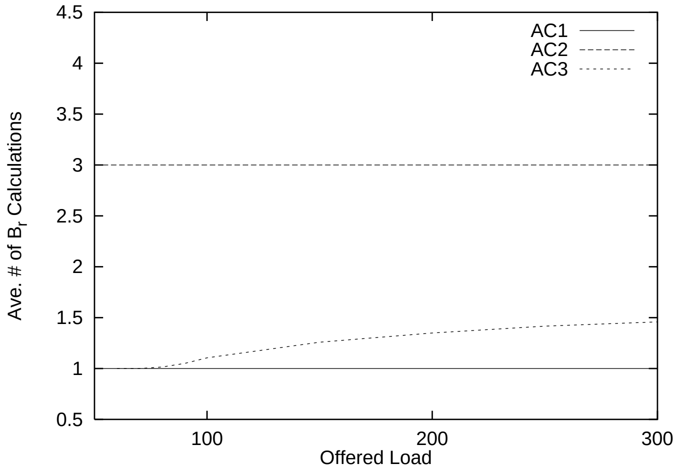


(a) $R_{vo} = 1.0$

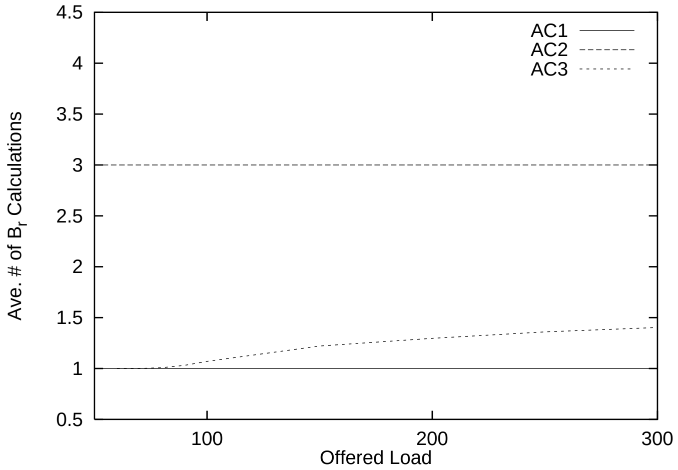


(b) $R_{vo} = 0.5$

Figure 4.13: Comparison among **AC1**, **AC2**, and **AC3** using P_{CB} and P_{HD} vs. offered load for high user mobility.



(a) high user mobility



(b) low user mobility

Figure 4.14: Comparison among **AC1**, **AC2**, and **AC3** using average number of B_r calculations for an admission test vs. offered load.

Cell	P_{CB}	P_{HD}	T_{est}	B_r	B_u
1	1.28e-1	5.10e-3	1	1.04	89
2	9.59e-1	1.30e-2	45	102	89
3	9.57e-1	1.30e-2	45	93.6	97
4	2.41e-1	6.01e-3	1	2.13	93
5	9.79e-1	2.58e-2	45	89.9	99
6	3.83e-1	7.12e-3	1	1.97	87
7	9.80e-1	2.90e-2	45	85.8	100
8	2.75e-1	6.97e-3	2	3.22	94
9	9.57e-1	1.80e-2	45	10.4	88
10	9.26e-1	1.51e-2	45	102	81

(a) **AC1**

Cell	P_{CB}	P_{HD}	T_{est}	B_r	B_u
1	6.23e-1	6.53e-3	2	5.63	97
2	6.66e-1	6.57e-3	2	5.44	84
3	7.30e-1	7.54e-3	10	22.5	75
4	8.06e-1	7.80e-3	7	19.2	73
5	7.65e-1	7.31e-3	10	24.2	74
6	5.65e-1	6.74e-3	5	11.3	66
7	6.17e-1	6.42e-3	3	5.81	89
8	8.10e-1	6.87e-3	3	9.18	87
9	7.68e-1	7.42e-3	8	20.1	78
10	6.46e-1	5.41e-3	1	4.50	90

(b) **AC3**

Table 4.2: Status in each cell at the end of simulations when the offered load is 300 and $R_{vo} = 1.0$ with high user mobility.

	AC1		**AC3**	
Cell	P_{CB}	P_{HD}	P_{CB}	P_{HD}
1	0.	0.	5.61e-02	0.
2	5.24e-01	6.63e-03	5.38e-01	4.58e-03
3	9.66e-01	1.09e-02	7.83e-01	7.38e-03
4	2.84e-01	5.98e-03	7.06e-01	5.78e-03
5	9.45e-01	4.15e-02	6.35e-01	5.93e-03
6	4.17e-01	7.28e-03	7.44e-01	7.96e-03

Table 4.3: Status in each cell at the end of simulations when the offered load is 300, $R_{vo} = 1.0$, and all mobiles follow one direction with high mobility.

terms of P_{CB} while meeting the constraint $P_{HD} \leq P_{HD,target}$. B_r can change dramatically depending on the traffic condition in adjacent cells even with the same T_{est} as observed in Table 4.2. However, according to Table 4.2 (a) of **AC1**, the performance of each cell is found to fluctuate greatly, i.e., the performance in terms of P_{CB}, P_{HD}, T_{est}, and B_r drastically differ in roughly every two cells. This is not fair to those mobiles which want to establish new connections in cells with a very high P_{CB}, e.g., cells <2>, <3>, <5>, <7>, <9>, and <10> in the table. More importantly, P_{HD}'s of these cells are not bounded. This phenomenon was anticipated as explained in Section 4.4.4 when the admission test checks the current cell only as was done in **AC1**.

Table 4.3 shows the status of each cell at the end of simulations with a different mobility pattern when the offered load = 300 and $R_{vo} = 1.0$. For these simulations, the direction of mobiles are not chosen randomly. Instead, all mobiles follow the direction from cell <1> to cell <10>. Moreover, two border cells, i.e., cells <1> and <10>, are disconnected. Now, cell <1> won't have any incoming mobiles from adjacent cells. Naturally, P_{HD} will be zero at cell <1>. For **AC1**, we observe a behavior similar to that in Table 4.2 (a). Especially, because cell <1> doesn't care about the status of cell <2>, the BS of cell <1> accepted all new connection requests, hence $P_{CB} = 0$. Cell <2> also doesn't care about the status of cell <3>. These make cell <3> over-crowded, and eventually result in a very high P_{CB} (near 1) and over-target P_{HD} at cell <3>. This type of patterns appears every other cell as shown in the table. On the other hand, for **AC3**, cell <1> cares about cell <2>, and blocks some new connection requests. Every cell <i> cares about the status of cell <$i+1$>. Eventually, balanced performance is observed over the entire system while every cell meeting the constraint on P_{HD}.

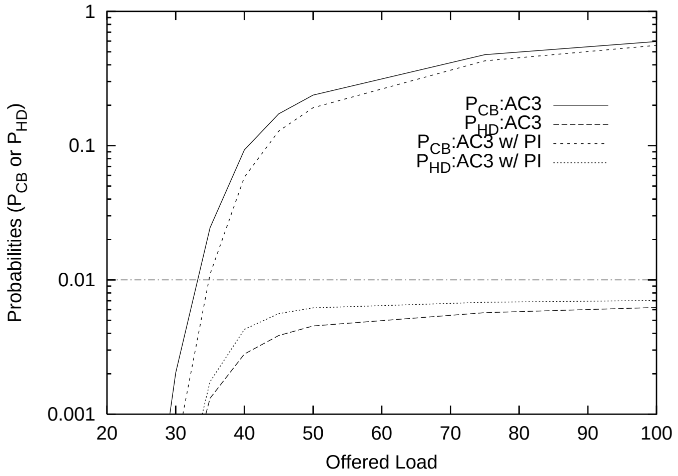
4.7.5 Adaptive Reservation with Path Information

In the real world, only a subset of mobiles will be equipped with navigation systems. However, we compare two extreme cases to evaluate the advantages of path information, i.e., (1) path information is not available for any mobile, and (2) path information is available for every mobile. For this comparison, we consider two-dimensional cellular environment since the path information will not be very advantageous for the case of one-dimensional environment. Figure 4.15 shows P_{CB} and P_{HD} of both cases as the offered load increases. First of all, P_{HD} is bounded for both cases, thus achieving the design goal. As expected, **AC3 w/ PI** outperforms **AC3** in terms of P_{CB}, i.e., more new connections can be admitted into the network with **AC3 w/ PI**. Even though the difference is not significant, we can determine that path information is quite advantageous.

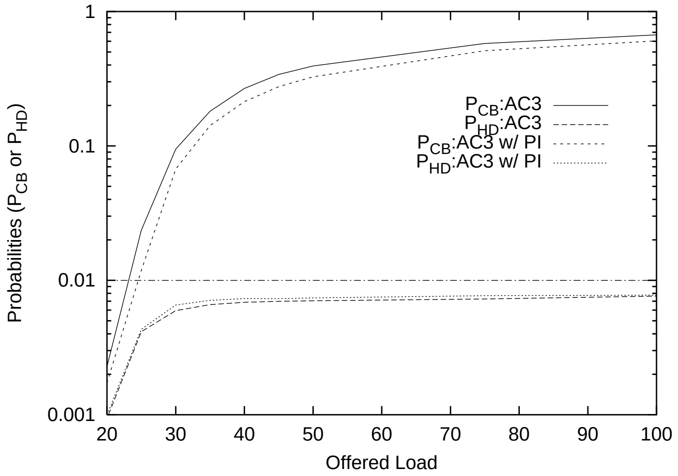
Now, we consider the computation complexity of the admission control schemes by comparing the numbers of numerical operations (including summations and multiplications) and comparisons for an admission decision. Comparisons include decisions such as if t_{soj} is larger than a value in summations of Eqs. (4.4) and (4.8). We did not include the complexity and cost for the interface between the navigation systems and the network. Figure 4.16 shows the average numbers of numerical operations and comparisons for an admission decision for both cases. We observe the complexity gap is significant. Interestingly, the complexity for $R_{vo} = 0.5$ is much smaller than that for $R_{vo} = 1.0$. This is because the admission decision complexity depends on the number of existing connections in the current and adjacent cells, but for $R_{vo} = 0.5$, there are fewer existing connections in the system on average for a given offered load.

4.7.6 Time-Varying Traffic/Mobility Case

We now vary the connection generation rate λ and speed range $[SP_{min}, SP_{max}]$ over time. Each simulation is run for two days in simulation time. Figure 4.17 (a) shows time-varying averages of mobiles' speeds and offered loads. First, for a given value of the average speed (marked by S), the speed range is given by $[S - 20,\ S + 20]$ (km/h). Second, the original offered load (marked by L_o) is the traffic load from the new connections generated, which is the offered load L defined in Eq. (4.16). In this time-varying case, a blocked connection request will be re-requested with probability $1 - 0.1N_{ret}$ after waiting 5 seconds, where N_{ret} is the number of times a connection request has been made. So, depending on P_{CB}, the actual offered load L_a will vary, i.e., the larger P_{CB}, the larger L_a. From the figure, we observe that the values of L_a for different schemes are different when the

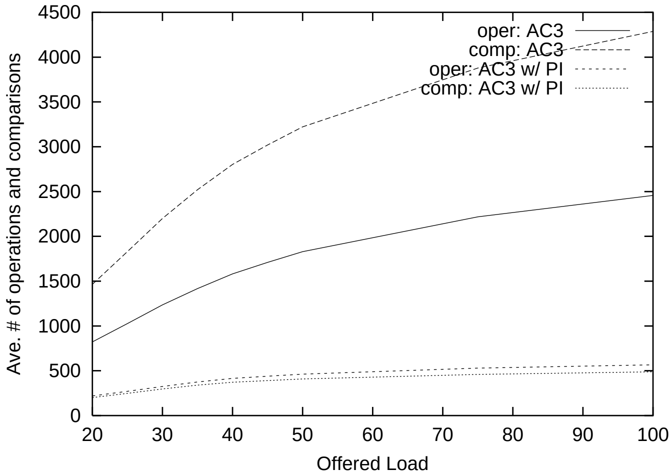


(a) $R_{vo} = 1.0$

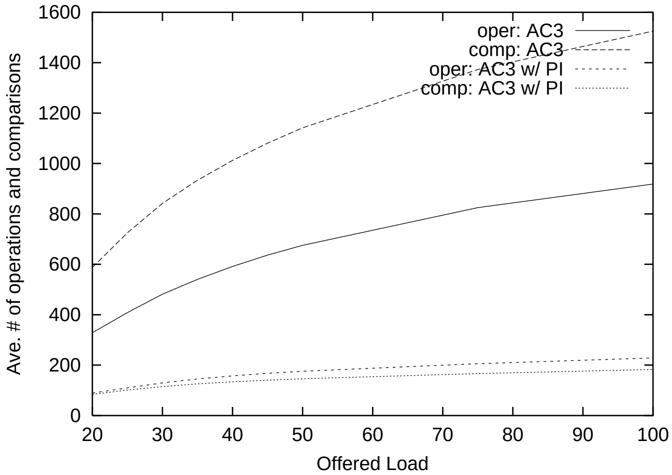


(b) $R_{vo} = 0.5$

Figure 4.15: Comparison between **AC3** and **AC3 w/ PI** using P_{CB} and P_{HD} vs. offered load.

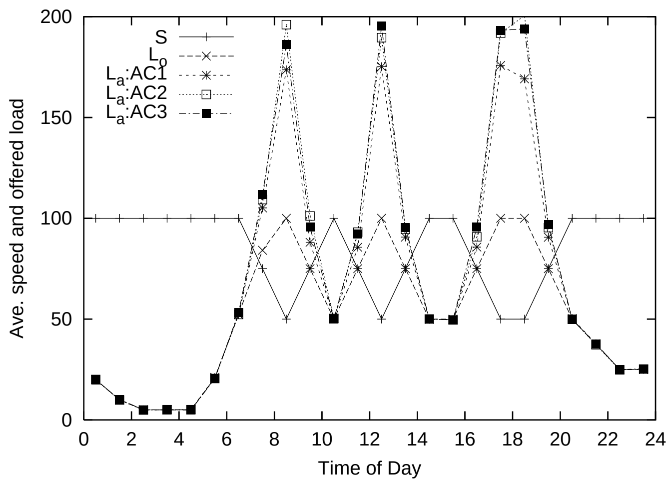


(a) $R_{vo} = 1.0$

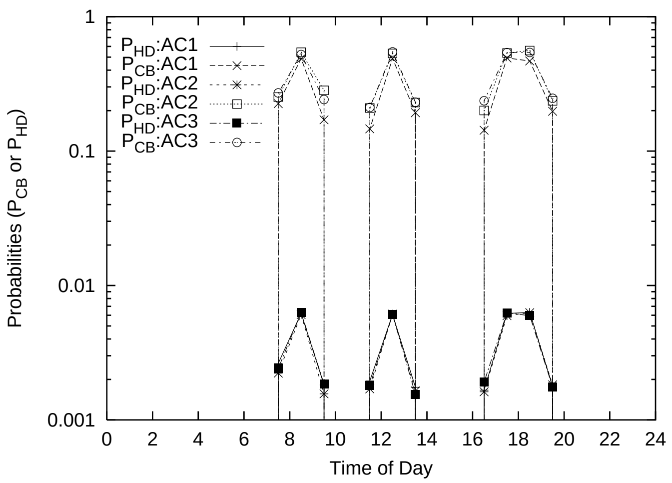


(b) $R_{vo} = 0.5$

Figure 4.16: Comparison between **AC3** and **AC3 w/ PI** using the average number of numerical operations and comparisons vs. offered load.



(a) speed and offered load



(b) probabilities

Figure 4.17: Simulation results for the time-varying case: (a) mobiles' average speed and offered load vs. time of day and (b) P_{CB} and P_{HD} vs. time of day.

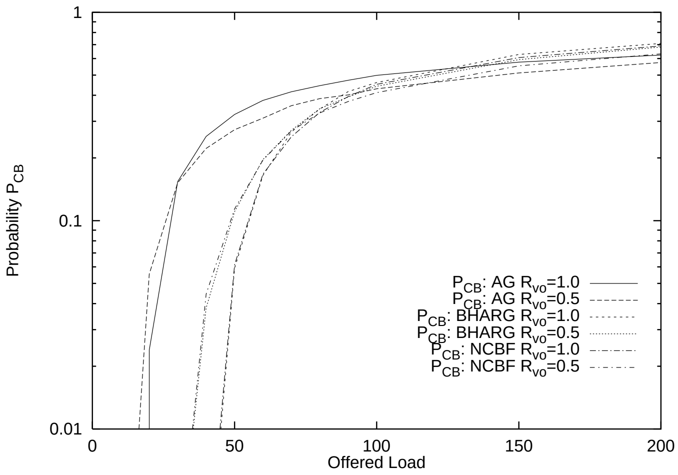
system is highly-loaded even with the same L_o. Note that the fluctuations of the offered load and speed represent the reality, that is, the offered load peaks during rush hours (e.g., around 9 a.m., 1 p.m., and 5–6 p.m.) at low speeds. Figure 4.17 (b) shows P_{CB} and P_{HD} over time of day for three different schemes. We don't consider the schemes with path information here. The probability samples represent the average probability during the corresponding one-hour period, i.e., P_{CB} at $t = 8.5$ represents the average over the time interval $[8, 9]$. First, we observe that outside the peak hour regions, both P_{CB} and P_{HD} are negligibly small. During the peak hours, P_{HD} is almost the same for different schemes, and bounded by $P_{HD,target}$ ($= 0.01$). On the other hand, P_{CB} of **AC1** is found to be lower than that of the other two schemes, and the differences between P_{CB}'s of **AC1** and **AC3** are larger compared to those from the stationary case in Figure 4.13. This is due to the positive feedback effect of the offered load increase; that is, from the original offered load, the difference between **AC1** and **AC3** could be small, but this small difference could be amplified through the re-trials of each blocked connection request.

According to the results of the time-varying case, **AC1** is the best because it yields the lowest P_{CB} while meeting our goal. For this time-varying case, we considered only a regular traffic pattern: a high offered load for relatively short peak-hour periods (of 1 or 2 hours). However, **AC1** may have undesirable behaviors as previously observed in the time-invariant case, because there might be unexpected irregular traffic and mobility patterns in the real world. **AC3** was found to be robust in many different scenarios with relatively low complexity (up to 1.5 times that of **AC1** in our simulations). So, **AC3** and **AC3 w/ PI** are the most favorable among the schemes considered.

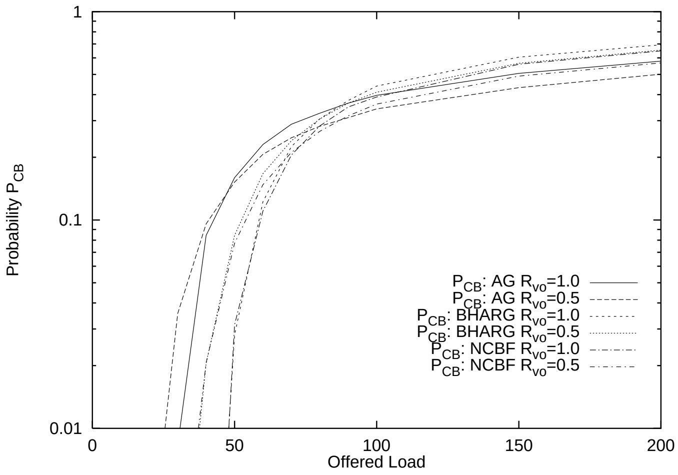
4.7.7 Comparison of AG, BHARG, and NCBF

Now, we compare three admission-control schemes based on per-connection bandwidth reservation, and then compare our best scheme **AC3** with **NCBF** which is claimed to be the best per-connection bandwidth reservation scheme, and finally compare **AC3** with **NAG**. The one-dimensional structure is considered for all comparisons. The two-dimensional structure is also considered for the comparison of **AC3** and **NAG**, which needs a more careful comparison, in Section 4.7.10. **NAG** was also claimed to be superior to the static bandwidth reservation scheme, while showing that the static reservation scheme cannot meet the design goal [71].

Figure 4.18 shows P_{CB} of three per-connection bandwidth reservation schemes as the offered load increases for the voice ratio $R_{vo} = 0.5$ and 1.0. It was observed that P_{HD}

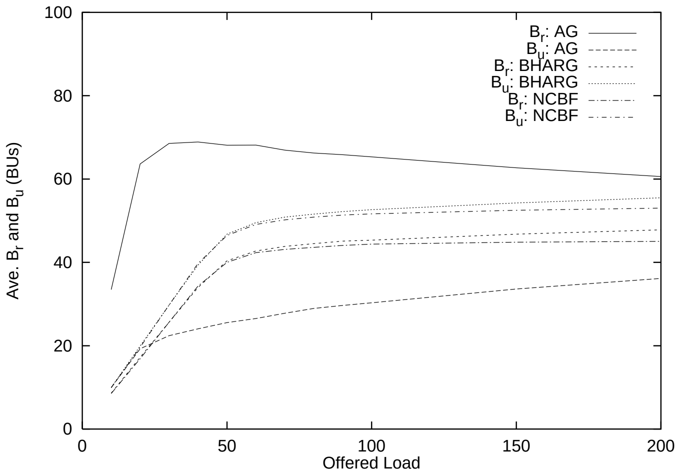


(a) high mobility

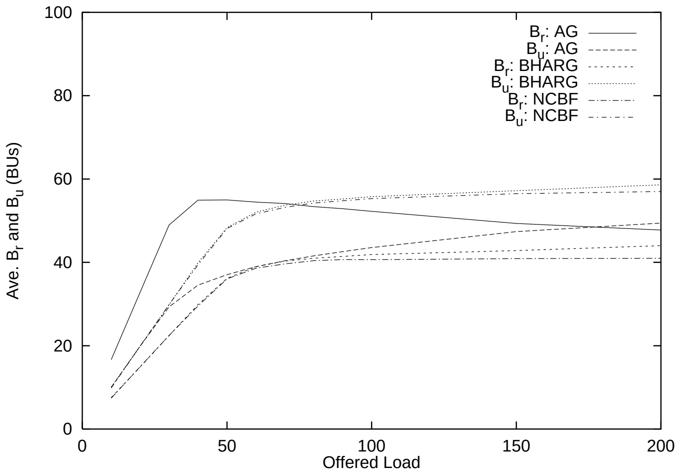


(b) low mobility

Figure 4.18: Comparison of **AG**, **BHARG**, and **NCBF** using P_{CB} vs. offered load.



(a) high mobility



(b) low mobility

Figure 4.19: Comparison of **AG**, **BHARG**, and **NCBF** using the average B_r and B_u vs. offered load for $R_{vo} = 1.0$.

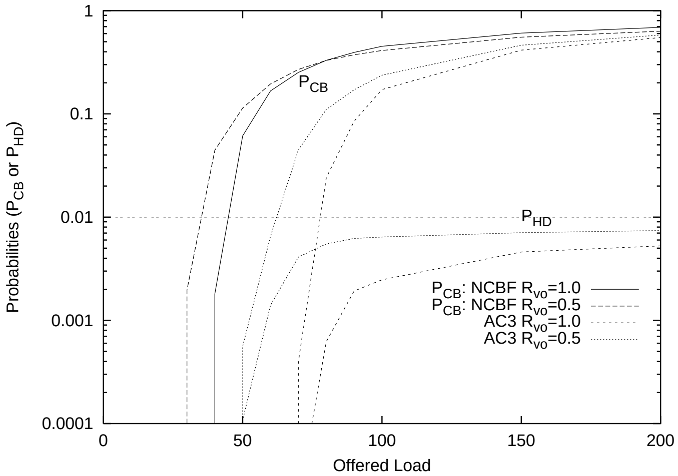
of all the three are (virtually) zero irrespective of the offered load, voice ratio, and user mobility, and thus omitted in the plots. In fact, we never observed any hand-off drops with **AG** and **NCBF** as it should be with **AG** especially, and observed only two hand-off drops throughout the whole simulations involving more than 100,000 hand-offs for each simulation run. It should be noted that the hand-off drops are (virtually) eliminated at the expense of blocking a large number of new connection requests even in lightly-loaded situations. The fact that P_{CB} is larger than 0.1 even for $L = 30$ with **AG** and for $L = 60$ with the other two schemes where $C = 100$ implies that these schemes severely under-utilize the link capacity. P_{CB}'s of **BHARG** and **NCBF** are observed to be less than that of **AG**. That is basically because **AG** reserves more bandwidth for hand-offs. In fact, P_{CB}'s of **BHARG** and **NCBF** get closer to that of **AG** for the low mobility case since the average number of cells within the mobility specification used for **AG** is small in this case. Comparing **BHARG** and **NCBF**, we observe that P_{CB}'s of both schemes are almost the same in a lightly-loaded region, but that of **NCBF** is a little bit smaller than that of **BHARG** in a heavily-loaded region. For example, for $R_{vo} = 1.0$ and high (low) mobility, P_{CB} of **BHARG** is about 0.80 (0.79) while that of **NCBF** is about 0.77 (0.74) at the offered load $L = 300$. This result is not easy to understand since this implies that **NCBF** reserves less bandwidth for hand-offs. Intuitively, **NCBF** is expected to reserve more bandwidth for hand-offs since it checks the bandwidth availability in the predicted next cell, and reserves the bandwidth before admitting a newly-requested connection. This will become clear when we examine the average reserved bandwidth with the next figure.

Figure 4.19 shows the average (target) reservation bandwidth B_r and utilized bandwidth B_u by the existing connections as the offered load increases for $R_{vo} = 1.0$. Note that B_r is a target for **BHARG** while it is the real reserved bandwidth for **AG** and **NCBF**. First, **BHARG** and **NCBF** work desirably by reserving less bandwidth when the system is lightly-loaded, and increasing the reservation bandwidth as the offered load increases. B_u is observed to be larger than B_r throughout the whole offered loads examined. For the case of **AG**, when the system is lightly-loaded, B_r is larger than B_u because new connections will rarely be blocked in this case, and for each admitted connection, its bandwidth is reserved in all cells within the mobility specification, which includes, on average, more than two cells in our experiments. The number of cells within a connection's mobility specification is dependent on the connection's lifetime and the mobile's speed. Accordingly, B_r is found to be smaller for Figure 4.21 (b) with low user mobility. Contrary to intuition, B_r starts to decrease beyond a threshold offered load even though B_u continues to increase. This phenomenon can be explained as follows. After the threshold offered load, the degree of blocking new connection requests becomes severer, implying that a connection with a smaller mobility specification (i.e., a smaller number of cells in its mobility specification) will have a better chance to be admitted. As the offered load increases, connections with large mobility specifications will be more likely to be blocked, and hence, there will be more connections with small mobility specifications in the system. The smaller the mobility specification, the smaller total bandwidth will be reserved throughout the system. So, the bandwidth reservation will decrease with the increase in offered load.

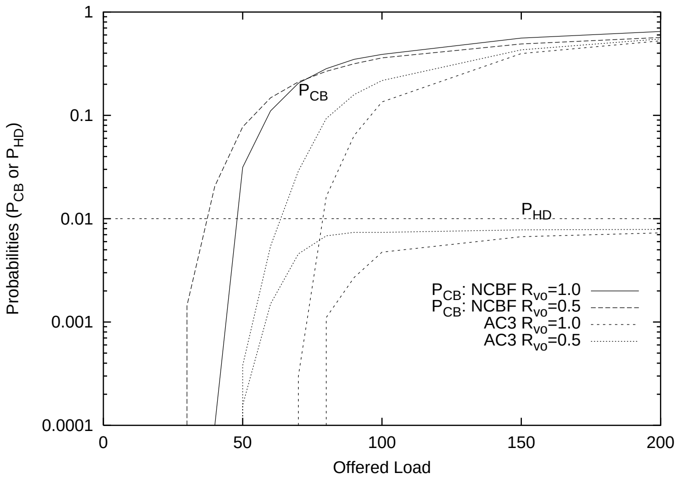
Both B_u and B_r of **NCBF** are also observed to be smaller than those of **BHARG** in moderately-loaded and heavily-loaded regions while they are almost the same in lightly-loaded regions. This is due to a similar reason why B_u of **AG** decreases beyond a threshold, i.e., connections which will end in the current cell without incurring any hand-off (or connections with relatively shorter connection lifetimes) will have more chance to be admitted since they don't require any bandwidth to be reserved in the predicted next cell while other connections will have less chance to be admitted. Note that **BHARG** does not differentiate these two types of connections during the admission control phase. Accordingly, **BHARG** will admit more connections requiring bandwidths to be reserved in the predicted next cells. This is why B_r of **BHARG** is larger than that of **NCBF**. Moreover, this explains why P_{CB} of **BHARG** was larger than that of **NCBF** in Figure 4.18. since the larger B_r, the larger P_{CB} generally. The reason why B_u of **BHARG** is larger also can be explained similarly. That is, with **NCBF**, the admitted connections' lifetimes are shorter than those with **BHARG**, meaning more connections in the system with **BHARG** on average for a given number of admitted connections. Even though **NCBF** admits more connections on average, the effect of shorter connection lifetimes appears stronger. Comparing these three schemes, **NCBF** is more attractive since it results in virtually no hand-off drops while achieving lower new connection blocks, i.e., less expensive in terms of bandwidth usage. In the next subsection, we compare **NCBF** with **AC3**.

4.7.8 Comparison of AC3 and NCBF

Figure 4.20 shows P_{CB} and P_{HD} of **AC3** and **NCBF** as the offered load increases for the voice ratio $R_{vo} = 0.5$ and 1.0. The P_{HD} of **AC3** is observed to be upper-bounded by the target value $P_{HD,target} = 0.01$ irrespective of the voice ratio and user mobility over the entire offered loads examined, so **AC3** attains the design goal. By comparing P_{CB} of two schemes, we can conclude that the hand-off drops of **NCBF** (and hence, the other two

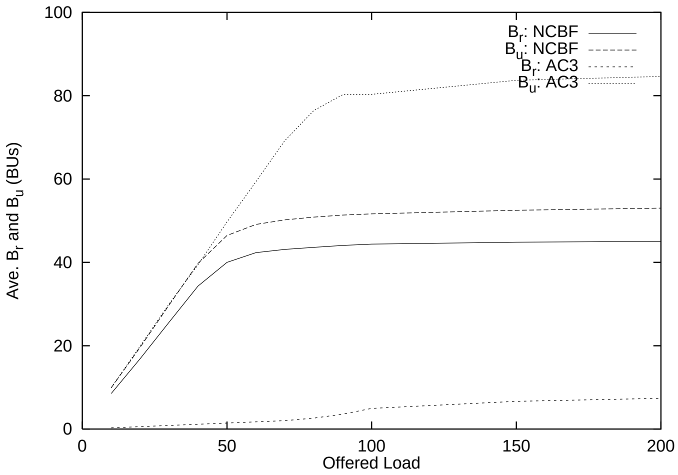


(a) high mobility

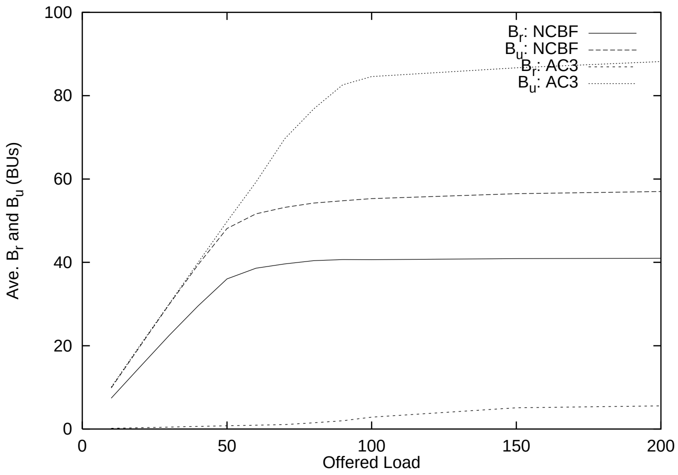


(b) low mobility

Figure 4.20: Comparison of **NCBF** and **AC3** using P_{CB} and P_{HD} vs. offered load.



(a) high mobility



(b) low mobility

Figure 4.21: Comparison of **NCBF** and **AC3** using the average B_r and B_u vs. offered load for $R_{vo} = 1.0$.

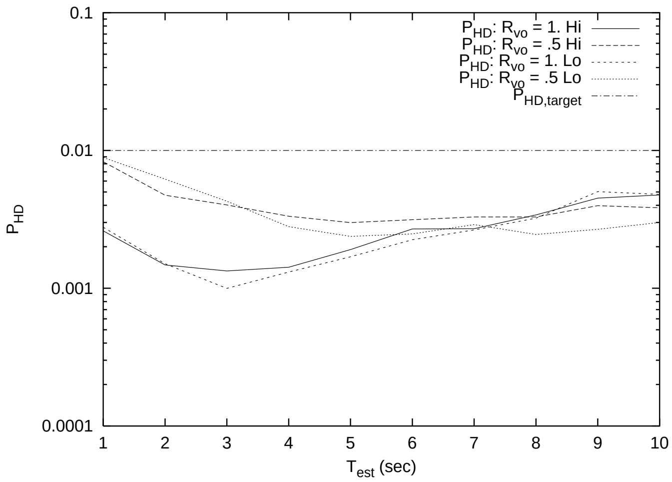
per-connection bandwidth reservation schemes) are eliminated at the expense of blocking a large number of new connection requests even in lightly-loaded situations.

This is clearer in Figure 4.21 which shows the average (target) reservation bandwidth B_r and utilized bandwidth B_u by the existing connections as the offered load increases for $R_{vo} = 1.0$. Note that B_r is a target for **AC3** while it is a real reserved bandwidth for **NCBF**. **AC3** works desirably by reserving less bandwidth when the system is lightly-loaded, and increasing the reservation bandwidth as the offered load increases. B_u is observed to be larger than B_r throughout the whole offered loads examined. **NCBF** has the same tendency, but the problem is that it reserves too much bandwidth compared to **AC3**, so less bandwidth is utilized, i.e., B_u is smaller. This is why P_{CB} of **NCBF** is much larger than that of P_{HD}. From all the above observations, one can conclude that guaranteeing (virtually) no hand-off drops through per-connection reservation is too expensive to be practically useful. Since wireless resources are scarce in general, per-connection bandwidth reservation schemes are practically unattractive. In practice, the service provider may support any of these three schemes (e.g., **NCBF** since it is the best among the three) as an option available to customers who are willing to pay the high price.

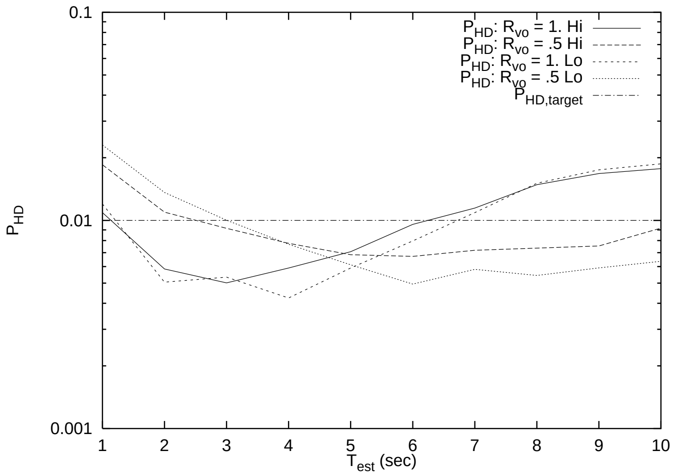
4.7.9 Comparison of AC3 and NAG: One-Dimensional Case

Now, we compare **AC3** and **NAG**, both of which have the same design goal to keep P_{HD} below a given target value. First, we consider the performance of **NAG** to show the degree of its dependency on the choice of T_{est}. Figure 4.22 plots the P_{HD} of **NAG** with different values of T_{est} for the offered load (a) $L = 100$ and (b) $L = 200$, where 'Hi' and 'Lo' in the figures represent high and low user mobility, respectively. Four different (R_{vo}, mobility) pairs were considered. First, from Figure 4.22 (a), all ranges of T_{est} satisfy the design goal for $L = 100$. Next, from Figure 4.22 (b), **NAG** is observed to achieve the design goal only with certain values of T_{est} for $L = 200$. Especially, this plot of **NAG** shows that the trade-off between large and small T_{est}'s, which was discussed at the end of Section 4.5.2.

The smaller P_{CB} the better as long as $P_{HD} \leq P_{HD,target}$. The values of P_{CB} were observed to be almost constant for all the examined values of T_{est} even though the corresponding graphs are not included here due to lack of space. So, the smaller P_{HD} the better in this case. The problem is that the dependency of P_{HD} on T_{est} is a function of user mobility and R_{vo}. Especially, the optimal T_{est} which achieves the smallest P_{HD} depends greatly on R_{vo}. We also conducted the same experiment to obtain Figure 4.22 for capacity



(a) $L = 100$



(b) $L = 200$

Figure 4.22: P_{HD} vs. estimation time T_{est}: **NAG**.

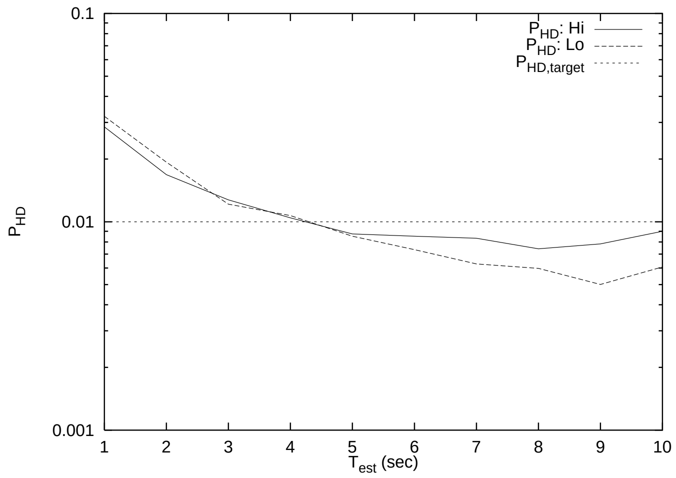
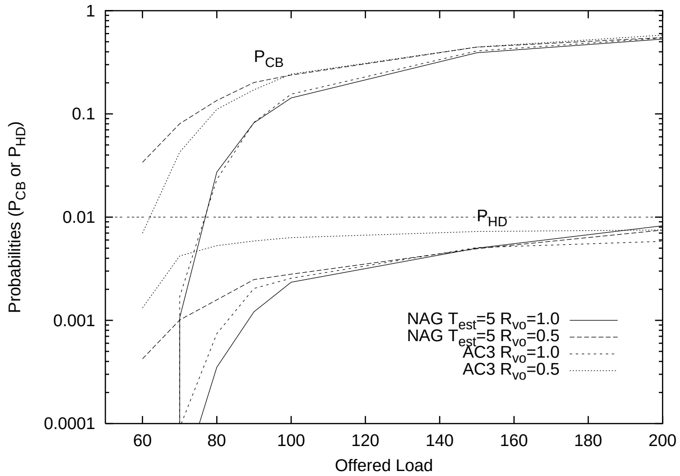


Figure 4.23: P_{HD} vs. estimation time T_{est} for $C = 20$ and $L = 40$: **NAG**.

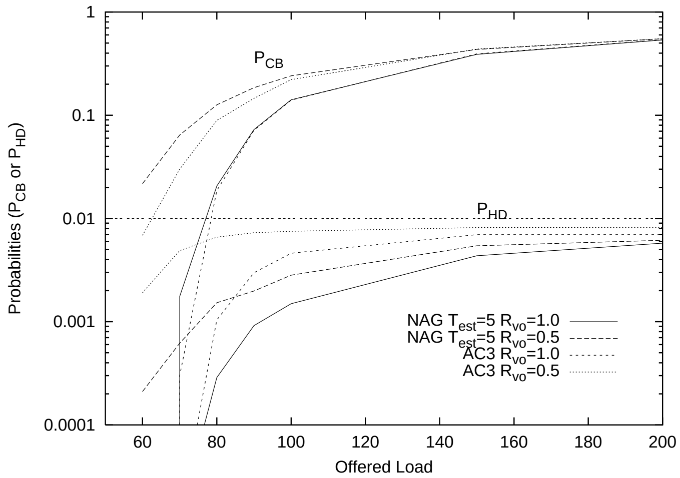
$C = 20$, $L = 40$, and $R_{vo} = 1.0$, and found that the optimal T_{est} depends also on the link capacity as shown in Figure 4.23. Determination of the optimal T_{est} should involve a form of experiment similar to the above. However, the optimal T_{est} depends on user mobility, voice ratio, and link capacity. Moreover, user mobility and voice ratio are actually time-varying, so it is difficult to determine the best value of T_{est} for a system. For further experiments, we choose $T_{est} = 5$ (sec), which is about the average of four different optimal T_{est}'s for four different cases in Figure 4.22, for the one-dimensional environment.

Figure 4.24 plots P_{CB} and P_{HD} as the offered load increases for **NAG** with $T_{est} = 5$ and **AC3**. Both schemes are found to achieve the design goal for the most of offered loads examined. As long as the design goal is met, which of the two achieves a smaller P_{HD} does not matter. In terms of P_{CB}, **AC3** performs better than **NAG** for the lightly-loaded region, and worse for the heavily-loaded region. For a very heavily-loaded region, both schemes yield about the same P_{CB}, but **NAG** is slightly better; the rightmost points of the graphs for $R_{vo} = 1.0$ are: (1) high-mobility: 0.695 (**AC3**) and 0.672 (**NAG**); and (2) low-mobility: 0.682 (**AC3**) and 0.676 (**NAG**).

Figure 4.25 shows the average utilized bandwidth B_u in a cell for both schemes. Note that in **NAG**, the bandwidth reservation is not explicitly defined, so the reserved bandwidths cannot be compared. This utilized bandwidth shows a similar comparison to that observed from Figure 4.24 between the two schemes, i.e., **AC3** is better for the lightly-loaded region, and worse for the heavily-loaded region. By examining the utilized bandwidth, **AC3** might appear worse than **NAG** in the highly-loaded region of the high mobility case, but actually it is not, because P_{CB} is an important performance measure, and P_{CB}'s are almost

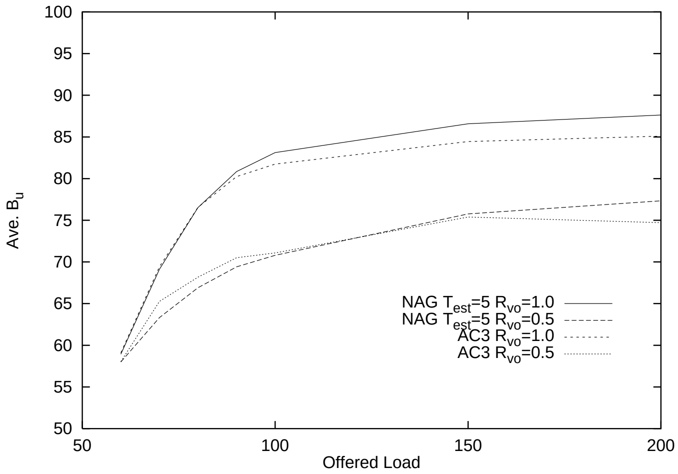


(a) high mobility

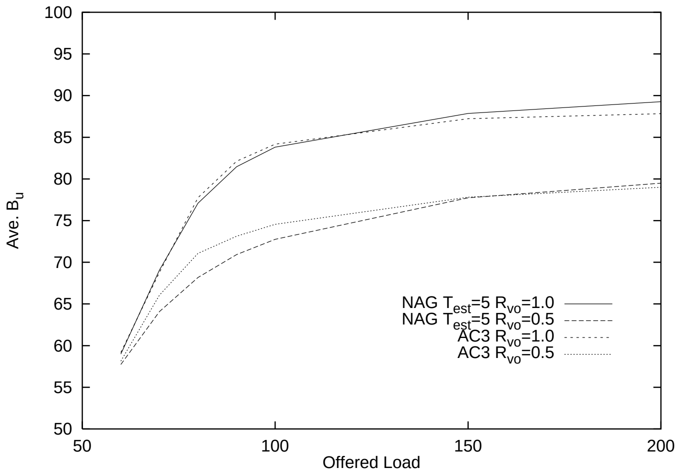


(b) low mobility

Figure 4.24: Comparison of **NAG** and **AC3** using P_{CB} and P_{HD} vs. offered load.



(a) high mobility



(b) low mobility

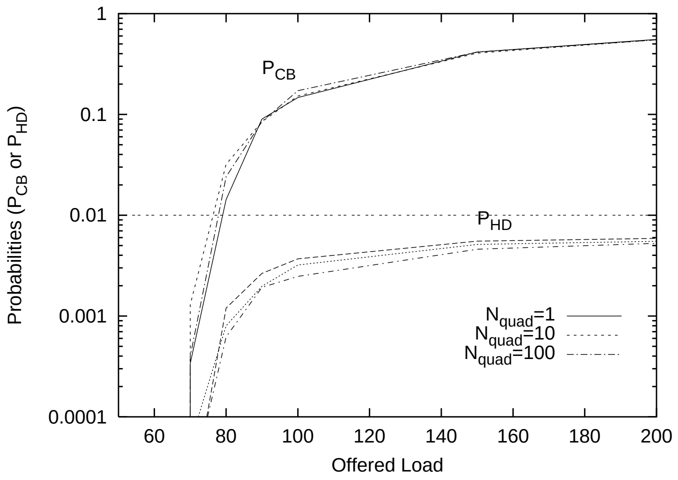
Figure 4.25: Comparison of **NAG** and **AC3** using the average utilized bandwidth B_u vs. offered load.

the same for both schemes for the highly-loaded region. Note that usually the higher average utilized bandwidth, the lower P_{CB} in a system, but it is not always true for different systems or even in a system with different traffic conditions.

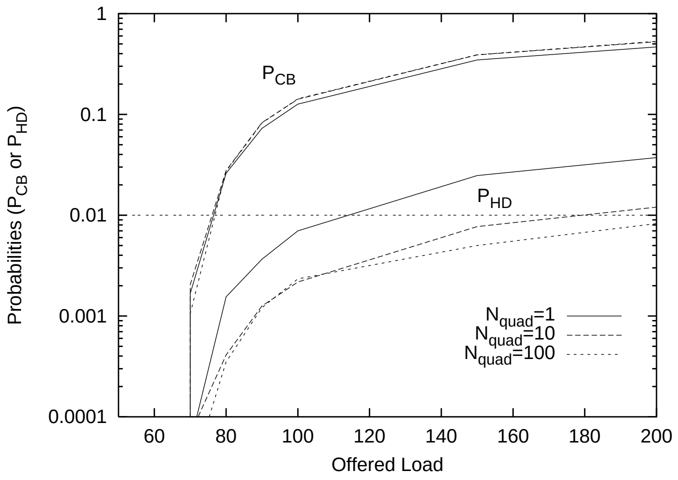
Next, we compare the complexity of the two schemes. We first examine their dependency on the accuracy of mobility estimation, which can be represented by the size of the cached history used for mobility estimation for each $prev$, i.e., the size of maximum hand-off estimation function, N_{quad}. Figure 4.26 plots P_{CB} and P_{HD} as the offered load increases for (a) **AC3** and (b) **NAG** with $R_{vo} = 1.0$ and $N_{quad} = 1$, 10, and 100. (Note that we have thus far used $N_{quad} = 100$.) From Figure 4.26 (a), we observe that **AC3** doesn't depend much on N_{quad} as different values of N_{quad} yield almost the same performance. **AC3** achieves the design goal even with $N_{quad} = 1$, implying that it uses only one cached history for mobility estimation. This indicates the robustness of **AC3** to the inaccuracy of mobility estimation thanks to the mobility estimation time window control. On the other hand, Figure 4.26 (b) shows that **NAG** starts to violate the design goal in the over-loaded region with $N_{quad} = 10$. This implies that **NAG** requires very accurate mobility estimation. Note that this difference of dependency on the mobility-estimation accuracy clearly separates the two in terms of memory and computation complexity. The memory required for cached history directly depends on N_{quad}, and the computation complexity of the hand-off probability p_h in Eq. (4.4) is also affected greatly by N_{quad}.

Figure 4.27 shows the average number of calculations of the hand-off probabilities p_h to decide admissibility upon request of a new connection. **NAG** considers both incoming and outgoing hand-offs by calculating p_h's and p_s's. Calculation of p_s requires as many calculations of p_h's as the number of adjacent cells. In addition, **NAG** requires the admissibility decision in both the current and all adjacent cells while **AC3** determines it adaptively according to the condition of adjacent cells. We observe that **NAG** requires at least 4 times as many p_h calculations as **AC3** does, where the lower the offered load, the more pronounced difference in the number of calculations between them.

Finally, we combine the dependency on N_{quad} and the number of p_h calculations. Figure 4.28 shows the average numbers of numerical operations (e.g., summations and multiplications) and comparisons used to make an admission decision. Comparisons include the decisions such as if t_{soj} is larger than a value in the summations of Eq. (4.4). For **NAG**, $N_{quad} = 10$ is used even though the design goal is not always met with this value while $N_{quad} = 1$ is used for **AC3**. The complexity to keep up with the average lifetime of each mobile's connections needed for **NAG** was not included. Moreover, the computation of the



(a) **AC3**



(b) **NAG**

Figure 4.26: Comparison of **NAG** and **AC3** using the dependency on the value of N_{quad} for $R_{vo} = 1.0$.

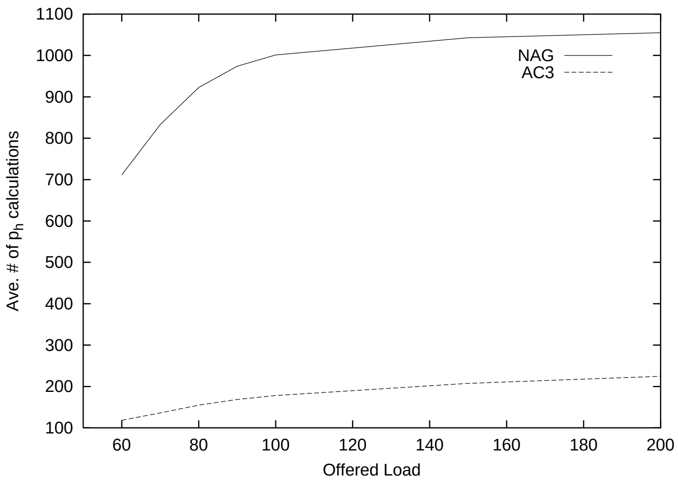


Figure 4.27: Complexity comparison of **NAG** and **AC3** using the number of hand-off probability p_h calculations for an admission test for $R_{vo} = 1.0$.

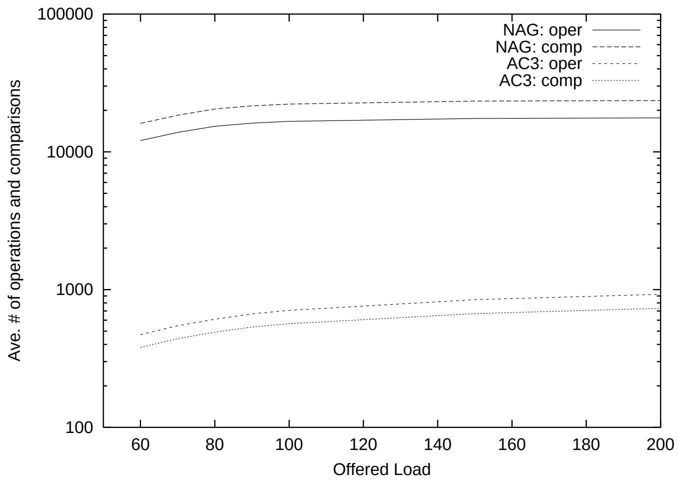


Figure 4.28: Complexity comparison of **NAG** (with $N_{quad} = 10$) and **AC3** (with $N_{quad} = 1$) using the average numbers of numerical operations and comparisons for an admission decision for $R_{vo} = 1.0$.

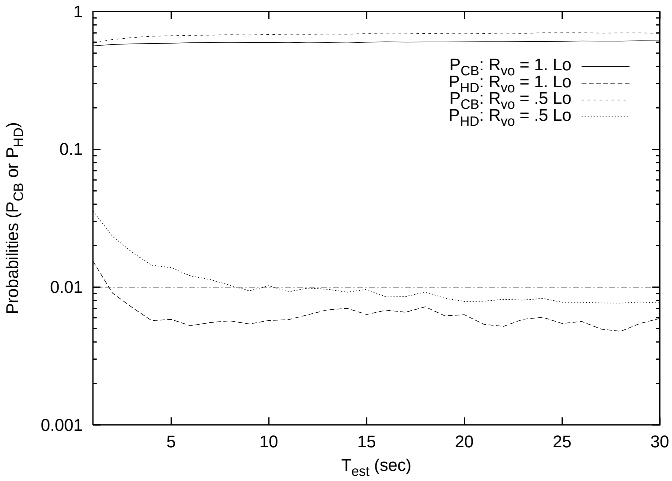


Figure 4.29: P_{CB} and P_{HD} vs. estimation time T_{est} in the two-dimensional environment with $C = 50$ and $L = 100$: **NAG**.

function $Q(\cdot)$ in Eq. (4.15) was also counted as one operation. Note that these are not fair to **AC3**. For **AC3**, the numbers of operations and comparisons used for the mobility time window control algorithm, normalized by the number of connection arrivals, are also added in the plots. From the graph, the complexity of **NAG** is found to be about 17.4 to 25.7 times of that of **AC3** in terms of the number of numerical operations, and about 29.6 to 42.3 times in terms of the number of comparisons. The lower the offered load, the larger the difference between them. So, we can conclude that **NAG** is much more expensive than **AC3** to attain a similar performance.

4.7.10 Comparison of AC3 and NAG: Two-Dimensional Case

In this subsection, we compare **AC3** and **NAG** in the two-dimensional environment following a similar step in the previous subsection for the one-dimensional structure. Note that the link capacity $C = 50$, and we consider the low mobility case only since high user mobility is not likely in a downtown area, which is modeled as a two-dimensional structure. First, Figure 4.29 shows P_{CB} and P_{HD} of **NAG** for different values of T_{est}. We observe that P_{HD}'s for both $R_{vo} = 1.0$ and 0.5 start to fall below the target value, 0.01, at around $T_{est} = 20$ (sec). On the other hand, we also observe that P_{CB}'s increase very slowly as T_{est} increases, i.e., the smaller T_{est}, the better in terms of P_{CB}. So, we select $T_{est} = 20$ (sec) for the two-dimensional structure for further experiments.

Figure 4.30 plots the two probabilities P_{CB} and P_{HD} for **NAG** with $T_{est} = 20$ (sec) and **AC3**. Basically, the general tendency is the same as that from the one-dimensional

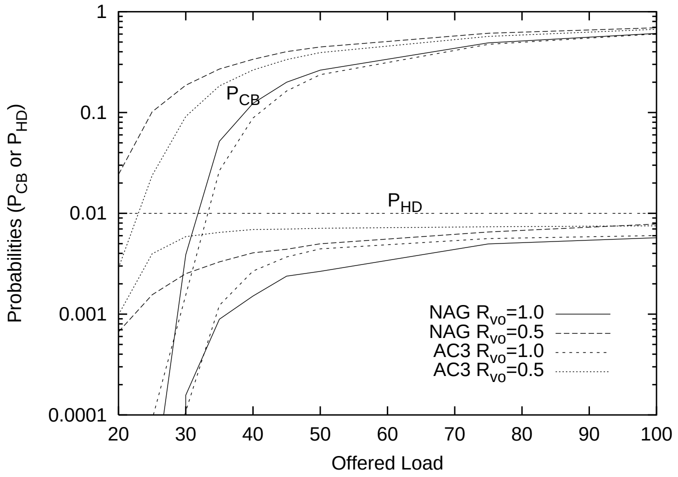
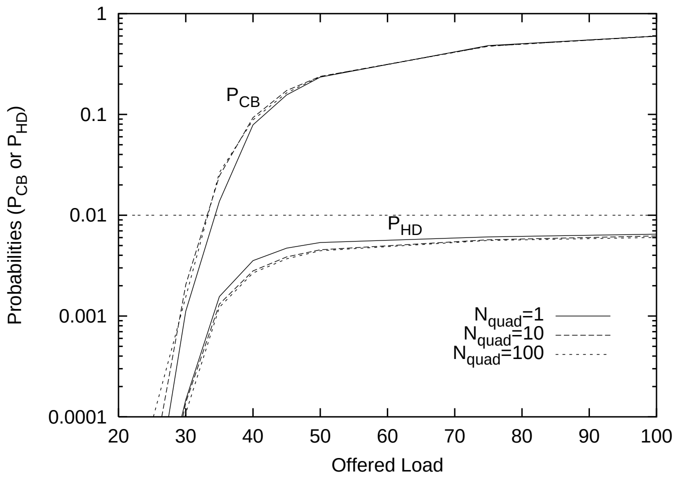


Figure 4.30: Comparison of **NAG** and **AC3** using P_{CB} and P_{HD} vs. offered load for the two-dimensional case.

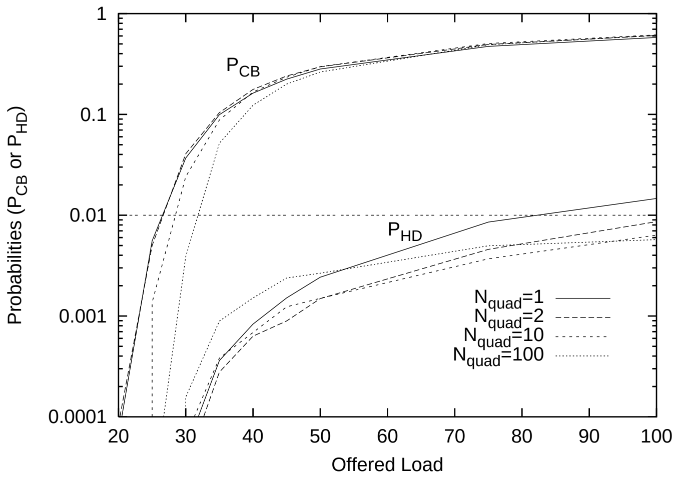
structure. One difference we can observe is that P_{CB} of **NAG** is distinctly larger than that of **AC3**. Note that P_{CB}'s were almost the same for both schemes in the one-dimensional structure. This difference seems more important in this case since P_{CB}'s are relatively large even in the lightly-loaded region. For example, the P_{CB}'s of both schemes are slightly larger than 0.1 (i.e., 10 %) at the offered load $L = 40$ (30) and $R_{vo} = 1.0$ (0.5).

Now, we examine the sensitivity of the two schemes to the accuracy of mobility estimation. From Figure 4.31, we observe that **AC3** is still very robust to the inaccuracy of mobility estimation while **NAG** is also quite robust in terms of the design goal achievement. **NAG** attains the design goal even with $N_{quad} = 2$. However, P_{CB} of **NAG** seems to be affected by the accuracy of mobility estimation in this case since we observe that the smaller N_{quad}, the larger P_{CB}. Note that P_{CB} of **NAG** was almost independent of N_{quad} in the one-dimensional case as observed from Figure 4.26 (b). With **NAG**, either (or possibly both) of P_{CB} and P_{HD} appears to be affected by the value of N_{quad} depending on the cellular structure, user mobility pattern, and others. In any case, the performance of **NAG** is heavily dependent on the mobility estimation accuracy.

Finally, we compare the computation complexity of **AC3** and **NAG**. Figure 4.32 shows the average number of numerical operations and comparisons used to make an admission decision for **AC3** with $N_{quad} = 1$ and **NAG** with $N_{quad} = 2$ for $R_{vo} = 1.0$. We observed a similar tendency to that from the one-dimensional structure. From Figure 4.32, one can find the complexity of **NAG** to be about 14.1 to 22.3 times of that of **AC3** in terms of the number of numerical operations, and about 10.3 to 15.8 times in terms of the number



(a) **AC3**



(b) **NAG**

Figure 4.31: Comparison of **NAG** and **AC3** using the dependency on the value of N_{quad} for $R_{vo} = 1.0$ and the two-dimensional case.

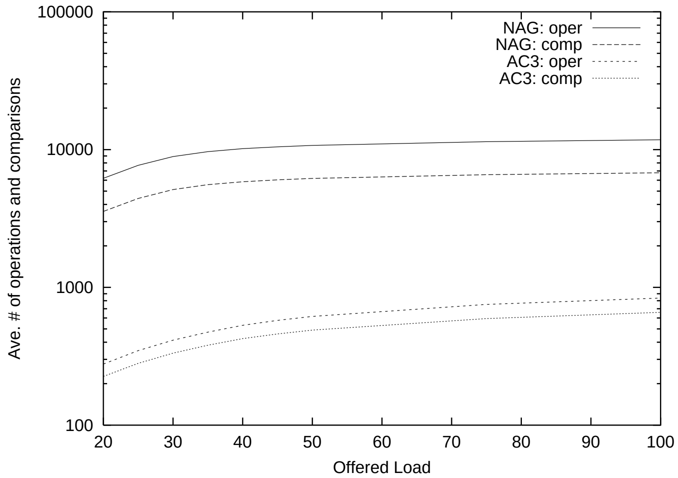


Figure 4.32: Complexity comparison of **NAG** (with $N_{quad} = 2$) and **AC3** (with $N_{quad} = 1$) using the average numbers of numerical operations and comparisons for an admission decision for $R_{vo} = 1.0$ and the two-dimensional case.

	P_{HD}	P_{CB}	Complexity	T_{est}
AC3	bounded	about the same	1	adapted
NAG	bounded with T_{est}	about the same	at least 10 times	shoud be assigned
AG	guaranteed zero	worst	N/A	N/A
BHARG	virtually zero	second worst	N/A	N/A
NCBF	virtually zero	third worst	N/A	N/A

Table 4.4: Summary of comparison among **AC3**, **NAG**, **AG**, **BHARG**, and **NCBF**.

of comparisons. The complexity difference between **AC3** and **NAG** seems to be smaller here since **NAG** with $N_{quad} = 2$ (instead of $N_{quad} = 10$) was compared. The lower the offered load, the larger the difference between them. From the two-dimensional results, we re-confirm that **NAG** is much more expensive than **AC3** to attain similar performance.

Table 4.4 summarizes the comparison results of the five different schemes considered thus far. Note that **AG** does not use the history-based mobility estimation, but relies on mobility specification, which is practically difficult to obtain. In fact, we didn't account for how to predict the next cell of a mobile for **BHARG** and **NCBF**. So, their complexity can't be compared fairly with the other two schemes.

4.8 Conclusion

In this chapter, we designed and evaluated predictive, adaptive bandwidth reservation for hand-offs and admission control so as to keep the hand-off dropping probability below a pre-specified value. Our schemes utilize the following two components to reserve bandwidth for hand-offs: (1) *hand-off estimation functions* which are used to predict a mobile's next cell and estimate its sojourn time probabilistically based on its previously-resided cell and the observed history of hand-offs in each cell; and (2) *mobility estimation time window control scheme* in which, depending on the observed hand-off drops, the estimation time window size is controlled adaptively for efficient use of bandwidth and effective response to (i) time-varying traffic/mobility and (ii) inaccuracy of mobility estimation.

We considered three different admission-control schemes depending on how many neighboring BSs participate in the admission decision of a new connection request. Through the performance and complexity comparisons, we concluded a hybrid one called **AC3** is superior to the others. Our best scheme is not optimal in the sense that there might be a better scheme resulting in a lower connection blocking probability while keeping the hand-off dropping probability below the target value. However, this scheme is not complex nor based on any impractical assumptions, and hence it is readily implementable. It is also shown to be robust and work well under a variety of traffic loads, connection bandwidths, and mobility.

We also compared our best scheme with five other connection bandwidth reservation schemes, called **STATIC**, **NAG**, **AG**, **BHARG**, and **NCBF**. **NAG** is made to utilize our mobility-estimation scheme since this mobility estimation is practically feasible. **NAG** was also generalized to accommodate heterogeneous connections. We showed how costly it is to make the hand-off dropping probability zero even under an impractical assumption by evaluating the performance of **AG**. The other two per-connection bandwidth reservation schemes, **BHARG** and **NCBF**, achieved a lower P_{CB}, but were found to be very expensive (even though they are less expensive than **AG**). So, one can conclude that per-connection bandwidth reservation is too expensive to be practical.

NAG was shown to require much more memory and computation than our best scheme **AC3** in order to meet the design goal. **NAG** is also observed to depend greatly on the design parameter T_{est}, which is difficult to adjust in real world. By contrast, **AC3** is robust to the inaccuracy of mobility estimation thanks to the mobility estimation window control while meeting the design goal over the entire range of the offered loads considered even with much less memory and computation. **AC3** is therefore preferable to, and practically more attractive than, **NAG**.

We plan to evaluate our scheme in more realistic and general environments. This will include more realistic moving patterns of users (e.g., combined vehicular, pedestrian, and stationary mobiles) and their effects. The modification of the proposed scheme to be used in the CDMA systems is also planned, where hand-off drops can be reduced due to (1) soft capacity notion and (2) soft hand-off support. We will also integrate our work with routing and re-routing in the wired networks by considering bandwidth reservation in the wired links along the routes of hand-off connections.

CHAPTER 5

BANDWIDTH MANAGEMENT FOR ADAPTIVE QOS

5.1 Introduction

The key to support wireless/mobile QoS is how to manage scarce wireless link bandwidth efficiently in a highly dynamic environment with time-varying channel conditions and user mobility. In view of rapidly-fluctuating wireless link bandwidths, we can use an *adaptive QoS* concept in which a connection's required bandwidth is not fixed at a single value, but is given as a *range* of bandwidth, $[b_{min}, b_{max}]$. Ideally, it is desirable to always provide the maximum required bandwidth b_{max} for each connection. However, as a user moves around, or due to dynamically-fluctuating channel and network conditions, this will not always be possible. For example, suppose a connection was provided the maximum bandwidth b_{max} by the BS in a cell. When the user with that connection moves into an adjacent cell, the new cell might not have enough bandwidth available to maintain the maximum required bandwidth b_{max} for the connection. If the cell has only bandwidth $b \in [b_{min}, b_{max})$ available, the connection need not be dropped, and instead, the bandwidth for that connection can be lowered to b. The minimum required bandwidth b_{min} can be considered as the bandwidth required to support the lowest-level quality of the connection the mobile user can "live with." Hence, we can use the minimum required bandwidth b_{min} for admitting/rejecting each new/handed-off connection.

Using this adaptive QoS concept, it is possible to utilize bandwidth more efficiently, thus increasing the service provider's reward/revenue, while reducing the number of hand-off drops and new connection blocks. In this chapter, we present and evaluate an adaptive QoS framework and bandwidth-adaptation mechanisms which maximize the service provider's reward/revenue. Distinct features of our scheme include: (1) penalizing connections which initiate adaptation actions too frequently; and (2) incorporating application-specific adaptation constraints, such as how often and how much adaptation can be made. Basically, we present in this chapter how the bandwidth reservation for hand-offs and admission control for new connections in the previous chapter can be modified to support adaptive QoS as well as the adaptation mechanisms and rules.

This chapter is organized as follows: Section 5.2 describes the wireless system model under consideration. Our adaptive bandwidth-allocation mechanism and adaptation rules are presented in Section 5.3. Using simulations, the performance of our scheme is evaluated, and compared with that of a non-adaptive scheme in Section 5.4. The chapter concludes with Section 5.5.

5.2 System Description

This section briefly describes the specific system considered in this chapter, including adaptive error control, adaptive applications, and reward/revenue models. As in the previous chapter, the cellular system uses a fixed channel allocation (FCA) scheme, and cell i has a wireless link capacity $C(i)$. The unit of link bandwidth is BU. Let $C_{i,j}$ be connection j in cell i, and $\mathbf{S}_i$ be the set of indices of the connections in cell i.

5.2.1 Adaptive Error Control

Our adaptive bandwidth management scheme assumes adaptive error control to handle time-varying channel conditions. There are a number of well-known adaptive error-control schemes, including:

- **Adaptive modulation:** with the same transmission power and bandwidth, M-ary modulation with a larger M can achieve a higher data transmission rate at the expense of poorer error performance. Depending on the underlying channel condition, the modulation can be adapted (e.g., adaptation between BPSK and QPSK in WaveLAN [68]).

- **Adaptive usage of error-control codes:** by encoding the information (or packet) using an error-control code, redundant bits are appended. Depending on the observed channel condition, error-control codes can be used adaptively, i.e., the worse the channel condition gets, the more redundancy appended to the original packet [50, 33, 32, 20].

One may use a combination of the above techniques. A common characteristic of these techniques is that the better the channel condition, the more efficiently the channel resources can be used, i.e., a higher transmission rate can be achieved with the same bandwidth. We represent the *bandwidth usage efficiency* of a connection with r_c (≤ 1), i.e.,

$$b_g = r_c w_a, \tag{5.1}$$

where w_a is the bandwidth allocated to this connection and b_g is the actual throughput (or rate) perceived by (or granted to) this connection. Note that r_c is a time-varying function of the location of the connection and its environment, and will be determined by the underlying error-control scheme.[1]

For the given link capacity $C(i)$ of cell i, the sum $W_{a,i}$ of bandwidths allocated to all the connections in the cell should be bounded by $C(i)$:

$$W_{a,i} = \sum_{j \in \mathbf{S}_i} w_a(C_{i,j}) \leq C(i), \tag{5.2}$$

where $w_a(C_{i,j})$ is the bandwidth allocated to connection $C_{i,j}$. Note that the sum, $B_{g,i} = \sum_{j \in \mathbf{S}_i} b_g(C_{i,j})$, of user-perceived actual throughputs, where $b_g(C_{i,j})$ is the actual throughput perceived by connection $C_{i,j}$, and so its upper-bound, which is actually the user-perceived link capacity, is time-varying and dependent on the bandwidth allocated to each connection, i.e., $w_a(C_{i,j})$. We assume that r_c can have a number of different discrete values, i.e., $r_c \in \{r_1\ (= r_{min}), r_2, \cdots, r_n\ (= r_{max})\}$ since only limited sets of modulations and error-control codes are used due mainly to the resultant computational complexity proportional to the size of the sets. The underlying adaptive error-control scheme should be good enough to handle long-term variations of the channel, e.g., resulting from shadowing and signal attenuation, while it may not handle short-term variations, e.g., resulting from short-term fading. Short-term variations can then be handled by proper MAC and packet scheduling, e.g., our solution for WLAN in Chapter 2. These combined strategies render a specific (average) amount b_g of actual throughput perceived by each connection at the cost of bandwidth $(1 - r_c)w_a$. Note that the bandwidth usage efficiency r_c should reflect the time-averaged — not instantaneous — behavior of the channel a connection is experiencing.

5.2.2 Adaptive Application and Reward/Revenue Models

We assume that each connection is specified by a *range*, $[b_{min},\ b_{max}]$, of the required actual throughput. For example, real-time multimedia traffic can be represented and trans-

[1]Conceptually, r_c can be interpreted as the channel code rate of the code being used if an adaptive error-control strategy is used.

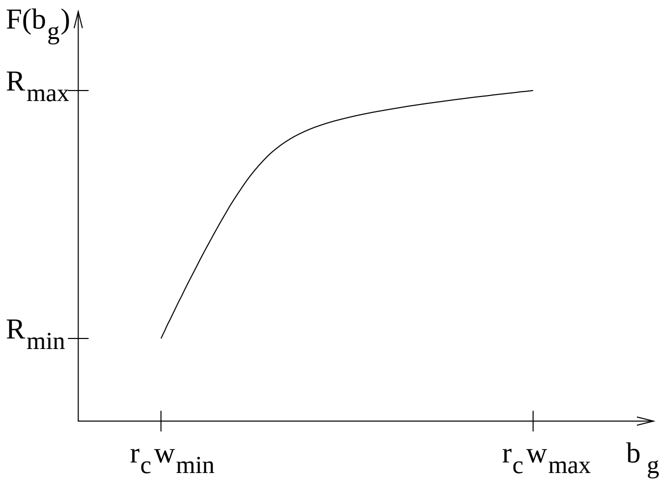


Figure 5.1: The reward function $F(b_g)$, where $b_g = r_c w_a$, and $w_{max\ (min)} = b_{max\ (min)}/r_c$.

ported as multi-layer scalable flows [72]. For coded video and audio, for instance, it is possible to change the output rate by adjusting some encoding parameters. Moreover, when a multi-level or hierarchical coding scheme is used, the encoded data can be selectively skipped without significantly degrading the perceived quality of the decoded information. A non-real-time connection might require the minimum throughput it can "live with." On the other hand, some non-real-time connections may not require any guaranteed throughput at all as we assumed in Chapters 2 and 3. All the traffic of this type can be handled in an aggregate manner without per-connection bandwidth allocation. More specifically, the required actual throughput of a connection is given by a set of discrete throughput values, $\mathbf{B} = \{b_1(= b_{min}), b_2, \cdots, b_m(= b_{max})\}$, depending on the underlying application. Note that the required bandwidth range, i.e., $[w_{min}, w_{max}]$, where $w_{max\ (min)} = b_{max\ (min)}/r_c$, of a connection with the required actual throughput range $[b_{min}, b_{max}]$ is time-varying depending on the value of r_c.

A reward function is chosen such that the service provider earns $F(b_g)$ units of reward/profit by providing actual throughput $b_g \in \mathbf{B}$ to a connection or an application. The reward function F is connection-specific, and can be an arbitrary non-decreasing function of b_g, depending on the adopted pricing scheme. Figure 5.1 shows an example reward function curve. Note that if a reward function is seen in a service provider's centric view (i.e., given as a function of the allocated bandwidth w_a $(= b_g/r_c)$), then it is a time-varying function (since it depends on the time-varying parameter r_c).

Depending on its application, a connection specifies (1) its desired minimum duration Δt between two consecutive adaptations, called the *desired minimum inter-adaptation in-*

terval, and (2) the desired maximum change Δb on its granted throughput in one adaptation process, called the *desired maximum throughput adaptation*. These are referred to as connection-specific *adaptation constraints* throughout this chapter. For example, a real-time video connection might require Δt to be several seconds and Δb to be a moderate positive number since too frequent and drastic fluctuation of the perceived video quality is not desirable, while a non-real-time connection probably specifies them to be zero since the more bandwidth assigned, the better in this case. Note that these two values specify how often and how much a connection's perceived actual throughput can be adapted. In summary, connection $C_{i,j}$ is specified by a set: $\{\mathbf{B}_{i,j}, F_{i,j}(\cdot), \Delta t_{i,j}, \Delta b_{i,j}\}$. An interesting problem is then how to adapt the allocated bandwidth in order to maximize the total reward with these adaptation constraints when each connection's reward function is a time-varying function of its allocated bandwidth.

5.3 Adaptive QoS via Bandwidth Management

This section describes our framework for bandwidth management to support adaptive QoS in cellular networks so as to maximize the service provider's total reward subject to connection-specific adaptation constraints.

5.3.1 Bandwidth-Adaptation Mechanism

In integrated wireless/wired networks, the wireless part of an end-to-end connection is usually the bottleneck, thus needing to adapt its bandwidth allocation according to dynamically-changing traffic loads due to user mobility and time-varying wireless channel conditions. In case of a cellular network, the BS of a cell has complete authority to determine its link condition and control the link bandwidth allocated to each connection (and the bandwidth reserved for hand-offs. In this dissertation, we concentrate on the adaptation actions initiated or ordered by BSs due to the wireless link load fluctuations.

There are two types of adaptation actions: bandwidth *upgrade* and *downgrade*. Bandwidth downgrade actions are crucial since they directly affect the rejection of new connection requests and the dropping of handed-off/on-going connections. There are two cases that invoke a downgrade action:

- **Connection arrival:** upon arrival of a new or handed-off connection at a cell, the BS needs to check if the cell has enough bandwidth to accommodate this new arrival while providing at least the minimum required bandwidth to each existing connection. If there happens to be a bandwidth demand conflict, i.e., the sum of allocated bandwidth is greater than the link capacity, the BS will attempt to downgrade connections' bandwidths (including the new arrival's) without compromising their required minimum to accommodate the new one.

- **Change of channel condition:** a downgrade action can also be triggered when a connection's channel condition gets worse, and the error-control mechanism requires more bandwidth (for redundancy) to maintain this connection's QoS. If there is not enough bandwidth available to allocate to this connection, a downgrade action should be invoked.

An upgrade action, on the other hand, can be triggered when additional bandwidth becomes available as a result of (1) a connection termination, (2) outgoing hand-offs, or (3) an improved channel condition that decreases the need for a connection's error-control redundancy. However, unlike event-driven downgrade actions, upgrade actions should be performed more judiciously utilizing each connection's desired minimum inter-adaptation interval. This way we can reduce a wireless link bandwidth oscillation with frequent upgrade and downgrade actions, thus causing significant overhead and user-perceived quality to fluctuate severely, both of which are undesirable. Note that adaptation actions are not free; they too will consume resources.

The actual adaptation mechanism will differ depending on the downlink or uplink connections. Uplink adaptation is relatively simple since the BS issues adaptation commands to each mobile with an application connection, and then the mobile will perform the commanded adaptation. However, downlink adaptation is costlier since the wireless downlink is the end-most link for an end-to-end connection. Downlink adaptation can be in one of the following two forms:

- **End-host adaptation**: performed by the source node. The BS needs to send an adaptation request/command to the source node. This will introduce some time delay and will be costlier due to the need of a feedback action.

- **Intermediate node's filtering**: there could be an intermediate node (e.g., a proxy server [102]), or the BS itself to filter the received packets for an application connection. Unfortunately, the wired network bandwidth consumed by those packets filtered out by an intermediate node will be wasted. Note that this adaptation mechanism is relevant to a downgrade action only.

The network uses these two downlink adaptation options in order to minimize the adaptation cost depending on the underlying wired network configurations. Note that a connection's upgrade process should be preceded by a probing process to check if all the intermediate links along the route have enough bandwidth for the upgrade.

5.3.2 Connection-Admission Control

As mentioned above, a downgrade action can be triggered by the admission of a new or handed-off connection. Described below is an admission-control process for both new and handed-off connections. Let $b_{min}(C_{i,j})$ and $b_{max}(C_{i,j})$ be $C_{i,j}$'s minimum and maximum required actual throughputs, respectively.

The admission-control process checks if it is always possible to provide each connection its minimum required actual throughput. The admission test for a handed-off connection C_{ho} in cell i can be represented by

$$\sum_{i} b_{min}(C_{i,j})/\hat{r}_c(C_{i,j}) + b_{min}(C_{ho})/r_{min} \leq C(i), \tag{5.3}$$

where the *worst-case* bandwidth usage efficiency $\hat{r}_c$ is given by

$$\hat{r}_c(C_{i,j}) = \begin{cases} r_c(C_{i,j}), & \text{if } C_{i,j} \text{ is stationary,} \\ r_{min}, & \text{if } C_{i,j} \text{ is moving.} \end{cases} \tag{5.4}$$

That is, the $\hat{r}_c$ value of a stationary connection is determined at its current bandwidth usage efficiency while that of a moving connection is determined at the minimum bandwidth usage efficiency r_{min}. The rationale behind using $\hat{r}_c(C_{i,j})$ instead of $r_c(C_{i,j})$ in Eq. (5.3) is that $r_c(C_{i,j})$ of a moving connection is supposed to vary over time, and the connection might be dropped in a cell when its channel condition gets worse. For the same reason, r_{min} is used for a handed-off connection. A newly-admitted connection is assumed to be moving, then if its bandwidth usage efficiency does not vary and it does not hand off during a certain time period (e.g., the maximum sojourn time $T_{soj,max}$ in the cell where the connection is currently residing as defined in Section 4.4.3), it is assumed stationary. The channel a stationary connection experiences could vary over time due to its time-varying environment. However, the "average" channel behavior will not change much, so r_c, which is determined based on the time-averaged behavior of a channel, will stay virtually constant.

Admission control of new connections is based on the scheme proposed in the previous chapter, in which the adaptive QoS concept was not used. A portion of the link capacity is set aside in each cell for possible hand-offs from its adjacent cells to keep the hand-off dropping probability below a target value. This reserved bandwidth can be used only for hand-offs from adjacent cells, but *not* for newly-requested connections in the cell. A newly-requested connection C_{new} in cell i also requires an admission test:

$$\sum_i b_{min}(C_{i,j})/\hat{r}_c(C_{i,j}) + b_{min}(C_{new})/r_{min} \leq C(i) - W_{r,i}, \tag{5.5}$$

where $W_{r,i}$ is the *target reservation bandwidth* — the required bandwidth to be reserved for hand-offs — in cell i. Upon arrival of a new connection request, $W_{r,i}$ is updated predictively and adaptively — before performing the admission test Eq. (5.5) on the request — depending on the neighbors' loads. For the admission decision, in addition to checking bandwidth availability in cell i as in Eq. (5.5), some neighbors of cell i also check their bandwidth availability as described in Section 4.4. Even though r_{min} is used for the admission test for each new connection, its actual r_c value can be determined from the underlying error-control scheme. For example, one can use r_{max} (i.e., $r_c = r_{max}$) at the beginning of connection setup, then decrease r_{max} gradually according to the observed channel condition.

The maximum actual throughput $b_{max}(C_{new})$ is provided initially to the newly-admitted connection. Then, connection-specific parameters are defined as:

$$\Delta t^+_{new} := \Delta t^-_{new} := \Delta t_{new}, \tag{5.6}$$

$$\Delta b^+_{new} := \Delta b^-_{new} := \Delta b_{new}, \tag{5.7}$$

where Δt^+ (Δt^-) is the *target* minimum time for an upgrade (downgrade) action measured since the last adaptation, and Δb^+ (Δb^-) is the *target* upper-bound on the actual throughput increase (decrease) in each upgrade (downgrade) action. If there happens to be a bandwidth demand conflict after admitting a new or handed-off connection, a downgrade action is invoked to resolve it. Bandwidth adaptation rules are detailed in the next subsection.

5.3.3 Adaptation Rule

Basic questions regarding bandwidth adaptation are (1) when to adapt, (2) who to adapt, and (3) how much to adapt. Our bandwidth adaptation will attempt to maximize the aggregate reward from already-admitted connections while satisfying connection-specific adaptation constraints. To better explain this adaptation rule, we define two parameters for each connection, i.e., the upgrade slope $\Delta R^+_{i,j}$ and the downgrade slope $\Delta R^-_{i,j}$ of connection $C_{i,j}$'s reward function $F_{i,j}$ are, respectively, defined by

$$\Delta R^+_{i,j} = \begin{cases} 0, & \text{if } l \text{ is the maximum,} \\ \frac{F_{i,j}(r_c(C_{i,j})w_{l+1}) - F_{i,j}(r_c(C_{i,j})w_l)}{w_{l+1} - w_l}, & \text{otherwise,} \end{cases} \tag{5.8}$$

$\Delta t_{adj} := 0; \qquad \Delta b_{adj} := 0;$

while $(W_{a,i} > C(i))$ {

    if $(\Delta t_{adj} = \max_j \Delta t^-_{i,j})$ $\Delta b_{adj} := \Delta b_{adj} + 1$ (BUs);

    while ($\nexists$ k s.t. $\Delta R^-_{i,k} \neq \infty$, $\tau_{i,k} > \Delta t^-_{i,k} - \Delta t_{adj}$, and $\beta_{i,k} < \Delta b^-_{i,k} + \Delta b_{adj}$)

        $\Delta t_{adj} := \Delta t_{adj} + 1$ (sec);

    choose j with min $\Delta R^-_{i,j}$, $\tau_{i,j} > \Delta t^-_{i,j} - \Delta t_{adj}$, and $\beta_{i,j} < \Delta b^-_{i,j} + \Delta b_{adj}$;

    $W_{a,i} := W_{a,i} - (w_l - w_{l-1})$, where $l = level(C_{i,j})$;

    $lev(C_{i,j}) := lev(C_{i,j}) - 1$;

    if $(\Delta t_{adj} \neq 0)$ $\Delta t^-_{i,j} := \Delta t^-_{i,j} + \Delta t_{adj}$;

    else $\Delta t^-_{i,j} := \Delta t_{i,j}$;

    if $(\Delta b_{adj} \neq 0)$

        if $(\Delta b^-_{i,j} > 1$ (BU)) $\Delta b^-_{i,j} := \Delta b^-_{i,j} - 1$ (BU);

    else $\Delta b^-_{i,j} := \Delta b_{i,j}$;

}

Figure 5.2: The pseudo-code of an algorithm to resolve a bandwidth demand conflict among connections via downgrade in cell i.

$$\Delta R^-_{i,j} = \begin{cases} \infty, & \text{if } l = 1, \\ \frac{F_{i,j}(r_c(C_{i,j})w_l) - F_{i,j}(r_c(C_{i,j})w_{l-1})}{w_l - w_{l-1}}, & \text{otherwise,} \end{cases} \tag{5.9}$$

where l is the level of the actual bandwidth received by connection $C_{i,j}$, i.e., $b_g(C_{i,j}) = b_l$ or $l = level(C_{i,j})$.

When a new or handed-off connection is admitted, or a connection's channel condition gets deteriorated, the aggregate bandwidth allocation in the cell might become larger than the link capacity, i.e.,

$$W_{a,i} = \sum_{j \in \mathbf{S}_i} w_a(C_{i,j}) > C(i). \tag{5.10}$$

Then, the allocated bandwidths of some connections (possibly including the newly-admitted or handed-off connection) should be reduced. The downgrade process in cell i works as shown in Fig. 5.2, where $\tau_{i,j}$ is the time elapsed since $C_{i,j}$'s last bandwidth adaptation, and $\beta_{i,j}$ is its actual throughput decrease during the current downgrade. If $C_{i,j}$ never experienced an adaptation before, $\tau_{i,j}$ is set to the time elapsed since its setup. As clear from the pseudo-code, while meeting its adaptation constraints, a connection with the minimum downgrade slope is chosen and downgraded until the aggregate bandwidth allocation becomes less

if ($C_{i,j}$ initiated a downgrade) {

$\Delta t^+_{i,j} := \Delta t^+_{i,j} + 1$ (sec);

if ($\Delta b^+_{i,j} > 1$) $\Delta b^+_{i,j} := \Delta b^+_{i,j} - 1$ (BU);

}

Figure 5.3: The pseudo-code of a process to penalize an on-going connection $C_{i,j}$ which initiated a downgrade action in cell i.

than, or equal to, the link capacity. If it is not possible to find a connection satisfying the adaptation constraints, the constraints are "loosened" until such a connection is found, and then this connection's constraints are tightened, i.e., Δt^- (Δb^-) is increased (decreased), to compensate for the violation of the constraints later. Note that keeping all the on-going connections (or accommodating a new connection) is more important than not violating the connections' adaptation constraints. This is why Δt and Δb is said to be *desired* (as opposed to "required") values in Section 5.2. If a newly-admitted connection initiates a downgrade action, then its Δb^- (Δt^-) is set to ∞ ($-\infty$) to render it unlimited adaptation from its maximum actual throughput provided initially.

As shown in Fig. 5.3, when on-going connection $C_{i,j}$ initiated a downgrade action due to its hand-off or perceived channel deterioration, it is "penalized" to have less chance to upgrade/upgrade in future. This will reduce the fluctuation of wireless link bandwidth. An upgrade action is triggered when the total allocated bandwidth is less than the unreserved bandwidth, i.e.,

$$W_{a,i} < C(i) - W_{r,i}. \tag{5.11}$$

The upgrade process in cell i works as shown in Fig. 5.4. Basically, while satisfying its adaptation constraints, a connection with the maximum upgrade slope is chosen, and upgraded as long as the aggregate allocated bandwidth is less than the unreserved bandwidth. When no adaptation action was invoked for a time duration T_{adapt}, which is the maximum cell inter-adaptation interval, the process in Fig. 5.5 is initiated to re-allocate bandwidths among the existing connections. This selects those connections satisfying adaptation constraints with the maximum upgrade slope and the mimimum downgrade slope, and upgrades and downgrades their allocated bandwidths, respectively, if it results in a positive gain of reward. Even without any link-bandwidth fluctuation, each connection's status in terms of adaptation constraints varies with time, e.g., a connection becomes available for adaptation by passing Δt^+ or Δt^- since the last adaptation, so this adaptation increases the aggregate

$index = 1$;

while ($index = 1$ and $\exists\ k$ s.t. $\Delta R^{+}_{i,k} \neq 0$, $\tau_{i,k} > \Delta t^{+}_{i,k}$, and $\beta_{i,k} < \Delta b^{+}_{i,k}$) {

  choose j with max $\Delta R^{+}_{i,j}$, $\tau_{i,j} > \Delta t^{+}_{i,j}$, and $\beta_{i,j} < \Delta b^{+}_{i,j}$;

  $l = lev(C_{i,j})$;

  if $(W_{a,i} + (w_{l+1} - w_l) < C(i) - W_{r,i})$ {

    $W_{a,i} := W_{a,i} + (w_{l+1} - w_l)$;

    $lev(C_{i,j}) := lev(C_{i,j}) + 1$;

  }

  else $index = 0$;

}

Figure 5.4: The pseudo-code of an algorithm to allocate the residual bandwidth to connections (i.e., upgrade) in cell i.

reward.

5.4 Comparative Performance Evaluation

This section presents and discusses the evaluation results of the proposed scheme. We first describe the assumptions and specifications used for the simulation study.

5.4.1 Simulation Assumptions and Specifications

In our simulation environment, mobiles are traveling along a straight road, i.e., a one-dimensional cellular environment. We make the following assumptions for our simulation study:

A1. The cellular system is composed of 10 cells forming a ring as in Section 4.7.1, for which the diameter of each cell is 1 km. Cells are numbered from 1 to 10, i.e., cell $<i>$ represents the i-th cell.

A2. Connection requests are generated according to a Poisson process with rate λ (connections/second/cell) in each cell. A newly-generated connection can appear anywhere in the cell with an equal probability.

A3. Mobiles can travel in either of two directions with an equal probability with a speed chosen randomly between 40 and 60 (km/hour). Each mobile will run straight through

$index = 1;$

while $(index = 1)$ {

  choose j with max $\Delta R^+_{i,j}$, $\tau_{i,j} > \Delta t^+_{i,j}$, and $\beta_{i,j} < \Delta b^+_{i,j}$;

  choose k with min $\Delta R^-_{i,k}$, $\tau_{i,k} > \Delta t^-_{i,k}$, and $\beta_{i,k} < \Delta b^-_{i,k}$;

  $W'_{a,i} := W_{a,i} + (w_{l_1+1} - w_{l_1}) - (w_{l_2} - w_{l_2-1})$, where $l_1 = lev(C_{i,j})$ and $l_2 = lev(C_{i,k})$;

  if $(\Delta R^+_{i,j} > \Delta R^-_{i,k}$ and $W'_{a,i} \leq C(i) - W_{r,i})$ {

    $W_{a,i} := W'_{a,i}$;

    $lev(C_{i,j}) := lev(C_{i,j}) + 1$;

    $lev(C_{i,k}) := lev(C_{i,k}) - 1$;

  }

  else $index = 0$;

}

Figure 5.5: The pseudo-code of an algorithm to re-allocate bandwidths to connections in cell i.

the road with the chosen speed, i.e., mobiles will never turn around.

A4. Each connection's lifetime is exponentially-distributed with mean 120 (seconds).

A5. A connection's required actual throughput set is given by $\mathbf{B} = \{1\ (= b_{min}), 2, 3, 4\ (= b_{max})\}$ (BUs). The reward function $F(\cdot)$ is given by $F(1) = 1$, $F(2) = 1.5$, $F(3) = 1.8$, and $F(4) = 2$. The desired minimum inter-adaptation interval is assigned by $\Delta t = 5$ (sec). The desired maximum throughput adaptation is assigned by $\Delta b = 2$ (BUs).

A6. A connection's bandwidth usage efficiency is given by $r_c = 0.95$ when its mobile is within 0.25 km range from a BS, and $r_c = 0.8$ otherwise.

A7. Each cell has a fixed link capacity 100 BUs, i.e., $C(i) = C = 100$ for all i.

Assumption **A6** represents the condition that (1) the farther from the BS, the worse the channel condition, which typically happens due to limited transmission power, and (2) two error-control codes are adaptively used to keep the user-perceived error probability below a given threshold. The parameters used include: the target hand-off dropping probability $P_{HD,target} = 0.01$, and the maximum cell inter-adaptation interval $T_{adapt} = 10$ (sec). Other parameters relevant to the bandwidth reservation for hand-offs are assigned the same values used in the previous chapter except for $N_{quad} = 10$.

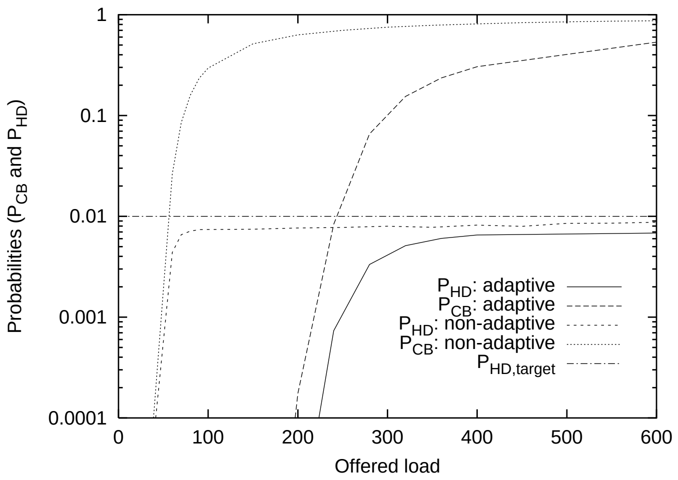


Figure 5.6: P_{CB} and P_{HD} vs. offered load.

A frequently-used measure is the *offered load* per cell, L, which is defined as connection generation rate $\times$ connections' "maximum" required actual throughput $\times$ average connection lifetime, i.e.,

$$L = 4 \cdot \lambda \cdot 120. \tag{5.12}$$

The physical meaning of the offered load per cell is the aggregate throughput required on average to support the maximum required actual throughput to all existing connections in a cell. Note that the definition of the offered load is different from the one in the previous chapter. As a reference for comparison, we consider the performance of a non-adaptive scheme as well, in which each connection's required actual throughput ranges over $[4, 4]$ (BUs), and a fixed channel usage efficiency $r_c = 0.8$ is used irrespective of the connection's location, and the rest is the same as the scheme **AC3** in the previous chapter.

5.4.2 Simulation Results and Discussion

Fig. 5.6 shows the performance of both the proposed adaptive and the non-adaptive schemes in terms of the new connection blocking probability P_{CB} and hand-off dropping probability P_{HD}. First of all, we observe that P_{HD}'s for both schemes are bounded by $P_{HD,target}$ $(= 0.01)$ over the entire range of the offered load thanks to the bandwidth reservation for hand-offs. However, P_{CB} and P_{HD} of the adaptive scheme are much smaller than those of the non-adaptive scheme. Note that these probabilities of the adaptive scheme are smaller than 0.0001 when the offered load is less than 200. This implies that the adaptive scheme can admit much more connections while dropping much fewer hand-offs. This is a

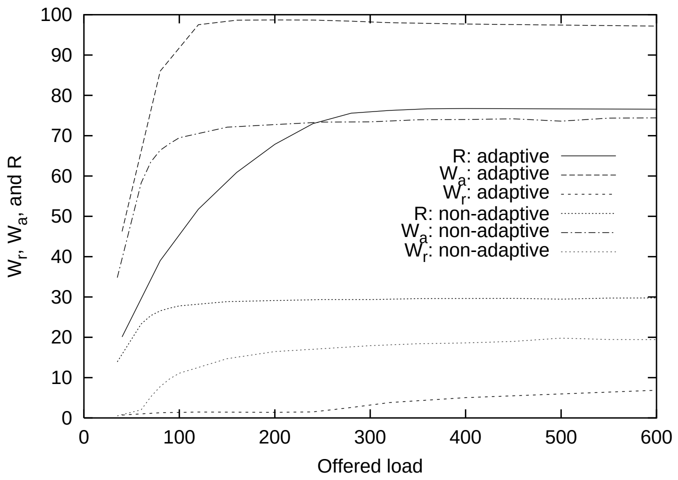


Figure 5.7: W_r, W_a, and R vs. offered load.

key advantage of the adaptive QoS approach.

The utilization of the scarce wireless-link bandwidth and the aggregate revenue are also as important as connection admissibility. Fig. 5.7 shows the average revenue R, the average aggregate allocated bandwidth W_a, and the average bandwidth W_r reserved for hand-offs in each cell. First, we observe that W_a of the adaptive scheme is saturated at almost 100 BUs, which is the link capacity, i.e., the adaptive scheme can fully-utilize the link capacity. On the other hand, W_a of the non-adaptive scheme is saturated at around 75 BUs. The reason for this is because up to 20-BU bandwidth is reserved for hand-offs (i.e., $W_r \approx 20$ in a heavily-loaded region) on average with this scheme, and only the unreserved bandwidth can be utilized. We observe that W_r of the adaptive scheme is almost zero even though it increases slowly as the offered load increases. This is why B_u of the adaptive scheme is almost 100 BUs in the heavily-loaded region. The reason why the average reserved bandwidths of the two schemes differ is that W_r is adaptively determined based on the estimated user mobility and observed hand-off dropping events in each cell so as to keep P_{HD} below the target value, i.e., 0.01 in our study. Note that with the adaptive scheme, a very small bandwidth needs to be reserved to this end.

Another interesting observation is that W_a of the adaptive scheme decreases slowly after peaking at around the offered load of 200. This is because W_r increases after passing the offered load of 200, and the reserved bandwidth cannot be utilized. However, we also observe that the sum of W_a and W_r is larger than the link capacity of 100 BUs especially with the adaptive scheme, e.g., at the offered load of 600, $W_a + W_r \approx 104$. The reason for this is that the reserved bandwidth $W_{r,i}$ in cell i affects only new connection-admission decisions and

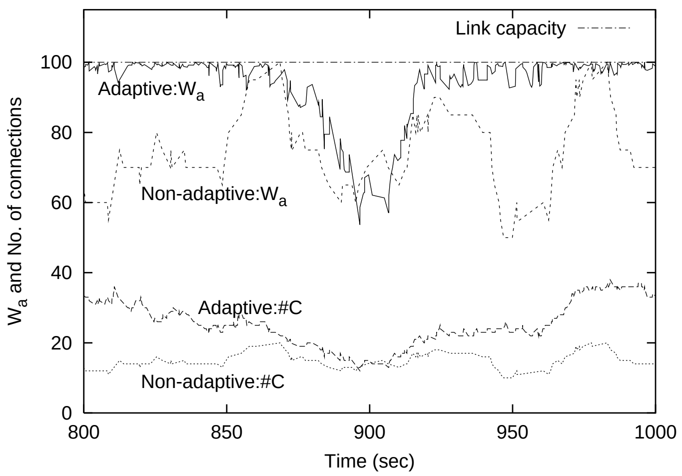


Figure 5.8: W_a and number of on-going connections vs. time when the offered load is 100.

upgrade processes while the whole link capacity can be utilized after a downgrade process. By comparing the aggregate revenue R of the two schemes, we found that R of the adaptive scheme is limited by a larger value, and also increases monotonically till a larger offered load is reached. Since our revenue function has decreasing marginal revenues for an extra unit of actual throughput, the monotonic increase of R as the offered load increases implies that more connections are admitted. One can expect that for the offered load larger than 300, most connections are provided 1 BU which corresponds to one unit of revenue, then the difference of R and W_a comes from the minimum channel usage efficiency r_{min}, which is 0.8 for our simulation study, i.e., $W_a \cdot 0.8 \approx R$.

Fig. 5.8 shows the number (marked by $\#C$) of on-going connections and the aggregate allocated bandwidth W_a in a cell as time goes for a simulation run with the offered load of 100. With the non-adaptive scheme, each connection is allocated 5 BU bandwidth, and hence granted 4 BU ($= 5r_c$) actual throughput. Thus, W_a is just five times the number of on-going connections. That also implies that no more than 20 connections can be accommodated in a cell. On the other hand, with the adaptive scheme, W_a is almost 100 if the number of connections is larger than a threshold, which is determined at between 20 and 23.75 depending on each connection's r_c, i.e., 20 (23.75) when all the connection's $r_c = 0.8$ (0.95). This observation re-confirms that with the adaptive scheme (1) more connections are admitted; and (2) the link bandwidth is efficiently utilized; especially, almost fully-utilized when the number of connections is larger than a threshold.

5.5 Conclusion

In this chapter, we have proposed a unified architecture for wireless bandwidth allocation and management in a highly dynamic environment with user mobility and time-varying channel conditions. Each connection is specified by a set of an acceptable actual throughput range, a revenue function, and adaptation constraints defining how often and how much bandwidth adaptation can be made. The BS allocates bandwidth to each connection so as to maximize the aggregate revenue while attempting to meet the adaptation constraints. An adaptation action can be triggered due to a connection in/out or a channel condition change. Our scheme also penalizes connections which initiate frequent downgrade adaptations in order to reduce link bandwidth fluctuations. Using simulations, we demonstrated how the proposed scheme works, and showed its advantages over a non-adaptive scheme. The proposed scheme is found to be able to admit more connections while utilizing the scarce link bandwidth more effectively.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

6.1 Research Contributions

This dissertation addressed the problem of providing diverse QoS to mobile users in wireless/mobile networks. Main contributions of the dissertation are as follows.

- In Chapter 2, we developed a unified WLAN architecture in order to provide diverse packet-level QoS for heterogeneous traffic. We proposed a polling-based MAC protocol for uplink access with slotted ALOHA-based and piggybacked transmission requests while the D-TDD transmission is used for more efficient and flexible utilization of a frequency channel. A variation of EDF scheduling was used for real-time traffic while a modified round-robin queueing was used for non-real-time traffic in the presence of location-dependent, time-varying, and bursty channel errors. An admission test for each new real-time connection request was also established. An extensive evaluation study has shown the proposed scheme to work well and achieve the design goals.

- An uplink MC-CDMA system was developed to provide diverse packet-level QoS for heterogeneous traffic in wide-area cellular systems in Chapter 3. To satisfy the pre-defined error performance requirements of each traffic class, we used a concatenated RS/convolutional code and developed a new scheme for allocating mobiles power levels. Connection-oriented transmission is achieved by assigning mobile-oriented code channels for real-time traffic while best-effort transmissions are supported for non-real-time traffic. Addressed are the admission test for class-I connections, transmission power allocation, and how to maximize the aggregate throughput for class-II traffic. Through both analyses and simulations, the performance of the proposed scheme was evaluated in detail.

- In Chapter 4, we developed predictive, adaptive bandwidth reservation for hand-offs and admission control for new connections in order to limit the hand-off dropping probability below a pre-specified target. As a basis for our schemes, we developed cell-specific history-based mobility estimation to predict the direction and speed of mobiles. Based on this prediction, the target reservation bandwidth is determined adaptively so as to attain the design goal. Using extensive simulation of various scenarios, we have evaluated the performance of our proposed scheme, and have also shown its superiority to other five existing schemes.

- Finally, we developed a unified architecture for wireless bandwidth allocation and management in a dynamic environment with user mobility and time-varying channel conditions in Chapter 5. The BS allocates bandwidth to each connection so as to maximize the aggregate revenue while attempting to meet the adaptation constraints specified by each connection. We identified the problem of the bandwidth adaptation triggered by time-varying wireless channels, and proposed an adaptive bandwidth-management scheme which handles unreliable and time-varying wireless channel effects in addition to user mobility. Using simulations, we demonstrated the advantages of the proposed adaptive scheme over a non-adaptive scheme.

6.2 Future Research Directions

Currently, many wireless applications and devices are emerging, and this trend is expected to continue in foreseeable future. There also remain many problems which require further investigation as described below.

- **Energy-aware adaptive QoS:** most wireless devices are battery-powered, and hence it is desirable that protocols/systems for wireless/mobile networking should be energy-efficient. There has been significant research into energy-efficient communication protocols, e.g., in link-layer error control [57, 20], in routing level [91], and in transport layer [54]. Especially, one can extend or modify our bandwidth management for adaptive QoS so as to meet the original design goal of maximizing the service provider's reward while saving the battery energy of mobiles as much as possible. This energy saving scheme may be implemented intelligently according to the energy balance, or the current battery life, as well as the corresponding application/connection QoS requirements.

- **Home networks:** future homes are expected to require communications among many different electronic devices such as TV, audio/video systems, personal computers, and home security systems, while high-speed access to the backbone network, i.e., the Internet, is available via emerging technologies such as cable modem and asynchronous digital subscriber loop (ADSL). Wireless technologies are considered as an attractive candidate for future home networks, e.g., HomeRF and SWAP [73]. This type of networks should support heterogeneous traffic including audio, video, and data, so QoS-provisioning is essential. The underlying environment can be considered as WLANs in terms of its geographical coverage. Two distinctive characteristics of home networks are: (1) most communication devices are stationary, so mobility support is not a concern; and (2) it is most likely that there will be no centralized access point or BS, so every communication mechanisms should be distributed. The first fact simplifies the problem of QoS-provisioning in some sense while the second makes the problem even more difficult.

- **Personal digital assistants (PDAs):** thanks to the development of powerful-yet-low-cost microprocessors and OSs as well as many useful applications, PDAs are becoming popular. Currently, PDAs can access existing wide-area cellular infrastructures via cell phone-connected modems or their own wireless modems (e.g., 3COM Palm VII connected organizer). Meanwhile, there are emerging radio-based low-power low-cost technologies for short-range communications, e.g., Bluetooth [40], and these will work as a means of communication in *personal area networks*, which are formed in an ad hoc fashion with such short-range communication devices as laptops, digital cameras, cell phones, and PDAs. Whether it is connected to a wide-area cellular network or a personal area network, a PDA should be able to provide adequate communication quality to its user depending on the running applications and the environment. At present, most applications for PDAs are expected to be involved with only text data and images due to their functional limitations in terms of computing power and display capability as well as limited batter power and wireless data transmission rates. For the same reasons, high-density images, for example, located in a web server, are also not desirable for PDAs. However, every web server does not provide a different version of web pages and data depending on the situation of each web browsing application. One possible solution is to place proxy servers in the wired infrastructure which, for example, can trans-code an image destined for a PDA into a degraded version [102]. Another interesting problem arises in personal area networks, in which many devices communicate with one another. In this environment, fair sharing of a scarce wireless link among different devices, again without a centralized controller, is a very interesting problem.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] P. Agrawal *et al.*, "SWAN: A mobile multimedia wireless network," *IEEE Personal Communications*, pp. 18–33, April 1996.

[2] H. Ahmadi, A. Krishna, and R. O. LaMaire, "Design issues in wireless LANs," *Journal of High-Speed Networks*, vol. 5, no. 1, pp. 87–104, 1996.

[3] S. Aridhi and C. L. Despins, "Performance analysis of type-I and type-II hybrid ARQ protocols using concatenated codes in a DS-CDMA Rayleigh fading channel," in *Proc. IEEE International Conference on Universal Personal Communications (ICUPC'95)*, pp. 748–752, 1995.

[4] A. Balachandran, A. T. Campbell, and M. E. Kounavis, "Active filters: delivering scaled media to mobile devices," in *Proc. Intn'l Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'97)*, pp. 133–142, May 1997.

[5] A. Banerjea, D. Ferrari, B. Mah, M. Moran, D. Verma, and H. Zhang, "The Tenet real-time protocol suite: design, implementation, and experience," *IEEE/ACM Trans. on Networking*, vol. 4, no. 1, pp. 1–10, February 1996.

[6] P. Bhagwat, P. Bhattacharya, A. Krishna, and S. K. Tripathi, "Using channel state dependent packet scheduling to improve TCP throughput over wireless LANs," *ACM Wireless Networks*, vol. 3, pp. 91–102, 1997.

[7] D. E. Borth, "An overview of personal communication systems," in *Microsystems Technology for Multimedia Application: An Introduction*, B. Sheu, M. Ismail, and E. Sanchez-Sinencio, editors, IEEE Press, New York, 1995.

[8] P. T. Brady, "A statistical analysis of on-off patterns in 16 conversations," *Bell Syst. Tech. J.*, vol. 47, pp. 73–91, January 1968.

[9] K. Brown and S. Singh, "A network architecture for mobile computing," in *Proc. IEEE INFOCOM'96*, pp. 1388–1396, 1996.

[10] A. T. Campbell, "Mobiware: QoS-aware middleware for mobile multimedia communications," in *Proc. IFIP Conf. on High Performance Networking*, 1997.

[11] J. I. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE Trans. on Information Theory*, vol. IT-25, pp. 505–515, September 1979.

[12] J. M. Capone and I. Stavrakakis, "Delivering diverse delay/dropping QoS requirements in a TDMA environment," in *Proc. ACM/IEEE MobiCom'97*, pp. 110–119, 1997.

[13] C.-S. Chang *et al.*, "Guaranteed quality-of-service wireless access to ATM networks," *IEEE Journal on Selected Areas in Communications*, pp. 106–118, January 1997.

[14] S. Choi and K. G. Shin, "Centralized wireless MAC protocols using slotted ALOHA and dynamic TDD transmission," *Performance Evaluation*, vol. 27 & 28, pp. 331–346, October 1996.

[15] S. Choi and K. G. Shin, "A cellular local area network with QoS guarantees for heterogeneous traffic," in *Proc. IEEE INFOCOM'97*, pp. 1032–1039, Kobe, Japan, April 1997.

[16] S. Choi and K. G. Shin, "Uplink CDMA systems with diverse QoS guarantees for heterogeneous traffic," in *Proc. ACM/IEEE MobiCom'97*, pp. 120–130, Budapest, Hungary, September 1997.

[17] S. Choi and K. G. Shin, "A cellular wireless local area network with QoS guarantees for heterogeneous traffic," *ACM Mobile Networks and Applications (MONET)*, vol. 3, no. 1, pp. 89–100, 1998. special issue on Mobile Multimedia Communications.

[18] S. Choi and K. G. Shin, "Comparison of connection-admission control schemes in the presence of hand-offs in cellular networks," in *Proc. ACM/IEEE MobiCom'98*, pp. 264–275, Dallas, Texas, October 1998.

[19] S. Choi and K. G. Shin, "Predictive and adaptive bandwidth reservation for hand-offs in QoS-sensitive cellular networks," in *Proc. ACM SIGCOMM'98*, pp. 155–166, Vancouver, British Columbia, September 1998.

[20] S. Choi and K. G. Shin, "Cost-effective adaptive error control using a hybrid of FEC and ARQ in wireless networks," Technical Report CSE-TR-402-99, University of Michigan, June 1999.

[21] S. Choi and K. G. Shin, "A comparative study of bandwidth reservation and admission control schemes in QoS-sensitive cellular networks," accepted to *ACM Wireless Networks*, April 1999.

[22] S. Choi and K. G. Shin, "An uplink CDMA system architecture with diverse QoS guarantees for heterogeneous traffic," to appear in *IEEE/ACM Trans. on Networking*, October 1999.

[23] R. D. Cideciyan and E. Eleftheriou, "Concatenated Reed-Solomon/convolutional coding scheme for data transmission in CDMA cellular systems," in *Proc. IEEE VTC'94*, pp. 1369–1373, 1994.

[24] J. Conan, "The weight spectra of some short low-rate convolutional codes," *IEEE Trans. on Communications*, vol. 32, pp. 1050–1053, September 1984.

[25] C. I. Cook, "Development of air interface standards for PCS," *IEEE Personal Communications*, pp. 30–34, Fourth Quarter 1994.

[26] L. M. Correia and R. Prasad, "An overview of wireless broadband communications," *IEEE Communications Magazine*, pp. 28–33, January 1997.

[27] D. C. Cox, "Wireless personal communications: What is it?," *IEEE Personal Communications*, pp. 20–35, April 1995.

[28] B. P. Crow, I. Widjaja, J. G. Kim, and P. T. Sakai, "IEEE 802.11 wireless local area networks," *IEEE Communications Magazine*, September 1997.

[29] E. Dahlman, B. Gudmundson, M. Nilsson, and J. Skold, "UMTS/IMT-2000 based on wideband CDMA," *IEEE Communications Magazine*, pp. 70–80, September 1998.

[30] D. Duchamp and N. Reynolds, "Measured performance of a wirelss LAN," in *Proc. 17th Conference on Local Computer Networks*, pp. 494–499, September 1992.

[31] D. Eckhardt and P. Steenkiste, "Measurement and analysis of the error characteristics of an in-building wireless network," in *Proc. ACM SIGCOMM'96*, pp. 243–254, 1996.

[32] D. A. Eckhardt and P. Steenkiste, "Improving wireless LAN performance via adaptive local error control," in *Proc. IEEE ICNP'98*, pp. 327–338, 1998.

[33] M. Elaoud and P. Ramanathan, "Adaptive use of error-correcting codes for real-time communication in wireless networks," in *Proc. IEEE INFOCOM'98*, pp. 548–555, 1998.

[34] K. Y. Eng *et al.*, "BAHAMA: A broadband ad-hoc wireless ATM local-area network," in *Proc. IEEE ICC'95*, pp. 1216–1223, 1995.

[35] W. W. Erdman, "Wireless communications: A decade of progress," *IEEE Communications Magazine*, vol. 31, no. 12, pp. 48–51, December 1993.

[36] N. R. Figueira and J. Pasquale, "Remote-queueing multiple access (RQMA): providing quality of service for wireless communications," in *Proc. IEEE INFOCOM'98*, 1998.

[37] C. Fragouli, V. Sivaraman, and M. B. Srivastava, "Controlled multimedia wireless link sharing via enhanced class-based queuing with channel-state-dependent packet scheduling," in *Proc. IEEE INFOCOM'98*, March 1998.

[38] K. S. Gilhousen, I. M. Jacobs, R. Padovani, A. J. Viterbi, L. A. Weaver, Jr., and C. E. Wheatley III, "On the capacity of a cellular CDMA system," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 2, pp. 303–312, May 1991.

[39] N. Guo, S. D. Morgera, and P. Mermelstein, "Common packet data channel (CPDC) for integrated wireless DS-CDMA networks," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 4, pp. 735–749, May 1996.

[40] J. Haartsen *et al.*, "Bluetooth: vision, goal, and architecture," *ACM Mobile Computing and Communications Review*, vol. 2, no. 4, pp. 38–45, October 1998.

[41] R. J. Haines and A. H. Aghvami, "Indoor radio environment considerations in selecting a media access control protocol for wideband radio data communications," in *Proc. IEEE ICC'93*, pp. 990–994, 1993.

[42] S. Han, *Fast low-cost failure recovery for real-time communication in multi-hop networks*, PhD thesis, Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI, 1998.

[43] V. Hayes, "Standardization effects for wireless lans," *IEEE Network Magazine*, pp. 19–20, November 1991.

[44] A. Hoffmann, R. J. Haines, and A. H. Aghvami, "Performance analysis of a token based MAC protocol with asymmetric polling strategy ("TOPO") for indoor radio local area networks under channel outage conditions," in *Proc. IEEE ICC'94*, pp. 1306–1311, 1994.

[45] D. Hong and S. S. Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized procedures," *IEEE Trans. on Vehicular Technology*, vol. 35, no. 3, pp. 77–92, August 1986.

[46] C.-L. I and R. D. Gitlin, "Multi-code CDMA wireless personal communications networks," in *Proc. IEEE ICC'95*, pp. 1060–1064, June 1995.

[47] C.-L. I, P. Pollini, L. Ozarow, and R. D. Gitlin, "Performance of multi-code CDMA wireless personal communications networks," in *Proc. IEEE VTC'95*, pp. 907–911, July 1995.

[48] C.-L. I and K. K. Sabnani, "Variable spreading gain CDMA with adaptive control for true packet switching wireless network," in *Proc. IEEE ICC'95*, pp. 725–730, June 1995.

[49] S. Jamin, P. Danzig, S. Shenker, and L. Zhang, "Measurement-based admission control algorithm for integrated service packet networks," *IEEE/ACM Trans. on Networking*, vol. 5, no. 1, pp. 56–70, February 1997.

[50] I. Joe, "An adaptive hybrid ARQ scheme with concatenated FEC codes for wireless ATM," in *Proc. ACM/IEEE MobiCom'97*, pp. 131–138, 1997.

[51] D. D. Kandlur, K. G. Shin, and D. Ferrari, "Real-time communication in multi-hop networks," *IEEE Trans. on Parallel and Distributed Systems*, vol. 5, no. 10, pp. 1044–1056, October 1994.

[52] M. J. Karol, Z. Liu, and K. Y. Eng, "Distributed-queueing request update multiple access (DQRUMA) for wireless packet (ATM) networks," in *Proc. IEEE ICC'95*, pp. 1224–1231, 1995.

[53] K. Keeton, B. A. Mah, S. Seshan, R. H. Katz, and D. Ferrari, "Providing connection-oriented network services to mobile hosts," in *Proc. of the USENIX symposium on mobile and location-independent computing*, August 1993.

[54] R. Kravets and P. Krishnan, "Power management techniques for mobile communication," in *Proc. ACM/IEEE MobiCom'98*, pp. 157–168, October 1998.

[55] S.-K. Kweon, *Real-time communication in integrated services packet networks*, PhD thesis, Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI, 1998.

[56] K. Lee, "Supporting mobile multimedia in integrated service networks," *ACM Wireless Networks*, vol. 2, pp. 205–217, 1996.

[57] P. Lettieri, C. Fragouli, and M. B. Srivastava, "Low power error control for wireless links," in *Proc. ACM/IEEE MobiCom'97*, pp. 139–150, 1997.

[58] D. A. Levine, I. F. Akyildiz, and M. Naghshineh, "A resource estimation and call admission algorithm for wireless multimedia networks using the shadow cluster concept," *IEEE/ACM Trans. on Networking*, vol. 5, no. 1, pp. 1–12, February 1997.

[59] R. R.-F. Liao and A. Campbell, "On programmable universal mobile channels in a cellular internet," in *Proc. ACM/IEEE MobiCom'98*, pp. 191–202, October 1998.

[60] Y.-B. Lin, "Reducing location update cost in a PCS network," *IEEE/ACM Trans. on Networking*, vol. 5, no. 1, pp. 25–33, February 1997.

[61] M. Listanti, F. Mascitelli, and A. Mobilia, "D^2MA: A distributed access protocol for wireless ATM networks," in *Proc. IEEE INFOCOM'98*, 1998.

[62] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in a hard-real-time environment," *Journal of ACM*, vol. 20, no. 1, pp. 46–61, January 1973.

[63] Z. Liu, M. J. Karol, M. El Zarki, and K. Y. Eng, "Channel access and interference issues in multi-code DS-CDMA wireless packet (ATM) networks," *Wireless Networks*, vol. 2, pp. 173–193, 1996.

[64] S. Lu, K.-W. Lee, and V. Bharghavan, "Adaptive service in mobile computing environments," in *Proc. Intn'l Workshop on Quality of Service'97*, 1997.

[65] S. Lu, T. Nandagopal, and V. Bharghavan, "A wireless fair service algorithm for packet cellular networks," in *Proc. ACM/IEEE MobiCom'98*, pp. 10–20, October 1998.

[66] S. Lu and V. Bharghavan, "Adaptive resource management algorithms for indoor mobile computing environments," in *Proc. ACM SIGCOMM'96*, pp. 231–242, August 1996.

[67] S. Lu, V. Bharghavan, and R. Srikant, "Fair scheduling in wireless packet networks," in *Proc. ACM SIGCOMM'97*, pp. 63–74, September 1997.

[68] Lucent Technologies. *IEEE 802.11 WaveLAN PC card - user's guide*, 1998.

[69] R. J. McEliece and L. Swanson, "On the decoder error probability for Reed-Solomon codes," *IEEE Trans. on Information Theory*, vol. IT-32, no. 5, pp. 701–703, September 1986.

[70] P. P. Mishra and M. Srivastava, "Effect of virtual circuit rerouting on application performance," in *Proc. IEEE ICDCS'97*, 1997.

[71] M. Naghshineh and M. Schwartz, "Distributed call admission control in mobile/wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 4, pp. 711–717, May 1996.

[72] M. Naghshineh and M. Willebeek-LeMair, "End-to-end QoS provisioning in multimedia wireless/mobile networks using an adaptive framework," *IEEE Communications Magazine*, vol. 35, no. 11, pp. 72–81, November 1997.

[73] K. J. Negus *et al.*, "HomeRF and SWAP: wireless networking for the connected home," *ACM Mobile Computing and Communications Review*, vol. 2, no. 4, pp. 28–37, October 1998.

[74] T. S. E. Ng, I. Stoica, and H. Zhang, "Packet fair queueing algorithms for wireless networks with location dependent errors," in *Proc. IEEE INFOCOM'98*, pp. 1103–1111, March 1998.

[75] T. Ojanpera and R. Prasad, "An overview of air interface multiple access for IMT-2000/UMTS," *IEEE Communications Magazine*, pp. 82–95, September 1998.

[76] T. Ottosson and A. Svensson, "Multi-rate schemes in DS/CDMA systems," in *Proc. IEEE VTC'95*, pp. 1006–1010, 1995.

[77] K. Pahlavan, A. Zahedi, and P. Krishnamurthy, "Wideband local access: wireless LAN and wireless ATM," *IEEE Communications Magazine*, pp. 34–40, November 1997.

[78] K. Pahlavan and A. H. Levesque, *Wireless Information Networks*, Wiley-Interscience, New York, NY, 1995.

[79] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 3rd edition, 1991.

[80] A. K. Parekh, *A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks*, PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, February 1992.

[81] N. Passas, S. Paskalis, D. Vali, and L. Merakos, "Quality-of-service-oriented medium acess control for wireless ATM networks," *IEEE Communications Magazine*, pp. 42–50, November 1997.

[82] C. E. Perkins, "Mobile IP," *IEEE Communications Magazine*, pp. 84–99, May 1997.

[83] J. G. Proakis, *Digital Communications, 2nd ed.*, McGraw Hill, New York, NY, 1989.

[84] M. B. Pursley, "Performance evaluation for phase-coded spread-spectrum multiple-access communication – part I: system analysis," *IEEE Trans. on Communications*, vol. 25, pp. 795–799, August 1977.

[85] S. Rajagopalan and B. R. Badrinath, "An adaptive location management strategy for mobile IP," in *Proc. ACM MobiCom'95*, 1995.

[86] P. Ramanathan and P. Agrawal, "Adapting packet fair algorithms to wireless networks," in *Proc. ACM/IEEE MobiCom'98*, pp. 1–9, October 1998.

[87] R. Ramjee, R. Nagarajan, and D. Towsley, "On optimal call admission control in cellular networks," in *Proc. IEEE INFOCOM'96*, pp. 43–50, 1996.

[88] D. Raychaudhuri *et al.*, "WATMnet: A prototype wireless ATM system for multimedia personal communication," *IEEE Journal on Selected Areas in Communications*, pp. 83–95, January 1997.

[89] K. Rijkse, "H.263: Video coding for low-bit-rate communication," *IEEE Communications Magazine*, pp. 42–45, December 1996.

[90] A. A. Shaikh, *Efficient dyanmic routing in wide-area networks*, PhD thesis, Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI, 1999.

[91] S. Singh, M. Woo, and C. S. Raghavendra, "Power-aware routing in mobile ad-hoc networks," in *Proc. ACM/IEEE MobiCom'98*, pp. 181–190, October 1998.

[92] M. Soroushnejad and E. Geraniotis, "Multi-access strategies for an integrated voice/data CDMA packet radio network," *IEEE Trans. on Communications*, vol. 43, no. 2/3/4, pp. 934–945, February/March/April 1995.

[93] R. Steele, "The evolution of personal communications," *IEEE Personal Communications*, pp. 6–11, Second Quarter 1994.

[94] F. Swarts and H. Ferreira, "Markov characterization of digital fading mobile VHF channels," *IEEE Trans. on Vehicular Technology*, pp. 977–985, November 1994.

[95] A. K. Talukdar, B. R. Badrinath, and A. Acharya, "On accommodating mobile hosts in an integrated services packet network," in *Proc. IEEE INFOCOM'97*, pp. 1048–1055, April 1997.

[96] A. K. Talukdar, B. R. Badrinath, and A. Acharya, "Rate adaptation schemes in networks with mobile hosts," in *Proc. ACM/IEEE MobiCom'98*, pp. 169–180, October 1998.

[97] F. A. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part III-polling and (dynamic) split-channel reservation multiple access," *IEEE Trans. on Communications*, vol. 24, no. 8, pp. 832–845, August 1976.

[98] C. K. Toh, "A hybrid handover protocol for local area wireless ATM networks," *ACM Mobile Networks & Applications (MONET)*, vol. 1, no. 3, pp. 313–334, 1996.

[99] A. J. Viterbi, *CDMA: Principles of Spread Spectrum Communication*, Addison-Wesley, Reading, MA, 1995.

[100] H. Wang and N. Moayeri, "Finite state morkov channel – a useful model for radio communication channels," *IEEE Trans. on Vehicular Technology*, pp. 163–171, February 1995.

[101] R. Yates, C. Rose, S. Rajagopalan, and B. Badrinath, "Analysis of a mobile-assisted adaptive location management strategy," *ACM Mobile Networks & Applications (MONET)*, vol. 1, no. 2, pp. 105–112, 1996.

[102] B. Zenel and D. Duchamp, "A general purpose proxy filtering mechanism applied to the mobile nevironment," in *Proc. ACM MobiCom'97*, pp. 248–259, September 1997.

[103] H. Zhang, *Service Disciplines For Packet-Switching Integrated-Services Networks*, PhD thesis, Department of Computer Science, The University of California, Berkeley, CA, 1993.

[104] Z. Zhang and A. S. Acampora, "Performance of a modified polling strategy for broadband wireless LANs in a harsh fading environment," in *Proc. IEEE GLOBECOM'91*, pp. 1141–1146, 1991.

[105] Y. Zhao, *Vehicle Location and Navigation Systems*, Artech House, 1997.

[106] Q. Zheng, *Real-Time Fault-Tolerant Communication in Computer Network*, PhD thesis, Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI, 1993.