Scheduling and Error Control in Cellular and Multi-hop Wireless Networks: Analysis and Optimization

by

TEERAWAT ISSARIYAKUL

A Thesis submitted to the Faculty of Graduate Studies of

The University of Manitoba

in partial fulfilment of the requirements of the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical and Computer Engineering

University of Manitoba

Winnipeg

Copyright © 2005 by Teerawat Issariyakul

0-494-08784-6

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN:
Our file *Notre référence*
ISBN:

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION

Scheduling and Error Control in Cellular and Multi-hop Wireless Networks: Analysis and Optimization

BY

Teerawat Issariyakul

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirement of the degree

Of

Doctor of Philosophy

Teerawat Issariyakul © 2005

Permission has been granted to the Library of the University of Manitoba to lend or sell copies of this thesis/practicum, to the National Library of Canada to microfilm this thesis and to lend or sell copies of the film, and to University Microfilms Inc. to publish an abstract of this thesis/practicum.

This reproduction or copy of this thesis has been made available by authority of the copyright owner solely for the purpose of private study and research, and may only be reproduced and copied as permitted by copyright laws or with express written authorization from the copyright owner.

Supervisor: Professor Ekram Hossain

ABSTRACT

Next-generation wireless networks are expected to provide ubiquitous broadband connectivity for multimedia services. Protocols and architectures for such networks will build upon the existing cellular and multi-hop wireless networking concepts and technologies. This dissertation deals with analysis and optimization of radio link level scheduling and error control protocols in cellular and multi-hop wireless networks, respectively.

For a cellular wireless network, the performance of a channel-quality-based opportunistic scheduling which maximizes system throughput at the link layer is analyzed using a Markov process. However, this type of scheduling rules can result in severe unfairness. Therefore, two optimization-based approaches to solve the wireless fair-queuing problem under a Time Division Multiple Access (TDMA)-based Medium Access Control (MAC) framework are proposed. By formulating a fair scheduling problem as an assignment problem, this dissertation proposes Optimal Radio Channel Allocation for Single-Rate Transmission (ORCA-SRT) and Optimal Radio Channel Allocation for Multi-Rate Transmission (ORCA-MRT) for fair bandwidth allocation in wireless data networks which support single-rate and multi-rate transmission at the radio link level. ORCA-SRT exhibits better performance than all other fair queuing algorithms in the literature, and provides a basis for ORCA-MRT. The key feature of ORCA-MRT is that while allocating transmission rate to each flow fairly it keeps the inter-access delay bounded. Two channel prediction models are proposed and extensive simulations are conducted to investigate the performance of ORCA-SRT and ORCA-MRT for different system parameters such as channel state correlation, number of flows, etc.

To evaluate the reliability and latency trade-off in a multi-hop wireless network, two analytical models are presented. The first one models the required number of transmissions for successful delivery of a packet over an H-hop wireless path. The second model derives complete statistics for end-to-end latency and reliability of a transmission of a batch of packets under multi-rate transmission. Both of the models take the effect of different Automatic Repeat reQuest (ARQ)-based error control

mechanisms into account.

The developed analytical models would be useful in analysis and optimization of cellular, multi-hop, and hybrid wireless networks such as multi-hop cellular networks.

Examiners:

Professor Ekram Hossain, Supervisor, Dept. of Electrical & Computer Engineering

---

Professor Robert D. McLeod, Member, Dept. of Electrical & Computer Engineering

---

Professor Miroslaw Pawlak, Member, Dept. of Electrical & Computer Engineering

---

Professor Dean K. McNeill, Member, Dept. of Electrical & Computer Engineering

---

Professor Jelena Misic, Outside Member, Dept. of Computer Science

---

Professor Victor C. M. Leung, External Examiner

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ARQ	Automatic Repeat reQuest
AMC	Adaptive Modulation and Coding
BER	Bit Error Rate
cdf	Cumulative Distribution Function
CDMA	Code Division Multiple Access
CDMA-HDR	Code Division Multiple Access with High Data Rate
CIF-Q	Channel Independent Fair Queuing
CTMC	Continuous-Time Markov Chain
DCF	Distributed Coordination Function
DTMC	Discrete-Time Markov Chain
FEC	Forward Error Correction
FDMA	Frequency Division Multiple Access
FMC	Fixed Modulation and Coding
FSMC	Finite State Markov Channel
FTP	File Transfer Protocol
GEC	Gilbert-Elliot Channel
GPS	Generalized Processor Sharing
IWFQ	Idealized Wireless Fair Queuing
M-QAM	M-ary Quadrature Amplitude Modulation
MAC	Medium Access Control
MFS	Multi-channel Fair Scheduler
OPP	OPPortunistic scheduling
ORCA-MRT	Optimal Radio Channel Allocation for Multi-Rate Transmission
ORCA-SRT	Optimal Radio Channel Allocation for Single-Rate Transmission
PDA	Personal Digital Assistant
pdf	probability density function
pmf	probability mass function

QoS	Quality of Service
RR	Round Robin
rst	connection reset state
RSC	Random State Channel
S.D.	Standard Deviation
sc	transmission success state
SINR	Signal-to-Interference plus Noise Ratio
TAPs	Transit Access Points
TCP	Transmission Control Protocol
TDM	Time Division Multiplexing
TDMA	Time Division Multiple Access
TPM	Transition Probability Matrix
UMTS	Universal Mobile Telecommunications System
WCDMA	Wideband Code Division Multiple Access
WFQ	Wireless Fair Queuing
WFS	Wireless Fair Service Algorithm
WLAN	Wireless Local Area Network
WPS	Wireless Packet Scheduling
WWAN	Wireless Wide Area Network

List of Symbols

$\mathbf{0}$ An all-zero matrix, 11

$\mathbf{I}$ An identity matrix, 11

$\boldsymbol{e}$ An all-one matrix, 11

$\delta^{ARQ}_{(lb,ub)}$ Residual improvement of ARQ, compared to lb and ub, 96

γ_{mob} Per-mobile throughput, 23

γ_{sys} System throughput, 23

$\mathscr{D}^{(k)}_{acc_i}$ Inter-access delay of mobile i from the $(k-1)^{th}$ and the k^{th} access opportunities, 52

$\mathscr{D}_{acc}$ Inter-access delay, 23

$\mathscr{D}_{e2e}$ End-to-end latency, 110

$\mathscr{D}_{rst}$ Connection reset delay, 24

$\mathscr{D}_{sc}$ Inter-success delay, 24

$\mathscr{S}$ Number of packets successfully delivered to the destination, 110

$\pi_m = [\boldsymbol{\pi}]_{1,m}$ Probability that the channel state at steady state is m, 16

ρ Average channel state correlation, 17

$\rho^{(i)}$ Channel state correlation in state i, 17

$\mathbf{C}_{MRT}$ Multi-rate transmission cost matrix, 69

$\mathbf{C}_{SRT}$ Single-rate transmission cost matrix, 50

ξ Conditional packet-dropping/connection-reset probability, 87

c^{CA}_{ij} Channel-aware cost of mobile i in slot j, 65

C_T Total cost, 51

c_{ij} Cost of assigning row i to column j, 51

$f_{\gamma^{err}_{sys}}(s)$ Distribution of system throughput under non-zero packet error probability, 26

$f_{\gamma_{sys}}(m)$ Distribution of system throughput under error-free wireless channel, 25

$f^{(t)}_{\mathcal{M}}(m) = \left[\boldsymbol{f}^{(t)}_{\mathcal{M}}\right]_{1,m}$ Probability that of the channel state in time slot t is m, 16

K Retransmission Limit, 23

L_i Lag counter of mobile i, 66

M Number of channel or FSMC states, 18

m'_{ij} Predicted channel state of mobile i in slot j, 63

n Total number of mobiles, 22

n_f Number of mobiles whose channels are modeled by FSMCs, 34

n_r Number of mobiles whose channels are modeled by RSCs, 34

n_{fin} Number of access opportunities until either a connection reset or a transmission success, 28

n_{rst} Number of access opportunities until a connection reset, 28

n_{sc} Number of access opportunities until a transmission success, 28

p_{err_c} Average packet error probability due to data collision, 85

p_{err_f} Average packet error probability due to fading, 86

$p^{(s)}_{err_f}$ Packet error probability due to fading when the channel state is s, 86

p_{err} Average packet error probability, 17

$p^{(m)}_{err}$ Packet error probability when channel state is m, 17

$p_{ij} = [\mathbf{P}]_{ij}$ Transition probability from state i to j, 12

p_{suc} Probablity that a packet is successfully tranmsitted, 86

$p^{(t)}_{tx}(m)$ Probability that a mobile will get a channel access at time slot t with tranmsission rate m, 24

r^{ARQ}_{link} Link reliability for a given ARQ protocol, 87

T_F Scheduling frame size, 51

w_i weight of flow i, 5

x_{ij} Solution of the assignment problem, 51

Acknowledgement

First and foremost, I would like to express my profound gratitude and appreciation to my supervisor, Professor Ekram Hossain, for his continuous support and invaluable guidance. Professor Ekram Hossain was a very understanding and open-minded mentor. He has opened my mindset to a new direction of research. I am very grateful to Professor Robert D. McLeod, Professor Miroslaw Pawlak, Professor Dean K. McNeill, and Professor Jelena Misic for being in my examination committee. I am also obliged to Professor Victor C. M. Leung for being the external examiner in my Ph.D. oral examination. My great appreciation also goes to Dr. Jeff Diamond and Professor Attahiru S. Alfa for their time and technical knowledge, which helped me a lot in my research work.

I would like to recognize Telecommunications Research Laboratory (TR*Labs*), Natural Sciences and Engineering Research Council of Canada (NSERC), and University of Manitoba for financial supports. The research environment in TR*Labs*, Winnipeg, has been very resourceful for me. Many thanks go to the staff in TR*Labs*, Winnipeg. Thuraiappah Vaseevaran, Carolyn Christman, and Julie Stewart, you guys have always been very helpful.

To all my friends in Winnipeg, it has been my pleasure spending my time with you guys. Dusit (Tao) Niyato, discussion with you has always led to something relaxing and interesting. Wattamon (Goog) Srisakuldee, I always appreciate your friendship. To my friend from AIT, Poramate (Pom) Tarasak, thanks for introducing me to my supervisor. To my good friend, Mea Wang, thanks for sharing knowledge and experience even after you left.

Last but not the least, I would like to thank all my family members for their continuous moral support. For my mom, Doungporn Issariyakul, I thank you for pushing me since I was young. For my aunt, Nopparat Issariyakul (Na Kaew), I thank you for always believing in me and never giving up on me. And to my dear lady, Wannasorn (Golf) Kruahongs, I thank you for having been there for me from the beginning. Your moral support and companionship have leveraged the hardship during all these years. Your presence has changed all these years from just a study to "something to remember". Thank you!

Dedication

To my mom whose unconditional love is unparalleled.
To my aunt (Na Kaew) whose goodwill and belief have never run out.

Chapter 1

Introduction

During last few years, wireless technologies have experienced phenomenal growth. Mobile devices such as laptops, *Personal Digital Assistant (PDA)*, and mobile phones have become more sophisticated and more proliferated. Categorized as the *Wireless Wide Area Network (WWAN)*, the third generation (3G) wireless systems such as *Universal Mobile Telecommunications System (UMTS)* or *Wideband Code Division Multiple Access (WCDMA)* are designed for vast coverage area, but can support only low data rate (e.g., 144 Kbps-2 Mbps) [1]. On the other hand, *Wireless Local Area Network (WLAN)*–such as IEEE 802.11 or HiperLAN–can provide high data rate (e.g., 1 Mbps–54 Mbps), however, with coverage no more than few hundred meters [2],[3]. In the next generation wireless systems, the merits of both WWANs and WLANs are expected to be incorporated to provide near ubiquitous coverage, flexible personalization, and support for multimedia at affordable transmission cost [4]. Design, analysis, and optimization of efficient/optimal wireless protocols for next generation wireless networks is a grand research challenge.

1.1 "Wireless" Protocol Stack

Wireless protocol stack consists of the following layers (Figure 1.1):

- **Physical layer**: This layer defines electrical, functional, and procedural characteristics for bit-stream transmission over a wireless channel.
- **Radio link layer**: Radio link layer defines protocols to transmit data between two nodes. Its responsibilities include the followings. First, it groups information and control (e.g., error checking/correcting and frame synchronization) bits,

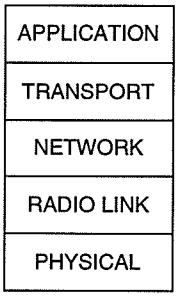


Figure 1.1. *Wireless protocol stack.*

and constructs a data frame. Secondly, it controls transmission error by means of retransmission (also known as *Automatic Repeat reQuest (ARQ)* schemes) and/or forward error correction (FEC). Thirdly, it defines medium access control (MAC) protocol, that is, how each node should access the shared wireless channel. Fourthly, it schedules multiple data flows at a particular node. Finally, it also performs link level buffer management functionalities. This dissertation focuses on radio link layer scheduling and error control protocols.

- **Network layer**: When a source node cannot directly reach the destination node, the network layer determines relay nodes which are responsible for forwarding data from the source to the destination. Due to network dynamics, it also needs to keep track of route status, and locates a new route when the current route is broken.
- **Transport layer**: The main responsibility of this layer is to supervise reliable data transmission at an end-to-end level. It controls data transmission rate to avoid network congestion. Both error control and congestion control in this layer operate only at the source and destination nodes. From an end-to-end perspective, packet reception pattern at the destination node reflects congestion and/or error in the intermediate nodes, and a transport protocol is designed to react only to this pattern.
- **Application layer**: This layer operates only in source or destination nodes. It employs the transport layer service for data transmission. Its main responsibilities include providing data to the transport layer, establishing a secured connection, interfacing with a user, and so forth.

Since a particular system consists of many aspects, designing the system as a whole is usually deemed too complicated. The wireless protocol stack provides a separation of the system functionalities so that each part can be designed separately. Recently, however, there has been a growing interest in *cross-layer* design for wireless protocols to optimize the entire transmission protocol stack performance (at the expense of more complexity in the system design).

1.2 Scheduling and Error Control in Cellular Wireless Networks

A cellular wireless network is a wide area network, which divides the entire service area into several sub-areas called *cells* (circles in Figure 1.2). Each cell is controlled by a *base station*. Major responsibilities of a base station include controlling resources within its jurisdiction and coordinating with other base stations to setup connections as well as to handle mobiles' mobility. This dissertation focuses on scheduling and error control mechanisms[1] of a base station in a particular cell.

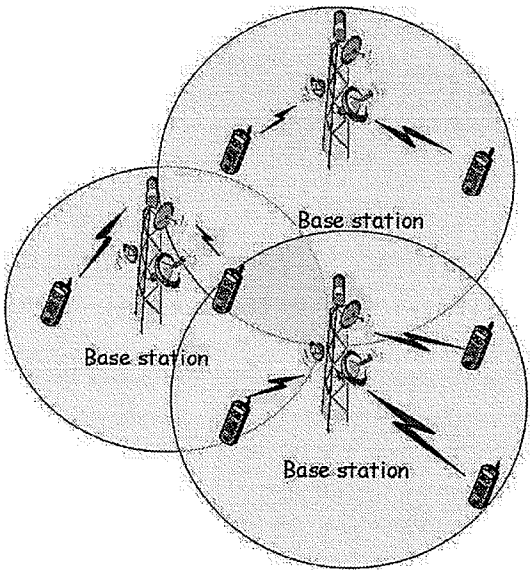


Figure 1.2. *A cellular wireless network.*

[1]From the wireless protocol stack perspective, these functionalities lie in the radio link layer.

One of the most important aspects in a cellular wireless network is *resource sharing*. Resource sharing (also known as *channel access*) defines a set of rules for each mobile to use the resource without incurring unacceptably high interference to other active mobiles. Conventionally, there are three resource sharing mechanisms:

- *Time Division Multiple Access (TDMA)*: Each mobile is granted a channel access at different time. Since there is only one active mobiles, the interference is minimal. This dissertation focuses on this type of channel access.
- *Frequency Division Multiple Access (FDMA)*: Several mobiles are allowed to access the channel simultaneously. Different mobiles use non-interfering frequency bands.
- *Code Division Multiple Access (CDMA)*: Several transmissions in the same frequency band can occur concurrently. Each mobile is identified by a particular *Psudo-Noise* (PN) code sequence. At the receiver, a multi-user detection technique uses the code sequence to extract information of the corresponding mobile. To a particular flow, transmissions from other flows acts as noise. Therefore, the power control in CDMA is very critical.

In practice, a TDMA system divides time domain into logical units called *time slots*. During each time slot, the base station allows only one mobile to access the channel. The functionality of a base station to decide which mobile is allowed to access the channel in each time slot is known as *scheduling*. For example, a *Round Robin* scheduler allows each of the backlogged mobiles to transmit in order.

Although we mainly focus on a TDMA-based system, the results in this dissertation are also applicable for some advanced versions of CDMA systems. For example, implemented under a CDMA system, a *Code Division Multiple Access with High Data Rate (CDMA-HDR)* scheduler [5] transmits data to only one mobile. A base station in a CDMA-HDR system obtains channel state information by using the following approach. The base station broadcasts so-called *pilot signal* periodically. Intercepting the pilot signal, each mobile determines the current received SINR, looks up the SINR-to-rate mapping table, and requests for corresponding data rate via the *reverse link data rate request channel*. Based on the received request, the scheduler selects only one mobile to access the channel. Due to time-varying nature of a wireless channel, this *opportunistic scheduling* exploits so-called *multi-user diversity* [6] and yields

higher throughput than a conventional CDMA system does.

In a wireless network, a scheduler needs to be aware of channel variation. For example, to maximize overall throughput (number of transmitted packets over a certain period of time), a *channel-quality-based opportunistic scheduling* principle allows only a mobile with the best channel condition to access the channel (Chapter 3) [7].

Alternatively, fair scheduling algorithms aims at fairly allocating resources among all mobiles. In a wired network, fair scheduling algorithms are generally derived from the *Generalized Processor Sharing (GPS)* algorithm [8]. GPS generally allocates resources (e.g., channel bandwidth) among all data flows in proportion to their weights. In particular, the scheduler allocates $X_i(t_1, t_2)$ and $X_j(t_1, t_2)$ amount of resources to mobiles i and j (with weights w_i and w_j, respectively) in the interval from t_1 to t_2 such that, given that flow i is backlogged, the allocation satisfies

$$\frac{X_i(t_1, t_2)}{X_j(t_1, t_2)} \geqslant \frac{w_i}{w_j}, \qquad \forall i, j, \tag{1.1}$$

where the equality holds when both flow i and j are backlogged at the same time.

The location-dependent and bursty channel errors make the original notion of GPS unsuitable for wireless networks. Fair queuing algorithms, such as *Wireless Packet Scheduling (WPS)* [9], *Channel Independent Fair Queuing (CIF-Q)* [10], and *Wireless Fair Service Algorithm (WFS)* [11], have been proposed to alleviate this problem.

In general, a mobile perceiving a *bad* channel (i.e., low received *Signal-to-Interference plus Noise Ratio (SINR)* at the receiver) should defer the transmission and let another mobile with a *good* channel (i.e., high received SINR) transmit data. The scheduler should compensate for that when the channel of the deferred mobile becomes *good* again. The mobile deferring the transmission is considered to be *lagging*, while the mobile receiving extra allocation is considered to be *leading*. Most of the wireless fair queuing algorithms in the literature consist of five main components [12]:

- **The error-free service model** allocates resources among all data flows in absence of error.
- **The lead and lag model** determines which data flows are *leading* and *lagging*.
- **The compensation model** specifies how *lagging* data flows can be compensated.

- **Slot queues and packet queues** classify different types of data flows (e.g., delay sensitive flows, error sensitive flows).
- **Channel monitoring and prediction** provide measurement and estimation of the channel state.

With multi-rate transmission capability, a mobile can dynamically adjusts transmission rate based on the current channel condition. Therefore, the number of transmitted packets in each allocated time slot might be different. The notion of fairness under multi-rate transmission can have different implications.

Definition 1.1 TEMPORAL FAIRNESS *is the property of a scheduler to fairly allocate time slots among all the mobiles so that they will experience similar delay.*

Definition 1.2 THROUGHPUT FAIRNESS *is the property of a scheduler to fairly allocate transmission rates so that all the mobiles will transmit similar amount of data over a certain period of time.* □

Another important issues in a wireless network is *error control mechanism*, which can largely be categorized into *Forward Error Correction (FEC)* and *Automatic Repeat reQuest (ARQ)* schemes. FEC inserts redundancy bits into a data frame before each transmission. When a received data frame is in error, the receiver uses these redundancy bits to correct the corrupted data frame. ARQ, on the other hand, combats an error-prone wireless channel by means of data retransmission. While able to correct the corrupted data with low latency, FEC could lead to inefficiency in resource usage, due to redundant bits in each data frame. ARQ, on the other hand, is more adaptive to channel variation but requires more latency (due to retransmission) to correct the error. In general, FEC is more suitable to real-time traffic, while ARQ is more appropriate for non-real-time data transmission.

1.3 Multi-hop Wireless Networks

A multi-hop wireless network is characterized by the absence of a direct communication link between source and destination nodes. Data transmission in this case must first be transmitted to nearby relay nodes, which in turn forward data to the destination node. This class of networks has a wide range of applications such as wireless

ad hoc networks [13], wireless mesh networks [14], wireless sensor networks [15], and multi-hop cellular networks [16].

A multi-hop wireless network enables short-range communication while preserving broad service coverage. Short-range communications leads to higher received SINR. Therefore, transmitting power can be reduced without compromising packet error probability. The reduction in power requirement implies longer battery life time, and smaller interfering region which could result in an increase in spatial reuse. Alternatively, with fixed transmission power, short-range communications leads to decreased packet error probability, hence allowing higher modulation order transmission which is more susceptible to noise. Short-range communication not only improves average throughput, but also helps distribute services fairly among mobiles with poorer channel condition.

An example of multi-hop wireless networks is illustrated in Figure 1.3, where Mobiles 1 and 2 are close to the base station, and therefore their SINR levels are expected to be comparatively high. Mobile 3, on the other hand, is far from the base station and obstructed by a building, and therefore the link between Mobile 3 and the base station experiences high path loss and shadowing, leading to low SINR. Although increasing transmitting power can solve this problem, it might lead to more interference and inefficient energy usage. Another solution is to transmit data to Mobile 3 via a relay node. Transmission in a multi-hop manner could lead to better overall throughput [17].

1.4 Motivation and Scope of This Dissertation

This dissertation deals with the analysis and optimization of scheduling and error control protocols in cellular and multi-hop wireless networks.

1.4.1 Cellular Wireless Networks

For a cellular wireless network, this dissertation emphasizes on a single-cell in a TDMA system. Particularly, the major interest is in scheduling and error control in the physical and link layers.

This dissertation analyzes radio link level performances of a channel-quality-based

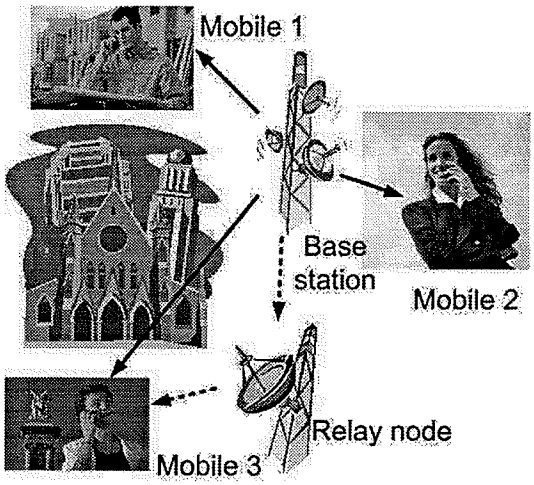


Figure 1.3. *A multi-hop cellular network.*

opportunistic scheduling under both correlated and uncorrelated wireless channels and under multi-rate transmission. The multi-rate transmission is assumed to be achieved through *Adaptive Modulation and Coding (AMC)* to adjust the transmission rate according to the channel condition. The residual error effect due to each AMC setting is counteracted by means of a limited persistence ARQ protocol. The novelty of the proposed analytical framework is the derivation of complete statistics (in terms of *probability mass function*) for both short-term and long-term performance measures such as system throughput, per-flow throughput, inter-success delay under both uncorrelated and correlated wireless channels. These performance measures can also be obtained in case of non-identical channels for different users. Although it maximizes system throughput, the above opportunistic scheduling could lead to unfair service allocation.

Another important aspect in scheduling is to provide fairness among all mobiles. In the literature, most of the packet-based wireless fair queuing algorithms employs heuristic-based methods to achieve fair bandwidth allocation (at the radio link level) among the mobiles. Therefore, the best fair-queuing allocation may not be achieved under some circumstances. In addition, they are designed primarily for single-rate transmission scenarios, where the number of transmitted packets per time slot is fixed and same for all mobiles. As a result, the implementation of the above fair schedul-

ing algorithms in multi-rate transmission scenarios (where each flow can change the number of transmitted packets per time slot based on its channel condition) might not be straightforward. In this dissertation, two optimization-based fair scheduling algorithms are proposed: one for single-rate transmission environment and another for multi-rate transmission environment.

1.4.2 Multi-hop Wireless Networks

Due to the lack of any central controller, problems in a multi-hop wireless network are generally more challenging than those in a cellular wireless network. In a multi-hop wireless network, each mobile coordinates with each other to keep the network up and running. Scheduling could be difficult, since most algorithms for multi-hop networks are usually in a distributed manner. Error control on the other hand could be conducted in the same way as it is in a cellular wireless network.

For a multi-hop wireless network, this dissertation models and analyzes the impact of a class of ARQ protocols on the performance of in a multi-hop wireless data network. The analysis is divided into two parts. The first part models the number of transmissions for successful delivery of a packet across a multi-hop path. The second part studies the performance of batch transmission in a multi-hop wireless network only with a small number of hops (e.g., two hops). The novelty of these models is that the probability mass function (*pmf*) for the number of transmissions required for end-to-end delivery of a packet or a batch of packets can be obtained under different hop-level error control policies. Therefore, the trade-off between reliability and latency can be analyzed.

1.4.3 Significance of the Results

Scheduling is one of the most important components of radio link design for cellular wireless networks. In the literature, it was shown that a channel-quality-based opportunistic scheduling can be used to maximize resource usage (system throughput). However, the impact of *Adaptive Modulation and Coding (AMC)*, ARQ, and channel variation on this scheduling algorithm was not thoroughly investigated. This dissertation derives complete statistics (i.e., *probability density function (pdf)/probability*

mass function (pmf)) for the delay and throughput of this type of scheduling. The statistics can be used to adjust the network parameters such as the level of *Quality of Service (QoS)*, radio link layer parameter settings, and the number of admissible connections so that the radio link performance can be optimized.

Another type of scheduling, namely, *fair scheduling* aims at allocating resources among customers in proportion to their weights. This type of scheduling algorithm would be useful to classify customers based on their levels of subscription. For example, *premium-class* customers should acquire larger portions of service than *regular-class* customers. This dissertation proposes two fair scheduling algorithms for cellular networks under single-rate and multi-rate transmissions. Both algorithms show improved performance over the algorithms in the literature.

As cellular networks evolve towards the next-generation wireless networks, incorporation of multi-hop communications into the cellular network seems to be inevitable. This dissertation presents two performance analysis models for multi-hop packet transmissions which reveal the trade-off between latency and reliability under different error control (i.e., retransmission) policies. In particular, ARQ with higher level of persistence provides higher reliability at the expenses of increased latency. For real-time traffic, data packets might become useless after some time. Therefore, the use of infinite-persistence ARQ may not be appropriate. The analytical model would be useful for engineering the network for provisioning required QoS for the different types of service in a multi-hop wireless setup.

1.5 Organization of This Dissertation

The organization of this dissertation is as follows:

- Chapter 2 outlines the following mathematical preliminaries: a Markov process and wireless channel models including a *Random State Channel (RSC)*, a *Gilbert-Elliot Channel (GEC)*, and a *Finite State Markov Channel (FSMC)*.
- Chapter 3 presents a framework for analyzing radio link level performance for opportunistic scheduling with automatic repeat request (ARQ)-based error control in multi-rate cellular networks.
- In Chapter 4, a fair scheduling algorithm under single-rate transmission environment–

Optimal Radio Channel Allocation for Single-Rate Transmission (ORCA-SRT)– is proposed.

- Chapter 5 extends the proposed ORCA-SRT to support multi-rate transmission environment. The modified algorithm is called *Optimal Radio Channel Allocation for Multi-Rate Transmission (ORCA-MRT).*
- In Chapter 6, the performance of transmission of a particular packet over a multi-hop wireless network with an arbitrary number of hops is analyzed.
- Chapter 7 studies the performance of a transmission of a batch of packets over a two-hop wireless path.
- Chapter 8 summarizes the main contributions of this dissertation, and discusses few future research directions.

1.6 Notations and Operations

Definitions of main notations and operators in this dissertation are as follows. Regular and boldface letters are used to represent scalar values and matrices, respectively. Matrices $\boldsymbol{e}$, $\mathbf{0}$, and $\mathbf{I}$ denote all-one, all-zero and identity matrices, respectively. The notation $[\mathbf{X}]_{i,j}$ returns entry (i, j) of the matrix $\mathbf{X}$. The operation $(\boldsymbol{x}, \boldsymbol{y})$ concatenates matrix $\boldsymbol{x}$ to the left of matrix $\boldsymbol{y}$.

Chapter 2

Mathematical Background

2.1 Homogeneous Markov Process

A homogeneous Markov process is a time-dependent stochastic process, whose value in the future depends on the value at present [18]. Mathematically, a stochastic process, $X(t)$, is called a homogeneous Markov process if

$$\begin{aligned} &P[X(t_0+t_1) \leqslant x | X(t_0) = x_0, X(\tau), \tau \in (\infty, t_0)] \\ &\quad = P[X(t_0+t_1) \leqslant x | X(t_0) = x_0], \forall t_1 > 0, \end{aligned} \tag{2.1}$$

where $P[A|B]$ is the probability of event A conditioned on event B. From (2.1), the evolution of a Markov process depends only on its present value (at t_0), not its past value. Equivalently, the present value of a Markov process captures all information in the past, and is sufficient to determine the future value.

The value of that a Markov process assumes at any time is usually called a *state*. A set of possible states for a particular Markov process can consist of finite or infinite numbers of of states. A Markov process can also be classified into *Discrete-Time Markov Chain (DTMC)* and *Continuous-Time Markov Chain (CTMC)*. In a DTMC, state changes occur only at a finite number of points in time domain. In a CTMC, on the other hand, the changes can occur at any time. Throughout this dissertation, only DTMCs whose evolution is in a set of finite states are considered.

2.1.1 Transition Probability

In a discrete-time domain, we represent $X(t)$ by the superscript or $X^{(t)}$, and define *transition probability* from state i to state j, p_{ij}, as well as *Transition Probability*

Matrix (TPM), $\mathbf{P}$, as follows:

$$p_{ij} = P[X^{(t)} = j | X^{(t-1)} = i], i, j \in \mathscr{X}, \tag{2.2}$$

$$\mathbf{P} = \begin{pmatrix} p_{00} & p_{01} & \cdots & p_{0j} & \cdots \\ p_{10} & p_{11} & \cdots & p_{1j} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ p_{i0} & p_{i1} & \cdots & p_{ij} & \cdots \\ \vdots & \vdots & \cdots & \vdots & \ddots \end{pmatrix}, \tag{2.3}$$

where $\mathscr{X}$ is the set of possible states of the DTMC

Apart from a TPM, we usually represent states in a DTMC by a row vector. More specifically, a state probability vector at time t is denoted by

$$\mathbf{X}^{(t)} = (X_0^{(t)} \quad X_1^{(t)} \cdots) \tag{2.4}$$

$$= \mathbf{X}^{(t-1)}\mathbf{P} = \mathbf{X}^{(0)}\mathbf{P}^t. \tag{2.5}$$

In a scalar from,

$$X_j^{(t)} = \sum_{\forall i \in \mathscr{X}} X_i^{(t-1)} p_{ij} \tag{2.6}$$

$$= \sum_{\forall i \in \mathscr{X}} X_i^{(0)} p_{ij}^{(t)}, \tag{2.7}$$

where $p_{ij}^{(t)}$ is the probability that the process will move from state i to state j in t steps. Equivalently, in a matrix form,

$$\mathbf{P}^{(t)} = \mathbf{P}^{(t')}\mathbf{P}^{(t-t')} = \mathbf{P}^{t'}\mathbf{P}^{t-t'}. \tag{2.8}$$

In a scalar form, $p_{ij}^{(t)}$ can be calculated from a well-known *Chapman-Kolmogorov* equation:

$$p_{ij}^{(t)} = \sum_{\forall k \in \mathscr{X}} p_{ik}^{(t')} p_{kj}^{(t-t')}. \tag{2.9}$$

2.1.2 Basic Properties of a DTMC

Basic properties of a DTMC include

$$0 \leqslant p_{ij} \leqslant 1, \quad i,j \in \mathscr{X}, \tag{2.10}$$

$$\sum_{\forall j \in \mathscr{X}} p_{ij} = 1, \tag{2.11}$$

$$\sum_{\forall i \in \mathscr{X}} p_{ij} > 0. \tag{2.12}$$

Literally, (2.10) is a general constraint for probability. Eq. (2.11) implies that Markov process always stays in $\mathscr{X}$. The state corresponding to the column which violates (2.12) is called an *ephemeral state* and can be removed from the process.

For an ergodic DTMC[1], there exists time-invariant steady state probability vector ($\boldsymbol{\pi} = (\pi_1, \pi_2, \cdots)$), which can be calculated from (2.13) and (2.14).

$$\begin{aligned} \boldsymbol{\pi} &= \boldsymbol{\pi}\mathbf{P}, \quad \text{or equivalently} \\ \pi_j &= \sum_{\forall i \in \mathscr{X}} \pi_i p_{ij}, \quad \forall j \in \mathscr{X}, \qquad (2.13) \\ \boldsymbol{\pi}\boldsymbol{e} &= 1, \qquad (2.14) \end{aligned}$$

where $\boldsymbol{e}$ is an all-one column vector.

2.2 Absorbing Markov Process

An *absorbing Markov process* is a special type of a non-ergodic Markov process which finally stops at one of the *absorbing states* [19]. Let the first m_0 states in the above absorbing DTMC be *absorbing* states and let states $m_0 + 1$, $m_0 + 2$, $\cdots$, $m_0 + M$ be *transient* states. A general form of the corresponding TPM, $\mathbf{P}$, is given by (2.15) below

$$\mathbf{P} = \left(\begin{array}{c|c} \mathbf{I} & \mathbf{0} \\ \hline \mathbf{R} & \mathbf{Q} \end{array}\right), \tag{2.15}$$

where the matrices $\mathbf{Q}$ and $\mathbf{R}$ are called *transient* and the *absorbing* TPMs, respectively. The probability that the DTMC is absorbed (or finished) at time t ($\boldsymbol{f}_t$), the

[1]A DTMC is ergodic if all states are communicable, aperiodic, and positive recurrent.

absorbing probability vector ($\boldsymbol{f}$), and the expected time to absorption corresponding to each absorbing state ($\boldsymbol{E}[t]$) can be calculated from (2.16)-(2.18) below [20]

$$\boldsymbol{f}_t = \begin{cases} \boldsymbol{\alpha}_0, & t = 0, \\ \boldsymbol{\alpha}\mathbf{Q}^{t-1}\mathbf{R}, & t \geqslant 1, \end{cases} \tag{2.16}$$

$$\boldsymbol{f} = \boldsymbol{\alpha}(\mathbf{I}-\mathbf{Q})^{-1}\mathbf{R}, \tag{2.17}$$

$$\boldsymbol{E}[t] = \boldsymbol{\alpha}(\mathbf{I}-\mathbf{Q})^{-2}\mathbf{R}, \tag{2.18}$$

where $\boldsymbol{\alpha}_0$ and $\boldsymbol{\alpha}$ are the probability vectors representing that the DTMC starts at the absorbing and transient states, respectively (i.e., $\mathbf{X}^{(0)} = (\boldsymbol{\alpha}_0, \boldsymbol{\alpha})$).

For a DTMC with single absorbing state, $\boldsymbol{f}_t$, $\boldsymbol{f}$, and $\boldsymbol{E}[t]$ in (2.16)-(2.18) becomes scalar values. In this case, the time to absorption follows *Phase-Type* (PH) distribution, and the above results can be simplified to

$$F_t = \sum_{i=0}^{t} f_i = \begin{cases} \alpha_0, & t = 0, \\ \alpha_0 + 1 - \boldsymbol{\alpha}\mathbf{Q}^{\mathrm{t}}e, & t \geqslant 1, \end{cases} \tag{2.19}$$

$$E[t] = \boldsymbol{\alpha}(\mathbf{I}-\mathbf{Q})^{-1}e, \tag{2.20}$$

where F_t is *Cumulative Distribution Function (cdf)* of time to absorption. The derivations of (2.19) and (2.20) are given in Appendix A.

2.3 Discrete Models for a Wireless Channel

A discrete channel models a wireless channel by dividing the entire range of SINR into M non-overlapping intervals. Each interval corresponds to a 'channel state' ($\mathscr{M} \in \{1, 2, \cdots, M\}$), which remains unchanged during one time slot (although the SINR may vary within the corresponding interval) [21]-[23].

The evolution of the channel states can be either independent or correlated (in case of fast and slow fading environment) in time domain. The channel state of a time-independent channel can be modeled by a *Random State Channel (RSC)*, while that in a correlated channel can be modeled by a DTMC-based model (either *Gilbert-Elliot Channel (GEC)* or *Finite State Markov Channel (FSMC)*).

When the channel state is m, a maximum of r packets can be transmitted during a time slot such that the average packet error probability is $p_{err}^{(m)}$. Although this

dissertation assumes that $r = m$, other channel-to-rate mapping functions (e.g., those in [23]) can be modeled in a similar manner. Hereafter, we will refer to higher channel states with higher SINR which can accommodate higher transmission rates as *good* or *better* channel states.

2.3.1 Random State Channel (RSC)

The channel state distribution of an RSC is given by

$$\boldsymbol{f}_{\mathscr{M}}^{(t)} = \boldsymbol{\pi}, \forall t, \tag{2.21}$$

where $f_{\mathscr{M}}^{(t)}(m) = \left[\boldsymbol{f}_{\mathscr{M}}^{(t)}\right]_{1,m}$ and $\pi_m = [\boldsymbol{\pi}]_{1,m}$ are the probabilities that the channel state is m in time slot t and at steady state, respectively. Since the wireless channel is uncorrelated in time domain, the distribution is identical for all time slots.

2.3.2 Gilbert-Elliott Channel (GEC)

[24] and [25] proposed a *Gilbert-Elliot Channel (GEC)* or two-state Markov channel model. In GEC, the channel is classified as *bad* ($b \equiv 0$) or *good* ($g \equiv 1$) (i.e., $\mathscr{M} \in \{0, 1\}$), and is assumed to be constant over a time slot. The probabilities that the channel in time slot t is *bad* and *good* are $\left[\boldsymbol{f}_{\mathscr{M}}^{(t)}\right]_{1,1}$ and $\left[\boldsymbol{f}_{\mathscr{M}}^{(t)}\right]_{1,2}$, where

$$\boldsymbol{f}_{\mathscr{M}}^{(t)} = \boldsymbol{f}_{\mathscr{M}}^{(t-1)} \cdot \mathbf{P}. \tag{2.22}$$

Channel states in GEC changes according to the transition probability matrix $\mathbf{P}$ defined below

$$\mathbf{P} = \left(\begin{array}{c|cc} \text{State} & b & g \\ \hline b & v & 1-v \\ g & 1-w & w \end{array}\right), \tag{2.23}$$

where v and w are the probabilities that the channel stays in *bad* and *good* states, respectively. Given that the packet error probabilities when the channel is in the *bad* and *good* states are $p_{err}^{(b)}$ and $p_{err}^{(g)}$, respectively, the steady state packet error probability (p_{err}) can be calculated from (2.24) below

$$p_{err} = \frac{p_{err}^{(b)}(1-v) + p_{err}^{(g)}(1-w)}{2-v-w}. \tag{2.24}$$

Definition 2.1 CHANNEL STATE CORRELATION IN STATE i ($\rho^{(i)}$) *is the probability of a channel to stay in the same state (i) in the next time slot.* □

Definition 2.2 AVERAGE CHANNEL STATE CORRELATION (ρ) *is average value of* $\rho^{(i)}$ *taken over all states.* □

In GEC, $\rho^{(b)} = v$, $\rho^{(g)} = w$, and

$$\rho = \frac{w(1-v) + v(1-w)}{2-v-w}. \tag{2.25}$$

Larger values of ρ imply that the channel states are more correlated, while smaller values signify a more independent channel. Note that, given the average packet error rate p_{err}, the above transition probabilities (v and w) can be obtained using normalized Doppler frequency $f_d\tau$ [26], where f_d is the maximum Doppler shift given by $f_d = \frac{v}{c} \cdot f_c$ (v is the velocity of the mobile, c is the light speed, f_c is the carrier frequency), and τ is the packet length. When $f_d\tau$ is small, the fading process is more correlated, while for higher values of $f_d\tau$, channel fading is more independent.

2.3.3 Finite State Markov Channel (FSMC)

In a typical wireless network, received SINR fluctuates fairly widely. Increasing the number of channel states in a Markov-based channel, each corresponding to a smaller range of SINR, would presumably result in a more accurate channel model.

Each state in an FSMC represents a range of SINR. With the *Fixed Modulation and Coding (FMC)*, transmissions in different states result in different average packet error probability. For a fixed average packet error probability corresponding to each state, *Adaptive Modulation and Coding (AMC)*[2] exploits channel variation, and adjusts the modulation index as well as the strength of the error correcting codes accordingly [21]-[23]. When the channel is in states which correspond to low SINR values, only lower-rate transmission is possible. For example, by setting the SINR threshold in a *Rayleigh* fading wireless channel to $\{6, 10, 14, 18, 21, 24\}$ dB, uncoded transmissions

[2]AMC has been adopted in systems such as 3GPP [27], 3GPP2 [28], IEEE 802.16 [29], IEEE 802.11 [30] and HIPERLAN/2 [31].

using *M-ary Quadrature Amplitude Modulation (M-QAM)*-based adaptive modulation with the modulation index of $\{2, 4, 8, 16, 32, 64\}$ can stabilize the average *Bit Error Rate (BER)* of each channel state to 10^{-3} [32],[33]. Similar examples for a *Nakagami* fading channel can be found in [21],[23],[34]. In this dissertation, we focus only on the FSMC model with AMC.

A FSMC is represented by a DTMC in which transitions only between adjacent states are allowed. A multi-rate transmission scenario, where M different transmission rates are possible, can be modeled by an M-state FSMC. Each state in the Markov chain corresponds to a certain range of SINR at which a particular transmission rate results in constant probability of packet loss.

The approaches for estimating FSMC parameters depend on the underlying physical layer model. Again, FSMC parameter estimation examples for Rayleigh and Nakagami fading channels were provided in [32] and [34], respectively. To free our model from physical layer assumptions, we assume that all the stationary parameters are available.

For multi-rate transmission scenarios, we assume $\mathscr{M} \in \{1(\text{worst}),2,\cdots,\text{M(best)}\}$. The average channel state correlation ($\rho \in [0, 1]$) defined in DEFINITION 2.2 can be calculated as follows:

$$\rho = \sum_{i=1}^{M} \rho^{(i)} \cdot \pi_i = \sum_{i=1}^{M} p_{ii} \cdot \pi_i, \tag{2.26}$$

where p_{ij} is the transition probability from state i to j and π_i is the steady state probability that the channel is in state i.

Observation 2.1 (NATURE OF FSMC): *In an FSMC, if all the steady state probabilities $(\pi_i, \forall_i)$ are specified, the values of $\rho^{(i)}$ cannot be arbitrarily chosen for all values of i.*

PROOF: *A TPM $\mathbf{P}$ in an FSMC is tridiagonal. Therefore, the number of variables in $\mathbf{P}$ is $3M - 2$, where M is the dimension of the matrix. The necessary equality conditions for a DTMC include (2.11) and (2.13). Since (2.11) and (2.13) include M and $M - 1$ linearly independent equations, total number of equations will be 3M-1, if all $\rho^{(i)} = p_{ii}, i = \{1, \cdots, M\}$ are specified. In this case, the number of equations is*

greater than the number of unknown variables, and the solution to this linear system does not exist. ■

Therefore, we fix ρ (instead of $\rho^{(i)}$), and calculate the TPM $\mathbf{P}$ by using the following algorithm:

- **Step 1:** Set $k = 0$ and $\rho_{(k)} = \rho$.
- **Step 2:** Find the solution $(p_{ij}, \forall i, j)$ of

$$min_{0 \leqslant p_{ij} \leqslant 1} \sum_{i=1}^{M} \pi_i \cdot \left(p_{ii} - \rho_{(k)}\right)^2 \tag{2.27}$$

  subject to (2.11) and (2.13).
- **Step 3:** Evaluate

$$\rho_{(k+1)} = \sum_{i=1}^{M} \pi_i \cdot p_{ii}.$$

- **Step 4:** Terminate if $|\rho_{(k+1)} - \rho|$ is sufficiently small. Otherwise, set $k = k + 1$, change the value of $\rho_{(k+1)}$, and go back to step 2.

With the above algorithm, we can choose any value of ρ, and all values of $\rho^{(i)}$ will be very close to the average value.

At steady state, $\boldsymbol{f}_{\mathscr{M}}^{(t_s)} = \boldsymbol{\pi}$, where t_s is the time slot at which the FSMC reaches the steady state. Due to the limiting behavior of an ergodic DTMC, $\boldsymbol{f}_{\mathscr{M}}^{(t_s+T)} = \boldsymbol{\pi} \cdot \mathbf{P}^T = \boldsymbol{\pi}, T \geqslant 1$. In other words, $\boldsymbol{f}_{\mathscr{M}}^{(t)}$ will be time-invariant if we start observing the channel from the point at steady state (i.e., $\boldsymbol{f}_{\mathscr{M}}^{(0)} = \boldsymbol{\pi}$), or if we observe the long-term behavior of the channel in the case that $\boldsymbol{f}_{\mathscr{M}}^{(0)} \neq \boldsymbol{\pi}$. Under these situations, the wireless channel can simply be represented by an RSC, where $\boldsymbol{f}_{\mathscr{M}}^{(t)} = \boldsymbol{\pi}, \forall t$.

Chapter 3

Channel-Quality-Based Opportunistic Scheduling

In a multi-user cellular wireless network, limited available radio resources must be allocated among all mobiles in the most effective manner. Opportunistic scheduling is a scheduling algorithm which exploits the time-varying nature of a wireless channel. A class of opportunistic scheduling which allows only one-by-one transmission rather than simultaneous transmissions can provide high average network throughput in a wireless network by exploiting the gain due to multi-user diversity [6], which depends on the asynchronous channel variations among mobile users. This type of opportunistic scheduling was shown to maximize network capacity when the network is not limited by available rate set and/or transmission power [35]. However, when the available rate set is finite and/or the transmission power is limited, multiple simultaneous transmissions (e.g., in a CDMA system) may maximize the capacity (i.e., *frame fill efficiency* [36]).

That the opportunistic scheduling can maximize a wireless system performance stochastically even under certain resource allocation fairness constraint, was proven in [37] using the notion of 'utility'. Simulation-based forward link data throughput performance in the Qualcomm *Code Division Multiple Access with High Data Rate (CDMA-HDR)* system [5], which uses an opportunistic scheduling scheme based on the *Proportional Fairness (PF)* criterion, was presented in [38]. The PF algorithm was designed to share the wireless channel resources fairly as well as to maximize channel throughput for best-effort data services. Scheduling mechanisms such as *exponential (EXP) rule* [39] and *modified largest weighted delay first (M-LWDF)* [40] were proposed for quality of service (QoS)-sensitive data service. While most of the

work on opportunistic scheduling aimed at enhancing the scheduling algorithm in different ways [41]-[44] (and the references therein), little attention has been paid on modeling and analyzing the basic scheduling mechanism under different channel dynamics and its impact on overall radio link level performance.

Again, to effectively utilize the scarce radio bandwidth, *Adaptive Modulation and Coding (AMC)* can be used to adjust the modulation index as well as the strength of the error correcting codes according to the current value of SINR at the receiver [21]-[23]. Since in general, AMC is not designed for absolute integrity, reliability can be provided at the radio link level by using ARQ-based error control mechanism which invokes a retransmission procedure in case of transmission failure.

In practice, choices of AMC adjustment correspond to a set of non-continuous transmission rates. In most AMC implementations, the wireless channel is modeled by an FSMC defined in Section 2.3.3. For each interval, an AMC is selected to satisfy the target average packet error probability constraint. For a single user system (i.e., without scheduling), the effects of traffic source on AMC-based system parameters were studied in [22] and [23]. The effects of different ARQ policies on radio link level buffer management were investigated in [21] and [45] considering FSMC and GEC models, respectively. However, the problem of scheduling was not addressed in these works.

This chapter presents a novel analytical framework to evaluate radio link level performance. The framework incorporates channel-quality-based opportunistic scheduling mechanism (and also round-robin scheduling), ARQ-based error recovery, and multi-rate transmission under both correlated and time-independent wireless channels. We determine the *probability mass function (pmf)* of performance metrics including system and per-mobile throughput, inter-access delay, inter-success delay, and connection reset delay. We present simulation results which validate the numerical results obtained from the analytical model. We investigate impacts of system and channel parameters on the different radio link level performance measures thoroughly.

The rest of the chapter is organized as follows. Section 3.1 provides a summary of the system model and the underlying assumptions. The analytical framework is presented in Section 3.2-3.4. The numerical and the simulation results as well as their useful implications are presented in Section 3.5. Finally, chapter summary is given in

Section 3.6.

3.1 System Model and Assumptions

3.1.1 System Description

Consider transmission in a cellular wireless network with a fixed size time slot. With a channel-quality-based opportunistic scheduling principle, the base station transmits/receives data to/from a mobile perceiving the best channel condition. We assume perfect channel state information at the base station, which might be achieved by a training-based channel estimation [46]. We also assume a continuously backlogged data flow corresponding to each mobile.

3.1.2 Wireless Channel Models

In this chapter, we consider both RSC and FSMC models defined in chapter 2. In presence of n mobiles, we study three following cases:

- **Case I** (*All-RSC*): Channel state for each mobile varies according to $\boldsymbol{\pi}$,
- **Case II** (*All-FSMC*): Channel state for each mobile follows the FSMC model. For these mobiles, we specify $\boldsymbol{\pi}$ as well as ρ, and use algorithms in Section 2.3.3 to calculate the TPM.
- **Case III** (*FSMC-RSC*): Channel states for n_r and n_f mobiles follow the RSC and the FSMC model, respectively.

3.1.3 Opportunistic Scheduling Policy and Automatic Repeat reQuest (ARQ) Mechanism

In each time slot, there could be k *eligible* mobiles whose channel states are the best among those of all n mobiles. Since we are considering a TDMA-based system, a channel *access opportunity* is given to only one eligible mobile. Among all eligible mobiles, the base station randomly selects one of them for transmission in the current time slot. Therefore, the transmission probability for each mobile in the eligible set is $1/k$.

To combat error probability inherent with each channel state, ARQ-based error recovery with limited persistence is employed to retransmit erroneous packets. For each mobile, a retransmission counter is maintained to keep track of the number of time slots in which all the transmissions have failed. When all the transmitted packets during a time slot are lost, the retransmission counter is incremented, and when the counter exceeds a certain limit K, the corresponding connection is reset. The counter is reset to zero when the connection is initiated or reset or when at least one of the transmitted packets is received successfully. By adopting this ARQ mechanism, a connection tends to be reset only when the mobile is turned off or experiences extremely *bad* channel condition. In both the cases, data packets in the buffer are subject to extremely long delay and might be discarded during a connection reset process[1].

3.1.4 Analytical Methodology

We analyze the performance of an opportunistic scheduling under the three above channel assumptions: *All-RSC*, *All-FSMC*, and *FSMC-RSC*. For each case, we divide the analysis into two parts. The first part assumes error-free wireless channels, and derives statistics for the following performance parameters:

Definition 3.1 SYSTEM THROUGHPUT (γ_{sys}) *is the number of packets successfully transmitted during a time slot.* □

Definition 3.2 INTER-ACCESS DELAY ($\mathscr{D}_{acc}$) *is the number of time slots between two channel access opportunities corresponding to the same mobile.* □

Definition 3.3 PER-MOBILE THROUGHPUT (γ_{mob}) *is the number of packets successfully transmitted to/from a particular mobile per time slot.* □

In the second part, we introduce non-zero packet error probability to the wireless channel, and use an ARQ mechanism for error recovery at the radio link level. In such a case, at least one packet might be successfully transmitted (sc), or the connection will be reset (rst) after some time due to limited persistence of the ARQ mechanism.

[1] A similar approach is used in IEEE 802.11 [31], where the buffer is flushed after seven transmission failures.

For a particular mobile, we measure the conditional delay to the occurrence of either sc or rst as follows:

Definition 3.4 INTER-SUCCESS DELAY *($\mathscr{D}_{sc}$) is the number of time slots between the points where the retransmission counter is zero and the point where at least one packet is successfully transmitted, given that sc will occur before rst.* □

Definition 3.5 CONNECTION RESET DELAY *($\mathscr{D}_{rst}$) is the number of time slots between the points where the retransmission counter is zero and the point where the connection is reset, given that rst will occur before sc.* □

3.2 Mathematical Model for Case I: All-RSC

In this section, we assume that channel states of all mobiles are independent and identically distributed (i.i.d.) and are modeled by an RSC model. The results for non-identical channels can also be obtained by the modification suggested in Appendix B.

Proposition 3.1 *For* **Case I**, *each mobile is scheduled for transmission at rate m in time slot t with probability $p_{tx}^{(t)}(m)$ (in (3.1)), where $F_m = \sum_{i=1}^{m} \pi_i$.*

$$p_{tx}^{(t)}(m) = \frac{(F_m)^n - (F_{m-1})^n}{n}. \tag{3.1}$$

PROOF: *see Appendix B.* ■

When identical to each other, each mobile has the same channel access probability $\sum_{m=1}^{M} p_{tx}^{(t)}(m) = 1/n$. Therefore, the probability that a mobile is not scheduled for transmission is $1 - 1/n$. Since the evolution of an RSC is a memoryless process, $p_{tx}^{(t)}(m) = p_{tx}(m), \forall t$ (i.e., time-invariant).

3.2.1 Error-Free Wireless Channel

We assume that $p_{err}^{(m)} = 0$ $(\forall m)$ and derive statistics for γ_{sys}, $\mathscr{D}_{acc}$, and γ_{mob}.

Theorem 3.1 *For* **Case I**, *the pmf of system throughput can be calculated from $f_{\gamma_{sys}}(m)$. The joint pmf that a particular mobile acquires channel access at time slot*

d, and perceives channel state m in that time slot can be calculated from $f_{\mathscr{M},\mathscr{D}_{acc}}(m,d)$, *where*

$$f_{\gamma_{sys}}(m) = (F_m)^n - (F_{m-1})^n\,,\ f_{\mathscr{M},\mathscr{D}_{acc}}(m,d) = \frac{f_{\gamma_{sys}}(m)(n-1)^{d-1}}{n^d}, \tag{3.2}$$

and n is the total number of mobiles. □

PROOF: *The probability that the best channel state among all mobiles (and therefore system throughput) is m can be calculated from (3.3) below*

$$\begin{aligned} f_{\gamma_{sys}}(m) &= Pr\{(\exists i : \mathscr{M}_i^{(t)} = m)\ \&\ (\mathscr{M}_i^{(t)} < m+1, \forall i)\} \\ &= \left(1-(1-f_{\mathscr{M}}^{(t)}(m|m))^n\right)\cdot\left(F_{\mathscr{M}}^{(t)}(m)\right)^n, \end{aligned} \tag{3.3}$$

where $\mathscr{M}_i^{(t)}$ *is the channel state of mobile i in time slot t,* $f_{\mathscr{M}}^{(t)}(m|m) = f_{\mathscr{M}}^{(t)}(m)/F_{\mathscr{M}}^{(t)}(m)$ *and* $F_{\mathscr{M}}^{(t)}(m) = \sum_{i=1}^{m} f_{\mathscr{M}}^{(t)}(i)$. *The probability* $f_{\mathscr{M},\mathscr{D}_{acc}}(m,d)$ *that the mobile is not selected for first* $d-1$ *time slots and is selected in time slot d can be calculated in (3.4), where* $p_{tx}^{(t)} = \sum_{\forall m} p_{tx}^{(t)}(m)$ *and* $p_{tx}^{(t)}(m)$ *can be calculated using (3.1).*

$$f_{\mathscr{M},\mathscr{D}_{acc}}(m,d) = p_{tx}^{(d)}(m)\prod_{t=1}^{d-1}\left(1-p_{tx}^{(t)}\right). \tag{3.4}$$

■

Corollary 3.1 *The statistics of system throughput* (γ_{sys}), *inter-access delay* ($\mathscr{D}_{acc}$), *and per-mobile throughput* (γ_{mob}) *for* **Case I** *can be calculated from (3.5), where* $E[\cdot]$ *is an expectation function.*

$$\begin{aligned} E[\gamma_{sys}] &= M - \sum_{m=1}^{M-1}(F_m)^n\,, \qquad E[\mathscr{D}_{acc}] = n, \\ f_{\mathscr{D}_{acc}}(d) &= \frac{(n-1)^{d-1}}{n^d}, \qquad E[\gamma_{mob}] = \tfrac{E[\gamma_{sys}]}{E[\mathscr{D}_{acc}]}. \end{aligned} \tag{3.5}$$

PROOF: *see Appendix C.* ■

When identical to each other, each mobile has to wait for n time slots for another channel access opportunity, which leads to a harmonic reduction (i.e., at the rate of $1/n$) in per-mobile throughput.

3.2.2 Error-Prone Wireless Channel and ARQ Mechanism

For non-zero packet error probability ($p_{err}^{(m)} = p_{err}, \forall m$), the pmf ($f_{\gamma_{sys}^{err}}(s)$) and the expected value ($E[\gamma_{sys}^{err}]$) of system throughput can be calculated from (3.6).

$$\begin{aligned} f_{\gamma_{sys}^{err}}(s) &= \sum_{m=1}^{M} p(s|m) \cdot f_{\gamma_{sys}}(m), \\ E[\gamma_{sys}^{err}] &= \sum_{s=1}^{M} s \cdot f_{\gamma_{sys}^{err}}(s), \\ p(s|m) &= \binom{m}{s} \cdot (1-p_{err})^{s} \cdot (p_{err})^{m-s}. \end{aligned} \tag{3.6}$$

We model the opportunistic scheduling with ARQ by an absorbing DTMC $\mathscr{X}^{(t)}$. At time slot t, $\mathscr{X}^{(t)} \in \{tx, rst, sc\}$ (Figure 3.1) represents transmitting (tx), connection reset (rst), or successful transmission states (sc). The states tx and sc are divided into sub-states tx_i and sc_j representing the retransmission counter ($i \in \{0, \cdots, K\}$) and the number of successfully received packets ($j \in \{1, \cdots, M\}$). We set all the sub-states $tx_i(\forall i)$ to be transient states and all other states to be absorbing states. In effect, there are $M+1$ absorbing states: rst and sc_s ($s \in \{1, \cdots, M\}$) corresponding to connection reset and successful transmission of s packets, respectively.

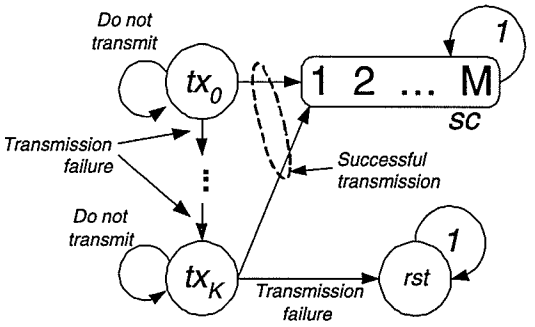


Figure 3.1. *A DTMC representing retransmission process for* **Case I** *with ARQ.*

The above process always starts with the retransmission counter set to zero in the sub-state tx_0, which implies three possibilities: connection initiation, connection reset, and successful transmission. At each transition (time slot), the mobile is granted and not granted a channel access with probabilities $1/n$ and $1-1/n$, respectively. The process finishes in state sc_s with probability $q_s = f_{\gamma_{sys}^{err}}(s)/n$, where the mobile acquires

a channel access (with probability $1/n$) and s data packets are successfully transmitted (with probability $f_{\gamma_{sys}^{err}}(s)$). With probability q_0, the mobile acquires a channel access but no packet is successfully transmitted. In this case, the retransmission counter will be incremented. If the transmission fails when $\mathscr{X}^{(t)} = tx_K$, the process will finish in state rst where the connection is reset. Since the process always starts from tx_0, we set the initial probability vector to $(1, \mathbf{0})$ and obtain the TPM ($\mathbf{W}$) as given by (3.7), where $\Omega_{ij} = [\mathbf{\Omega}]_{ij}$ and $\omega_{ij} = [\boldsymbol{\omega}]_{ij}$.

$$
\begin{aligned}
\mathbf{W} &= \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \boldsymbol{\omega} & \mathbf{\Omega} \end{pmatrix}, \\
\Omega_{ij} &= \begin{cases} 1 - \frac{1}{n}, & \forall i = j, \\ q_0, & j = i+1, i = \{1, \cdots, K\}, \\ 0, & otherwise, \end{cases} \\
\omega_{ij} &= \begin{cases} \overline{q_{j-1}}, & j = \{2, \cdots, M+1\}, \forall i, \\ q_0, & (i,j) = (K+1, 1), \\ 0, & otherwise. \end{cases}
\end{aligned} \tag{3.7}
$$

Since $\mathbf{\Omega}$ is digaonal dominant, a closed-form solution can also be obtained from THEOREM 3.2 below.

Theorem 3.2 *The probability that the DTMC process representing* **Case I** *with ARQ is absorbed to state sc_s at time slot d ($f_{\mathscr{S},\mathscr{D}}(s,d)$), the absorbing probability to state sc_s ($f_{\mathscr{S}}(s)$), and the expected time to absorption to state sc_s ($E[\mathscr{D}; s]$) can be calculated by using (3.8)-(3.10).*

$$
f_{\mathscr{S},\mathscr{D}}(s,d) = \begin{cases} \binom{d-1}{K} \left(\frac{n-1}{n}\right)^{d-K-1} (q_0)^{K+1}, s = 0, \\ q_s \sum_{j=1}^{K+1} \binom{d-1}{j-1} \left(\frac{n-1}{n}\right)^{d-j} (q_0)^{j-1}, s \geqslant 1. \end{cases} \tag{3.8}
$$

$$
f_{\mathscr{S}}(s) = \begin{cases} f_{\gamma_{sys}^{err}}(0)^{K+1}, & s = 0, \\ f_{\gamma_{sys}^{err}}(s) \left(\frac{1 - f_{\gamma_{sys}^{err}}(0)^{K+1}}{1 - f_{\gamma_{sys}^{err}}(0)}\right), & s \geqslant 1. \end{cases} \tag{3.9}
$$

$$
E[\mathscr{D}; s] = \begin{cases} n(K+1) f_{\gamma_{sys}^{err}}(0)^{K+1}, & s = 0, \\ \left(1 - f_{\gamma_{sys}^{err}}(0)^{K+1}(2 + K - (K+1) f_{\gamma_{sys}^{err}}(0))\right) \\ \quad \cdot \left(\frac{n f_{\gamma_{sys}^{err}}(s)}{(1 - f_{\gamma_{sys}^{err}}(0))^2}\right), & s \geqslant 1. \end{cases} \tag{3.10}
$$

□

PROOF: *See Appendix E.* ∎

Corollary 3.2 *For* **Case I** *with ARQ, the connection reset (p_{rst}) and successful transmission probability (p_{sc}) can be calculated from (3.11) and (3.12), respectively. Also, the conditional* pmf *and the expected values of connection reset and successful transmission delay can be calculated from (3.13)-(3.16), respectively.*

$$p_{rst} = f_{\mathscr{S}}(0) = f_{\gamma_{sys}^{err}}(0)^{K+1}. \tag{3.11}$$

$$p_{sc} = 1 - p_{rst}. \tag{3.12}$$

$$f_{\mathscr{D}_{rst}}(d) = \frac{f_{\mathscr{S},\mathscr{D}}(0,d)}{p_{rst}} = \binom{d-1}{K} \frac{(n-1)^{d-K-1}}{n^d}. \tag{3.13}$$

$$E[\mathscr{D}_{rst}] = \frac{E[\mathscr{D};0]}{p_{rst}} = n(K+1). \tag{3.14}$$

$$f_{\mathscr{D}_{sc}}(d) = \frac{\sum_{s=1}^{C} f_{\mathscr{S},\mathscr{D}}(s,d)}{p_{sc}}. \tag{3.15}$$

$$E[\mathscr{D}_{sc}] = \frac{\sum_{s=1}^{C} E[\mathscr{D};s]}{p_{sc}}. \tag{3.16}$$

PROOF: *Eqs. (3.11)-(3.16) can be obtained simply by applying* total probability theorem *and* Bayes' theorem *to (3.8)-(3.10).* ∎

Having obtained results for *Case I*, we now discuss some insightful implications. First, knowing that either rst or sc will occur, we define n_{rst} and n_{sc} as the number of access opportunities until the occurrence of rst and sc, respectively. Again, for rst to occur, the transmission must fail for $K+1$ consecutive access opportunities. In other words,

$$n_{rst} = \frac{E[\mathscr{D}_{rst}]}{E[\mathscr{D}_{acc}]} = K+1. \tag{3.17}$$

Similarly,

$$n_{sc} = \frac{E[\mathscr{D}_{sc}]}{E[\mathscr{D}_{acc}]} = \frac{1 - f_{\gamma_{sys}^{err}}(0)^{K+1}(2+K-(K+1)f_{\gamma_{sys}^{err}}(0))}{(1-f_{\gamma_{sys}^{err}}(0)^{K+1}) \cdot (1-f_{\gamma_{sys}^{err}}(0))}, \tag{3.18}$$

where the expected inter-access delay ($E[\mathscr{D}_{acc}] = n$) can be calculated from (3.5).

Secondly, from (3.10)-(3.18), we can obtain unconditional expected delay ($E[\mathscr{D}_{fin}]$) and number of access opportunities (n_{fin}) until the process finishes (either in rst or

sc) by using the *total probability theorem* as follows:

$$\begin{aligned} E[\mathcal{D}_{fin}] &= \sum_{s=0}^{M} E[\mathcal{D}; s] \\ &= n\left[(K+1)f_{\gamma_{sys}^{err}}(0)^{K+1} + \left(\sum_{s=1}^{M} f_{\gamma_{sys}^{err}}(s)\right) \cdot \right. \\ &\qquad \left. \left(\frac{1 - f_{\gamma_{sys}^{err}}(0)^{K+1}(2+K-(K+1)f_{\gamma_{sys}^{err}}(0))}{(1-f_{\gamma_{sys}^{err}}(0))^2}\right)\right] \\ &= n\left[(K+1)f_{\gamma_{sys}^{err}}(0)^{K+1} + \left(1 - f_{\gamma_{sys}^{err}}(0)\right) \cdot \right. \\ &\qquad \left. \left(\frac{1 - f_{\gamma_{sys}^{err}}(0)^{K+1}(2+K-(K+1)f_{\gamma_{sys}^{err}}(0))}{(1-f_{\gamma_{sys}^{err}}(0))^2}\right)\right] \\ &= n \cdot \frac{1 - f_{\gamma_{sys}^{err}}(0)^{K+1}}{1 - f_{\gamma_{sys}^{err}}(0)}. \end{aligned} \tag{3.19}$$

$$n_{fin} = \frac{E[\mathscr{D}_{fin}]}{E[\mathscr{D}_{acc}]}. \tag{3.20}$$

From (3.19) we can obtain $E[\mathscr{D}_{acc}] = n$ by setting K in (3.19) to zero (i.e., no retransmission).

Thirdly, conditioned on the occurrence of *sc*, the per-mobile throughput, which is the number of packets successfully transmitted by a mobile per unit time, can be calculated as follows:

$$E[\gamma_{mob}^{sc}] = \frac{E[\gamma_{sys}^{err}]}{E[\mathscr{D}_{sc}]}. \tag{3.21}$$

We can also calculate two more useful metrics: the number of successfully transmitted packets before connection reset and the delay until the connection is reset. Let a *transmission cycle* be an interval between the occurrence of either rst or sc and the next occurrence of either rst or sc. Let $\mathscr{Y}_i \in \{sc, rst\}$ be the state at the end of transmission cycle i. Then, the probability that the process will be in state rst for the first time at the end of i cycle is $p_{sc}^{i-1}(1-p_{sc})$, and the corresponding expected number of cycles is $E[k_{\mathscr{Y}}] = 1/(1-p_{sc})$. Also, the average number of successfully transmitted packets in each access opportunity is $E[\gamma_{sys}^{err}]$. Therefore, the number of successfully transmitted packets before connection reset and the delay until the connection is reset can be calculated from $E[\gamma_{sys}^{err}] \cdot E[k_{\mathscr{Y}}] = E[\gamma_{sys}^{err}]/(1-p_{sc})$ and $E[\mathscr{D}_{sc}] \cdot E[k_{\mathscr{Y}}] =$

$E[\mathscr{D}_{sc}]/(1-p_{sc})$, respectively. Furthermore, the expected per-mobile throughput until the connection is reset can be calculated from $(E[\gamma_{sys}^{err}] \cdot E[k_{\mathscr{D}}])/(E[\mathscr{D}_{sc}] \cdot E[k_{\mathscr{D}}])$ which is $E[\gamma_{mob}^{sc}]$ in (3.21). Therefore, $E[\gamma_{mob}^{sc}]$ represents both the number of packets successfully transmitted by a mobile per unit time and the average number of packets transmitted by a mobile until the connection is reset.

3.3 Mathematical Model for Case II: All-FSMC

In this section, we assume that channel states of all n mobiles follow the FSMC model. Denoted by $\boldsymbol{f}_{\mathscr{M}_i}^{(0)}$ and $\mathbf{P}_i$ are the initial probability vector and TPM representing the channel of mobile i. When the initial state is m_i, $\boldsymbol{f}_{\mathscr{M}_i}^{(0)} = \boldsymbol{e}_{m_i}$, where $\boldsymbol{e}_{m_i}$ is a row vector whose m_i^{th} entry is one and all other entries are zero.

3.3.1 Error-Free Wireless Channel

Again, we start the analysis with $p_{err} = 0$ and later extend the results for the case with non-zero packet error probability and with ARQ. In the absence of transmission error, the process consists of two steps: channel variation and scheduling. Since the channel model in this case exhibits time correlation, the events corresponding to each mobile (i.e., granted channel access or not) in two successive time slots are not independent. Therefore, the joint *pmf* $f_{\mathscr{M},\mathscr{D}_{acc}}(m,d)$ cannot be factored as in (3.4).

We use an n-dimensional DTMC $\mathscr{M}^{(t)} = (\mathscr{M}_1^{(t)}\mathscr{M}_2^{(t)}\cdots\mathscr{M}_n^{(t)})$ to keep track of channel states of all n mobiles, where $\mathscr{M}_i^{(t)} \in \{1,\cdots,M\}$ is the channel state of mobile i at time slot t. Correspondingly, the channel state probability vector in time slot t and TPM of all n mobiles can be calculated from $\boldsymbol{f}_{\mathscr{M}}^{(t)} = \boldsymbol{f}_{\mathscr{M}_1}^{(t)} \otimes \cdots \otimes \boldsymbol{f}_{\mathscr{M}_n}^{(t)}$ and $\mathbf{P} = \mathbf{P}_1 \otimes \cdots \otimes \mathbf{P}_n$, respectively, where $\otimes$ denotes the Kronecker product.

Due to the Kronecker product, the resulting states in the DTMC are arranged such that $\mathscr{M}_i^{(t)}$ increases in the reverse order of i. For example, with $n = 3$ and $M = 2$, $\mathscr{M}^{(t)} \in \{(111),(112),(121),(122),(211),\cdots,(222)\}$. Hereafter, we will refer to each entry in the matrices involving the FSMC process, by using an n-digit label $\boldsymbol{m} = m_1 m_2 \cdots m_n$, where the i^{th} digit is the channel state of mobile i.

Having obtained the model for channel variation, we now incorporate the scheduling mechanism into the model. Due to time-correlation, we model the transmission

process by using an absorbing DTMC $(\mathscr{X}^{(t)}, \mathscr{M}^{(t)})$, where $\mathscr{X}^{(t)} \in \{wait, tx\}$ and $\mathscr{M}^{(t)}$ represent the transmission state of the mobile 1 and channel variation of all mobiles, respectively[2]. In the following analysis, we drop $\mathscr{M}^{(t)}$ inherited in each $\mathscr{X}^{(t)}$ for the sake of explanation. At time slot t, $\mathscr{X}^{(t)} = wait$ and $\mathscr{X}^{(t)} = tx$ imply that the mobile is waiting for and is granted a channel access opportunity, respectively.

Let $\mathbf{G}$ and $\mathbf{G}'$ be diagonal matrices with the same dimension as $\mathbf{P}$. Conditioned on the channel states $\boldsymbol{m} = (m_1 \cdots m_n)$, the diagonal entries ($g(\boldsymbol{m})$ and $g'(\boldsymbol{m})$) of $\mathbf{G}$ and $\mathbf{G}'$ represent the probabilities that mobile 1 will be and will not be granted channel access opportunity, respectively, and can be calculated from

$$g(\boldsymbol{m}) = \begin{cases} 0, & \exists i > 1 : m_i > m_1, \\ \frac{1}{mult_{\boldsymbol{m}}(m_1)}, & m_i \leqslant m_1 \forall i, \end{cases} \tag{3.22}$$

$$g'(\boldsymbol{m}) = 1 - g(\boldsymbol{m}), \tag{3.23}$$

where the multiplicity of λ in $\boldsymbol{m}$ ($mult_{\boldsymbol{m}}(\lambda)$) is the number of digits in $\boldsymbol{m}$ which are equal to λ. If mobile 1 is not eligible[3], it will not obtain a channel access. If the mobile 1 is eligible, on the other hand, it will and will not acquire channel access with probability $g(\boldsymbol{m}) = 1/mult_{\boldsymbol{m}}(m_1)$ and $1 - g(\boldsymbol{m})$, respectively, where $mult_{\boldsymbol{m}}(m_1)$ is the number of eligible mobiles. Given $\boldsymbol{f}_{\mathscr{M}}^{(t-1)}$, the probability that mobile 1 will be and will not be scheduled for transmission at time slot t, is given by $\boldsymbol{f}_{\mathscr{M}}^{(t-1)}\mathbf{PG}$ and $\boldsymbol{f}_{\mathscr{M}}^{(t-1)}\mathbf{PG}'$, respectively.

We assume that mobile 1 acquires a channel access at time slot 0, reset $\mathscr{X}^{(0)} = wait$, and set every state with $\mathscr{X}^{(t)} = tx$ as the absorbing state. Therefore, the TPM $\mathbf{W}$ for the $(\mathscr{X}^{(t)}, \mathscr{M}^{(t)})$ process can be expressed by (3.24) below

$$\mathbf{W} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \boldsymbol{\omega} & \boldsymbol{\Omega} \end{pmatrix} = \begin{pmatrix} \mathbf{PG}' & \mathbf{PG} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}. \tag{3.24}$$

Since $(\mathscr{X}^{(t)}, \mathscr{M}^{(t)})$ always starts from $\mathscr{X}^{(0)} = wait$, the initial probability vector is $\boldsymbol{f}_{\mathscr{M}}^{(0)}$.

[2]Without loss of generality, we calculate statistics of mobile 1 in presence of $n-1$ mobiles.

[3]In a time slot, the channel state of an eligible mobile is the best among those of all mobiles.

Theorem 3.3 *For* **Case II** *without ARQ, if mobile 1 is granted a channel access opportunity at time slot 0,*

- *the probability that mobile 1 will be granted the next opportunity at time slot d and perceive channel state m is*

$$f_{\mathscr{M},\mathscr{D}_{acc}}(m,d|\boldsymbol{f}_{\mathscr{M}}^{(0)}) = \sum_{\forall m_1=m} \left(\boldsymbol{f}_{\mathscr{M}}^{(0)}\boldsymbol{\Omega}^{d-1}\boldsymbol{\omega}\right), \tag{3.25}$$

- *the probability that mobile 1 will perceive channel state m in the next access opportunity is*

$$f_{\mathscr{M}}(m|\boldsymbol{f}_{\mathscr{M}}^{(0)}) = \sum_{\forall m_1=m} \left(\boldsymbol{f}_{\mathscr{M}}^{(0)}(\mathbf{I}-\boldsymbol{\Omega})^{-1}\boldsymbol{\omega}\right), \tag{3.26}$$

- *the expected inter-access delay conditioned on* $\boldsymbol{f}_{\mathscr{M}}^{(0)}$ *is*

$$E[\mathscr{D}_{acc}|\boldsymbol{f}_{\mathscr{M}}^{(0)}] = \boldsymbol{f}_{\mathscr{M}}(\mathbf{I}-\boldsymbol{\Omega})^{-1}\mathbf{e}, \tag{3.27}$$

where $\mathbf{e}$ *is an all-one column vector.* □

PROOF: *Since the time to absorption of this DTMC is equivalent to the number of time slots mobile 1 has to wait for its next channel access opportunity, (3.25)-(3.27) can be obtained from (2.16)-(2.18). The absorbing state space for the above DTMC consists of several channel states. Therefore, the possibilities attributed to all the states are incorporated through the summation.* ■

Corollary 3.3 *For* **Case II**,

$$\begin{aligned} E[\gamma_{sys}] &= M - \sum_{m=1}^{M-1} (F_m)^n . \\ E[\mathscr{D}_{acc}|\boldsymbol{f}_{\mathscr{M}}^{(0)}] &= \boldsymbol{f}_{\mathscr{M}}^{(0)}(\mathbf{I}-\boldsymbol{\Omega})^{-1}\mathbf{e}. \\ E[\gamma_{mob}] &= \frac{E[\gamma_{sys}]}{E[\mathscr{D}_{acc}]}. \end{aligned} \tag{3.28}$$

PROOF: *Since the throughput is defined at the steady state, the result is the same as in (3.5). The result for* $E[D_{acc}|\boldsymbol{f}_{\mathscr{M}}^{(0)}]$ *follows directly from* THEOREM *3.3.* ■

3.3.2 Error-Prone Wireless Channel and ARQ Mechanism

We use an approach similar to that in Section 3.2.2 to model *Case II* with ARQ. Again, when channel state is m, s packets will be successfully transmitted with probability $p(s|m)$ in (3.6). We define the two following matrices ($\mathbf{Q}$ and $\mathbf{Q}'$) which will be used to model *Case II* with ARQ. Matrix $\mathbf{Q}$ maps channel states during an access opportunity to the number of successfully transmitted packets. The entry in row m and column s of $\mathbf{Q}$ are submatrices $p(s|m) \cdot \boldsymbol{e}$, where $\boldsymbol{e}$ is an all-one column vector with size M^{n-1}. Matrix $\mathbf{Q}'$ is a diagonal matrix whose m^{th} entry is $p(0|m) \cdot \mathbf{I}$, where $\mathbf{I}$ is an identity matrix with size M^{n-1}.

By allowing only K consecutive transmission opportunities without any successful packet delivery, the TPM ($\mathbf{W}$) for *Case II* with ARQ is formulated as (3.29) below

$$
\begin{aligned}
\mathbf{W} &= \begin{pmatrix} \boldsymbol{\Omega} & \boldsymbol{\omega} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}, \\
\boldsymbol{\Omega}_{ij} &= \begin{cases} \mathbf{PG}' & ,i = j, \\ \mathbf{PGQ}' & ,j = i+1, i = \{1, \cdots, K\}, \\ \mathbf{0} & ,otherwise, \end{cases} \\
\boldsymbol{\omega}_{i1} &= \begin{cases} [\mathbf{0}, \mathbf{PGQ}] & ,i = \{1, \cdots, K\}, \\ \mathbf{PG}[\mathbf{q}_0, \mathbf{Q}] & ,i = K+1, \end{cases}
\end{aligned}
\tag{3.29}
$$

where $\boldsymbol{\Omega}_{ij}$ and $\boldsymbol{\omega}_{i1}$ are the sub-matrices in row i and column j of $\boldsymbol{\Omega}$ and $\boldsymbol{\omega}$, and $\mathbf{q}_0 = \mathbf{Q}' \cdot e$.

Compared to that in Section 3.3.1, the model in this section divides a channel access state (tx) based on the transmission result. The probability that no packet is transmitted successfully and that the ARQ increments the retransmission counter is $\mathbf{PGQ}'$. If the transmission fails when the counter is K, the process will finish with probability $\mathbf{PGq}_0$. On the other hand, the process will finish with s successfully transmitted packets with probability located in column s of $\mathbf{PGQ}$.

The above process always starts with retransmission counter set to zero and finishes when at least one packet is successfully transmitted or when the counter exceeds K. Therefore, we set the initial probability vector to $(\boldsymbol{f}_{\mathscr{M}}^{(0)}, \mathbf{0})$. The row vector $\mathbf{0}$ appended to $\boldsymbol{f}_{\mathscr{M}}^{(0)}$ represents that the process must start with the retransmission counter set to zero. Connection reset and inter-success delay correspond to the time that

the process requires to be absorbed to a state with zero and at least one successfully transmitted packet, respectively. We observe that $\mathbf{W}$ has the same form as in (3.24). Therefore, the results in this case are the same as in Theorem 3.3. By replacing the above results into COROLLARY 3.2, we can obtain the performance results in terms of $\mathscr{D}_{sc}$ and $\mathscr{D}_{rst}$.

3.4 Mathematical Model for Case III: FSMC-RSC

In this section, we assume that channel states for n_f and n_r mobiles follow the FSMC and RSC models, respectively[4]. Again, the channel states of mobile i are characterized by steady state probability vector $\boldsymbol{\pi}_i$ and by TPM $\mathbf{P}_i$ as well as initial probability vector $\boldsymbol{f}_{\mathscr{M}_i}^{(0)}$ in case of RSC and FSMC, respectively. Since an RSC model is equivalent to an FSMC when $\boldsymbol{f}_{\mathscr{M}_i}^{(0)} = \boldsymbol{\pi}_i$ (see discussion in Section 2.3.3), the performance results in this section can be obtained by using the model in Section 3.3 and setting $\boldsymbol{f}_{\mathscr{M}_i}^{(0)}$ of mobiles with RSC to $\boldsymbol{\pi}_i$.

In general, the worst case complexity is $O(k^3)$, where k is the size of $\boldsymbol{\Omega}$. With this solution, the size of $\boldsymbol{\Omega}$ is M^n and is $(K+1)\cdot M^n$ for the scenarios with and without ARQ, respectively. To reduce the complexity, we reduce the size of $\boldsymbol{\Omega}$ to M^{n_f} and $(K+1)\cdot M^{n_f}$ in each corresponding case. We also obtain the model for special case with $n_f = 1$ and $n_r = n-1$ where the size of $\boldsymbol{\Omega}$ becomes M and $(K+1)M$, respectively.

For *Case III*, we only need to keep track of channel states of mobiles with FSMC. Therefore, $\mathscr{M}^{(t)}$ in Section 3.3 reduces to $(\mathscr{M}_1^{(t)} \cdots \mathscr{M}_{n_f}^{(t)})$. Correspondingly, $\boldsymbol{f}_{\mathscr{M}}^{(0)} = \boldsymbol{f}_{\mathscr{M}_1}^{(0)} \otimes \cdots \otimes \boldsymbol{f}_{\mathscr{M}_{n_f}}^{(0)}$ and $\mathbf{P} = \mathbf{P}_1 \otimes \cdots \otimes \mathbf{P}_{n_f}$. Entries $g(\boldsymbol{m})$ and $g'(\boldsymbol{m})$ in $\mathbf{G}$ and $\mathbf{G}'$ are also modified to

$$g(\boldsymbol{m}) = \begin{cases} 0, & \exists i > 1 : m_i > m_1, \\ \sum_{k=0}^{n_r} \frac{\binom{n_r}{k}(\pi_{m_1})^k (F_{m_1-1})^{n_r-k}}{k+mult_{\boldsymbol{m}}(m_1)}, & m_i \leqslant m_1 \forall i, \end{cases} \tag{3.30}$$

$$g'(\boldsymbol{m}) = 1 - g(\boldsymbol{m}), \tag{3.31}$$

where F_{m_1-1} is the probability that channel state of a mobile in RSC will be less than

[4] *Case III* converges to *Case I* when $n_f = 0$ and $n_r = n$ and to *Case II* when $n_f = n$ and $n_r = 0$.

m_1. The derivation of (3.30) is given in Appendix B. The performance results after this point can be derived in the same way as in Section 3.3.

For the case[5] with $n_f = n-1$ and $n_r = 1$, we only keep track of the channel state of mobile 1. In Appendix B, we show that $g(m) = \frac{f_{\gamma_{sys}}(m)}{n\pi_c}$. We set $\mathbf{P}$ to $\mathbf{P}_1$, and set the (m,s) and (m,m) entries of $\mathbf{Q}$ and $\mathbf{Q}'$ to $p(s|m)$ and $p(0|m)$, respectively. After these basic matrices are obtained, we use the same methodology as for the *All-FSMC* case to obtain the relevant results.

3.5 Performance Evaluation

3.5.1 Numerical and Simulation Settings

For the different mobiles in a cell, we assume M-state i.i.d. wireless channels with each state being equally likely, and denote a steady state probability vector for each mobile by $\boldsymbol{\pi}$. We first set $p_{err} = 0$ and vary the number of mobiles (n) and the number of channel states (M) to study $f_{\gamma_{sys}}(m)$, $E[\gamma_{sys}]$, and $E[\gamma_{mob}]$ in *Case I* and *Case II*. In the former case, the expected inter-access delay ($E[\mathscr{D}_{acc}]$) is always equal to the number of mobiles. For the latter case, we show numerical results only for $n_f = 1$ and $n_r = n-1$ (the special case in Section 3.4). The results for more general cases can be generated from our framework as well.

For $n = 2$, we study the effect of average channel correlation (ρ in DEFINITION 2.2) and initial channel state on $\mathscr{D}_{acc}$ for the mobile with FSMC. When the initial state of the mobile is i, the initial probability vector is set to $\boldsymbol{e}_i$ whose i^{th} entry is 1 and all other entries are 0. We then investigate the impact of n and M on $E[\mathscr{D}_{acc}|\boldsymbol{e}_i]$. Again, $f_{\gamma_{sys}}(m)$, $E[\gamma_{sys}]$, and $E[\gamma_{mob}]$ are long-term performance metrics and are the same as those in *Case I*.

Next, we introduce non-zero $p_{err}^{(m)} = p_{err}, \forall m$, and show the results for expected inter-success delay ($E[\mathscr{D}_{sc}]$) and connection reset probability (p_{rst}) as a function of p_{err} and maximum number of retransmissions (K). The results for p_{sc} are complementary to those for p_{rst} (e.g., $p_{sc} = 1 - p_{rst}$). From (3.13), $E[\mathscr{D}_{rst}]$ always equals to

[5]This particular case could be useful to estimate the statistics at the mobile, which is aware only of its own state. The next best assumption is to use $\boldsymbol{\pi}$ as the initial probabilities or to use RSC model for all other $(n-1)$ mobiles.

$n(K+1)$. System throughput and per-mobile throughput are expected to decrease monotonically with increasing p_{err} and decreasing K. Based on the results of p_{rst}, p_{sc}, $E[\mathscr{D}_{rst}]$, $E[\mathscr{D}_{sc}]$, and $E[\mathscr{D}_{acc}](=n)$, we can verify the formulation of n_{rst}, n_{sc}, n_{fin}, and $E[\mathscr{D}_{fin}]$ in (3.17)-(3.20). These results as well as those in *Case II* can be obtained from our framework easily and are omitted for brevity.

We also compare the performance results for an opportunistic scheduler (shown with legend 'OPP') to those for a round-robin scheduler (shown with legend 'RR'), where all the mobiles are scheduled in sequence regardless of their channel states. Due to the deterministic nature of the round-robin scheduling, $f_{\gamma_{sys}}(m) = \pi_m$, $E[\gamma_{sys}] = \sum_{\forall m} m \cdot \pi_m$, $E[\mathscr{D}_{acc}] = E[\mathscr{D}_{acc}|\boldsymbol{e}_i] = n$, $f_{\gamma_{sys}^{err}}(s) = \sum_{\forall m} \pi_m \cdot p(s|m)$, and the results for p_{rst}, p_{sc}, $E[\mathscr{D}_{rst}]$, and $E[\mathscr{D}_{sc}]$ can be obtained from COROLLARY 3.2.

3.5.2 Simulation Methodology

We validate our analytical results by means of simulations using *MATLAB*. We collect data samples over 10^4 time slots and find their averages for each performance metric. A sample for $E[\gamma_{sys}]$ is the transmission rate for the selected mobile in each time slot. A sample for $E[\mathscr{D}_{acc}]$ is the interval between two consecutive slots where the same mobile is selected. These samples are classified based on their initial states and are used to calculate $E[\mathscr{D}_{acc}|\boldsymbol{e}_i]$. With non-zero p_{err}, a sample for $E[\mathscr{D}_{sc}]$ is the number of time slots from a connection reset or initiation (rst) or a transmission success (sc) to the next sc event. Again, the statistics for the number of time slots from an rst or an sc event to the next sc event is identical to those between two consecutive sc events because they both start with the retransmission counter set to zero.

3.5.3 AMC without ARQ: Results and Discussions

3.5.3.1 All-RSC

Figures 3.2-3.3 plot the expected values of system throughput ($E[\gamma_{sys}]$) and per-mobile throughput ($E[\gamma_{mob}]$) as functions of the number of channel states (M) and the number of mobiles (n). In both the figures, the simulation results (shown with legend 'Sim') follow the numerical results very closely. As expected, in case of an equally likely wireless channel, increasing M increases $E[\gamma_{sys}]$ and $E[\gamma_{mob}]$. In Appendix D,

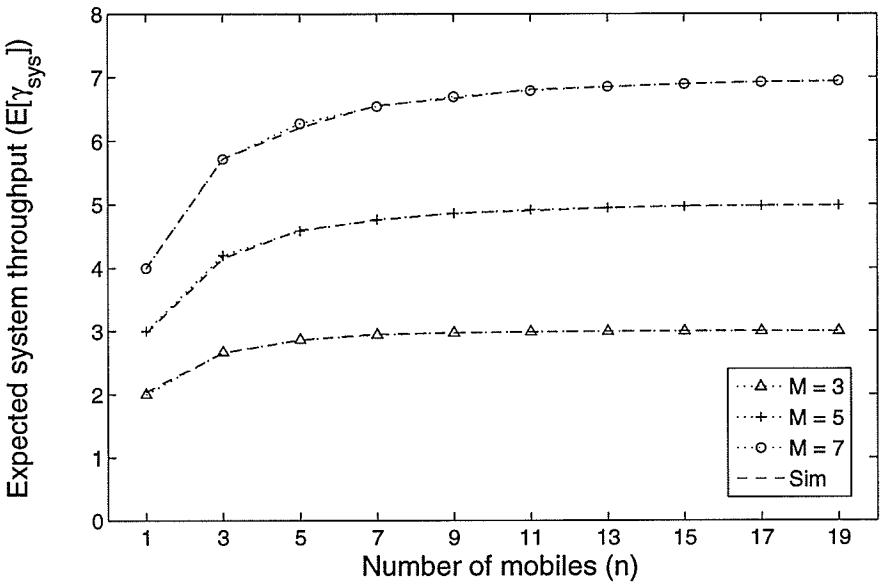


Figure 3.2. *Expected system throughput in* **Case I** *without ARQ.*

we also prove the this statement is also true for any equally likely channel with a channel-to-rate mapping which is an increasing function in m.

Increasing the number of mobiles leads to increased $E[\gamma_{sys}]$ due to the multi-user diversity gain. When $n = 1$, the *pmf* of system throughput $f_{\gamma_{sys}}(m) = \pi_m$ (Figure 3.4) and $E[\gamma_{mob}] = E[\gamma_{sys}]_{OPP} = E[\gamma_{sys}]_{RR}$ (Figure 3.3), where $E[\gamma_{sys}]_{RR} = \sum_{\forall m} m \cdot \pi_m$ is the average system throughput under round-robin scheduling, which is 2, 3, and 4 for $M = 3$, 5, and 7, respectively. For $n > 1$, $f_{\gamma_{sys}}(m)$ is shifted from $\boldsymbol{\pi}$ towards the best state and $E[\gamma_{sys}]_{OPP} > E[\gamma_{sys}]_{RR}$. Despite increasing diversity gain, admitting more mobiles into the system always results in reduction in per-mobile throughput as can be observed in Figure 3.3. To prove this statement, let differentiate (3.5),

$$\frac{\partial}{\partial n} E[\gamma_{mob}] = \sum_{m=1}^{M-1} \left((F_m)^{n-1} \cdot \left(\frac{F_m}{n^2} - 1 \right) \right) - \frac{M}{n^2}.$$

Since $\frac{F_m}{n^2} - 1 \leqslant 0$ with the equality when $n = 1$, $\frac{\partial}{\partial n} E[\gamma_{mob}] \leqslant 0$, and the per-mobile throughput is a decreasing function of n.

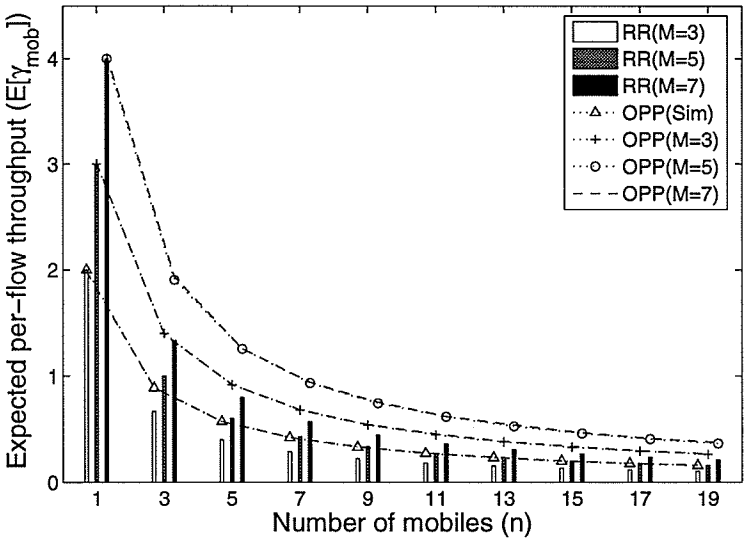


Figure 3.3. *Expected per-mobile throughput in* **Case I** *without ARQ.*

3.5.3.2 FSMC-RSC

Figure 3.5 plots the expected inter-access delay ($E[\mathscr{D}_{acc}|e_i]$) as a function of initial channel state (i) and average channel correlation (ρ) for $n_r = 1$ and $n_f = 1$. As a comparison, we also draw the delay (= 2 time slots) for the *All-RSC* case with $n = 2$. Again, this line acts as long-term inter-access delay for *Case II.*

From Figure 3.5, we observe that the inter-access delay for the mobile with FSMC depends strongly on the initial channel state. When the initial state is *good/bad*, the mobile will experience shorter/longer inter-access delay ($E[\mathscr{D}_{acc}|e_i]$). Note that, the initial state i in $E[\mathscr{D}_{acc}|e_i]$ refers to the state (i) when the mobile last acquired the channel access. At steady state, the state i is distributed according to $f_{\gamma_{sys}}(i)$. From these statistics, we can calculate long-term (or steady state) inter-access delay by using the *total probability theorem*; $E[\mathscr{D}_{acc}] = \sum_{i=1}^{M} f_{\gamma_{sys}}(i) \cdot E[\mathscr{D}_{acc}|e_i]$ (which is equal to 2 time slots in Figure 3.5). From Figure 3.4, we observe that the probability that a mobile with *bad* (initial) state (e.g., 1) acquires a channel access ($f_{\gamma_{sys}}(i)$) is rather small. In such a case, it is more likely that the mobile will have to wait for a long period of time before it is granted another channel access (Figure 3.5).

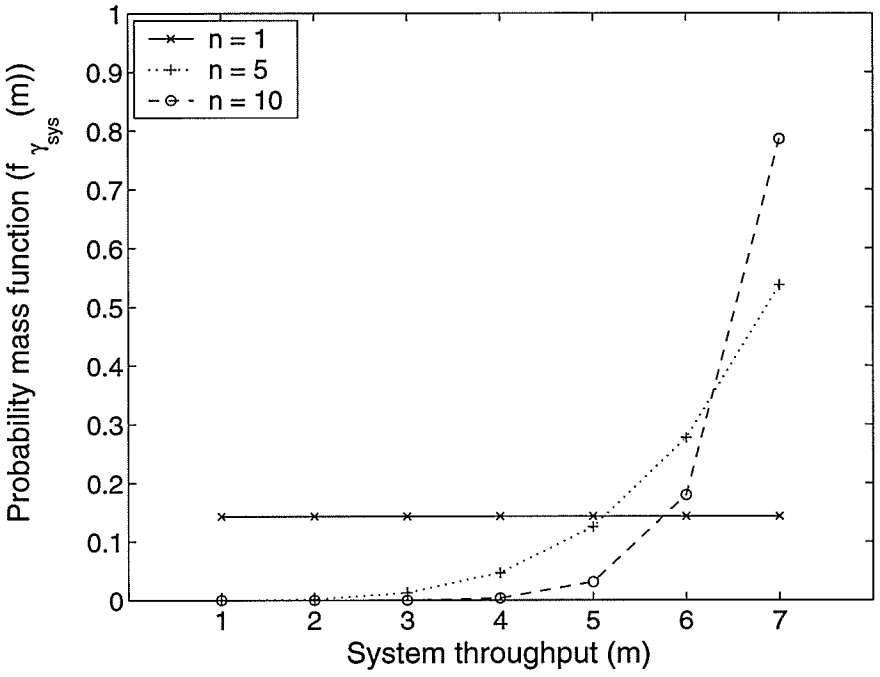


Figure 3.4. *Probability mass function of system throughput in* **Case I** *without ARQ for $M = 7$ (continuous plots are used for better readability).*

The delay variations due to initial state is augmented with increasing ρ, since the mobile is more likely to stay in the same state. We can observe in Figure 3.5 that the range of the delay with $\rho = 0.8$ is larger than that with $\rho = 0.3$.

In Figure 3.6, we set $\rho = 0.5$ and $n_r = 1$, and plot $E[\mathscr{D}_{acc}|\boldsymbol{e}_i]$ as a function of n. We also plot long-term delay ($= n$ time slots) for comparison. Intuitively, inter-access delay increases as n increases. We can observe stronger dependency of inter-access delay on initial states for larger number of mobiles. In this case, the mobile with FSMC starting from a *bad* initial state (e.g., located further from the base station) might not acquire the channel access after becoming eligible because of an increased number of total eligible mobiles. If not selected when being eligible, the mobile might experience *bad* channel states later in time. Again, it needs some time to regain the eligible status (e.g., be closer to the base station). As a result of these two effects, inter-access delay may increase significantly for *bad* initial states. This result is aggravated when M increases (e.g., $M = 7$), since the mobile might need

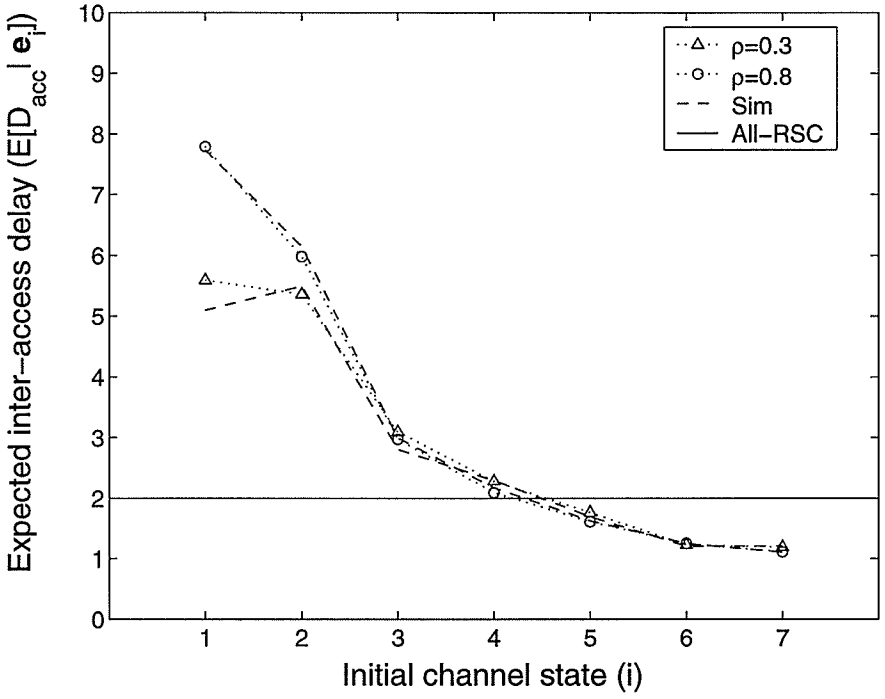


Figure 3.5. *Impact of initial channel state on expected inter-access delay.*

longer time to claim the eligible status.

To summarize, larger values of both n and M lead to large delay variation, which can be interpreted as a measure of instantaneous unfairness[6] among mobiles with different initial states. Here, we observe that an opportunistic scheduler offers the highest throughput but suffers from severe temporal unfairness (see DEFINITION 1.1). On the other hand, a round-robin scheduler achieves the best temporal fairness at the expense of degrading throughput.

3.5.4 AMC with ARQ: Results and Discussions

This section presents the results when ARQ is incorporated into the scheduler. In Figure 3.7, we set $n = 5$, $p_{err} = \{0, 0.15, 0.3\}$, and maximum number of retrans-

[6]Unfairness can be roughly estimated from the difference between the minimum and the maximum value of $E[\mathscr{D}_{acc}|e_i]$.

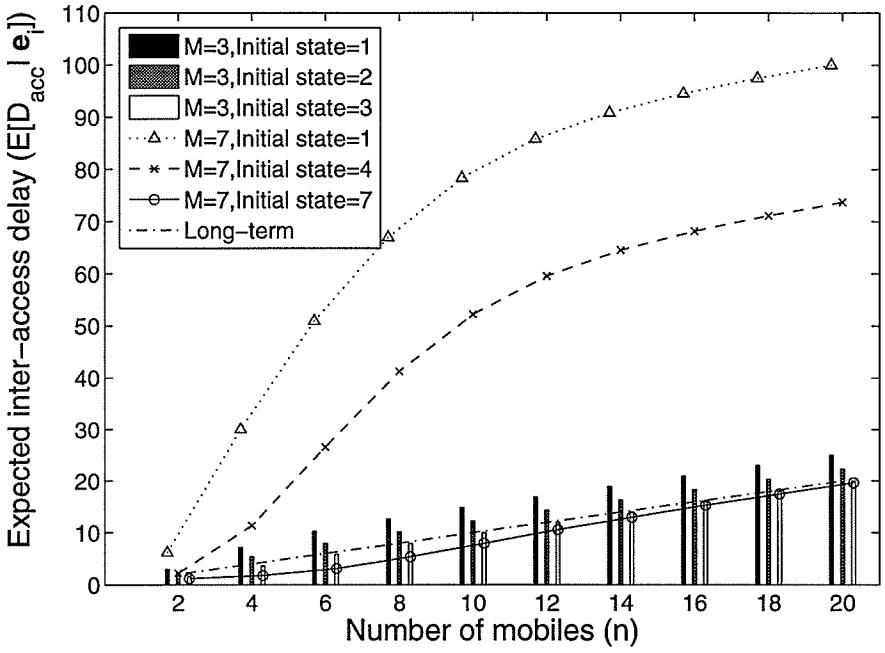


Figure 3.6. *Typical variations in expected inter-access delay with the number of mobiles for* **Case II** *without ARQ.*

missions $K = \{0, 1, 2, \infty\}$[7], and plot the expected inter-success delay ($E[\mathscr{D}_{sc}]$) for *Case I*. Figure 3.7(a) and Figure 3.7(b) represent the cases for $M = 3$ and $M = 7$, respectively. In Figure 3.8, we fix $K = 2$ and plot connection reset probability p_{rst} for different values of p_{err}. Figure 3.9 plots the joint cumulative distribution function (*cdf*) of cumulative delay d ($F_{\mathscr{S},\mathscr{D}}(s,d) = \sum_{i=1}^{d} f_{\mathscr{S},\mathscr{D}}(s,i)$) for $M = 7$ and $n = 10$. We plot the joint *cdf*s for $p_{err} = 0.05$ and $p_{err} = 0.1$ in Figure 3.9(a) and Figure 3.9(b), respectively. Finally, we show the *cdf* of inter-success delay ($F_{\mathscr{D}_{sc}}(d) = \sum_{i=1}^{d} f_{\mathscr{D}_{sc}}(i)$) with $M = 3$, $p_{err} = 0.15$, and $n = \{5, 10, 15\}$ in Figure 3.10.

Consider Figure 3.7 and 3.8 altogether. When $K = 0$, the transmission must be successful at the first channel access opportunity, otherwise the connection will be reset. Therefore, $E[\mathscr{D}_{sc}] = E[\mathscr{D}_{rst}] = E[\mathscr{D}_{acc}] = n$. For $K > 1$ and $p_{err} > 0$, each transmission might be unsuccessful and retransmission might be required.

[7]For $K = \infty$, we do not increase the retransmission counter and force the DTMC process to stay in state tx_0 for each failure.

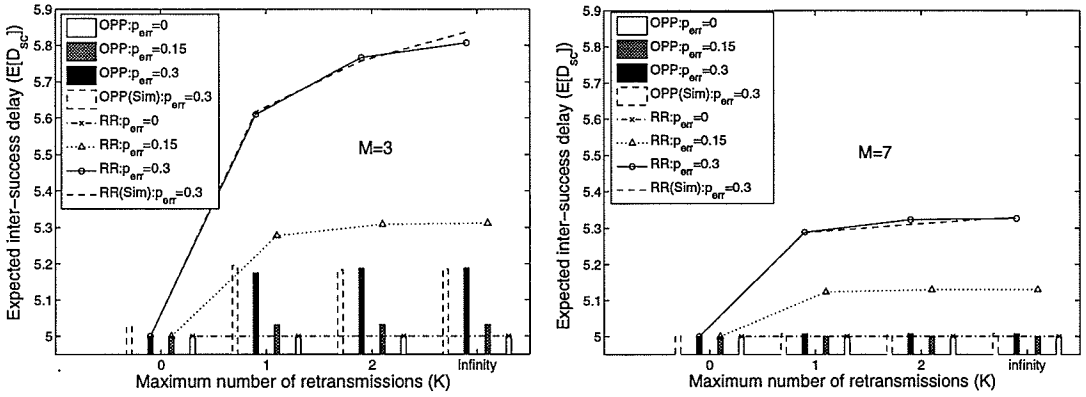


Figure 3.7. *Comparison of inter-success delay for opportunistic and round-robin scheduling with (a)* $M = 3$ *and* (b) $M = 7$.

Therefore, both p_{rst} and $E[\mathscr{D}_{sc}]$ increase with increasing p_{err}. On the other hand, $E[\mathscr{D}_{rst}](= n(K+1))$ is not affected by p_{err}, since it is conditioned on the occurrence of connection reset. Here, we observe the tradeoff between reliability ($p_{sc} = 1 - p_{rst}$) and latency ($\mathscr{D}_{sc}$) in that improving the successful transmission probability p_{sc} by means of retransmission could lead to an increase in $\mathscr{D}_{sc}$.

Since an opportunistic scheduler selects a mobile with *good* channel quality, the selected mobile is expected to transmit/receive several packets per time slot. With round-robin scheduling, on the other hand, the mobile tends to transmit/receive fewer number of packets per time slot. In Figure 3.8, we observe that with opportunistic scheduling p_{rst} is always less than that with round-robin scheduling. For an equally-likely wireless channel, for both the schemes, increasing M results in an increase in the probability of transmitting more packets, and therefore, decreases p_{rst}. Despite increasing delay, for opportunistic scheduling, $E[\mathscr{D}_{sc}]$ becomes saturated very quickly. The values of $E[\mathscr{D}_{sc}]$ are very close to those for infinite persistence even with only two retransmissions, while the difference in case of round-robin scheduling is still perceptible for $K > 2$.

Figure 3.9 presents the joint probability that s packets will be successfully transmitted within a certain threshold d. We observe that there exists a most likely point for the number of successfully transmitted packets (s) corresponding to each value

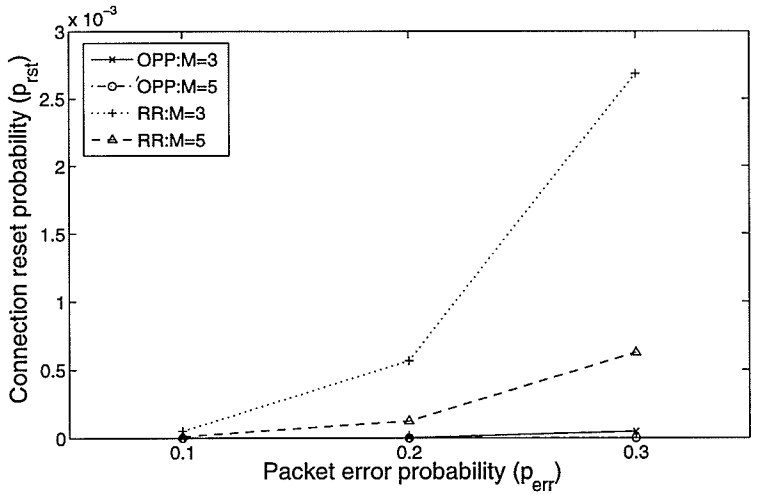


Figure 3.8. *Comparison of connection reset probability for opportunistic and round-robin scheduling.*

of cumulative delay. This point is a decreasing function in p_{err} (as can be verified in (3.6)), but remains unchanged for increasing cumulative delay.

By integrating all states $s > 0$ and normalizing the result, we obtain $F_{\mathscr{D}_{sc}}(d)$, which is the probability that at least one packet is successfully transmitted within the limit d. The plot of $F_{\mathscr{D}_{sc}}(d)$ in Figure 3.10 reveals that delay variation can be very wide, and the variation tends to increase with increasing n. Even with the best case ($n = 5$), after $E[\mathscr{D}_{sc}]$, the probability that a packet will be successfully transmitted is not more than 73.78%. This implies that the expected value might not be a true indicator of the instantaneous inter-success delay.

The above joint *pmf* and *cdf* would be useful in three different ways. First, we can utilize these statistics to obtain more accurate information about end-to-end round-trip time (RTT) in a wide-area wireless network, which in turn can be used to minimize the number of timeouts in the end-to-end flow control protocol (e.g., TCP timeouts). Secondly, we can also estimate the *pmf* of link layer bandwidth ($f_{\mathscr{R}}(r)$) from $\sum_{\forall s} \boldsymbol{f}_{\mathscr{S},\mathscr{D}}(s, s/r)$, and use this parameter to probabilistically set the transport layer transmission window size at the sender. Thirdly, for real-time data services, in which each data packet would be useless after some time τ, $F_{\mathscr{D}_{sc}}(\tau)$ is the probability to deliver packets within the delay limit, and can be used as a measure for QoS.

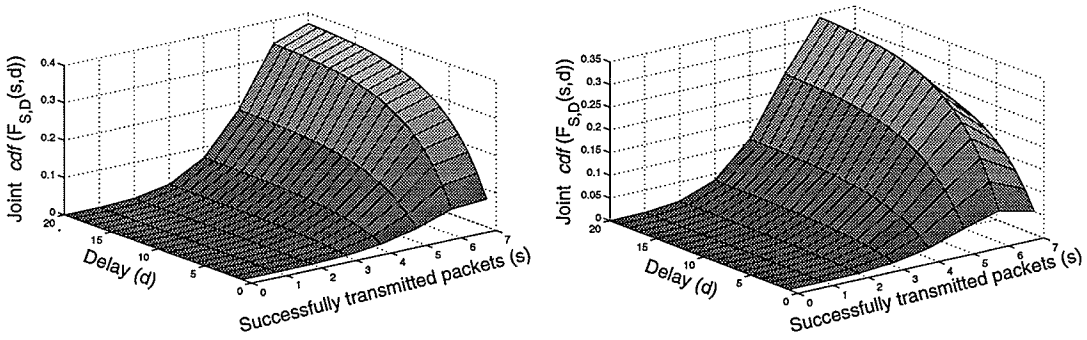


Figure 3.9. *Joint cumulative distribution function for* **Case I** *with ARQ when* (a) $p_{err} = 0.05$ *and* (b) $p_{err} = 0.1$.

3.6 Chapter Summary

This chapter presents an analytical model for radio link level channel-quality-based opportunistic scheduling with AMC and ARQ considering both uncorrelated and correlated wireless channels. We have derived complete statistics (in terms of probability mass functions) for the performance measures including system throughput, per-mobile throughput, inter-access delay, connection reset delay, and inter-success delay. Analytical results have been validated through simulations.

Although the multi-user diversity gain is an increasing function in number of mobiles and number of channel states, admitting more mobiles into the system always reduces per-mobile throughput, increases delay variation, and degrades temporal fairness. Inter-access delay for a mobile with an FSMC strongly depends on initial channel states. The delay variation (among different initial states) is an increasing function of channel state correlation and diversity gain. The *pmf* and/or *cdf* for the delay reveal the tradeoff between the probability of successful transmission (i.e, reliability) and the corresponding delay.

Performance results for an opportunistic scheduler have been compared to those of a round-robin scheduler under different AMC and channel parameters. In most cases, an opportunistic scheduling leads to higher throughput and lower connection reset probability due to multi-user diversity. However, it also leads to higher delay variation and temporal unfairness in some cases.

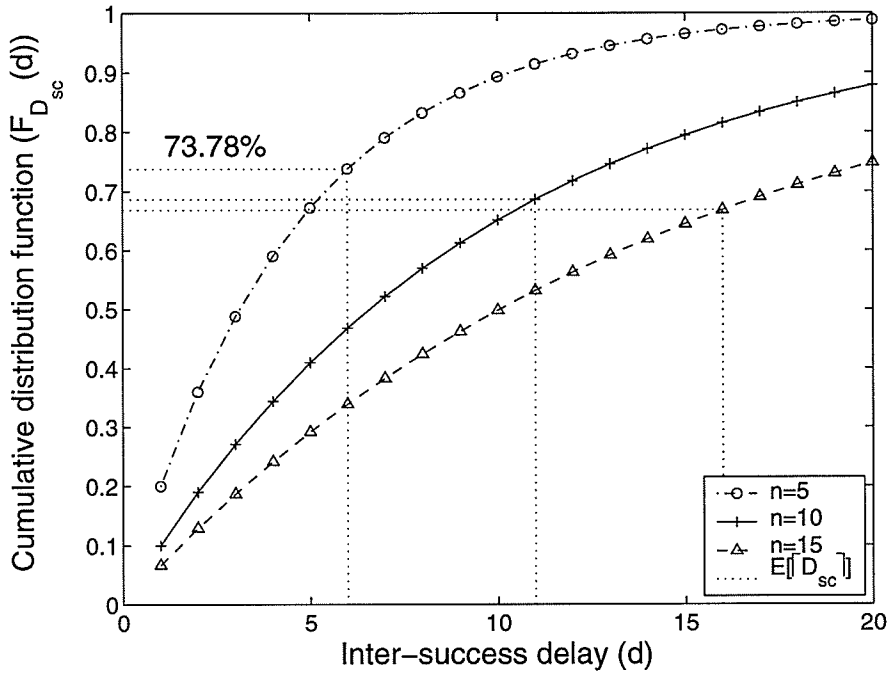


Figure 3.10. *Cumulative distribution function for inter-success delay in* **Case I** *with ARQ.*

Since the proposed analytical framework derives the complete statistics for radio link level throughput and delay, it would be useful in modeling and optimizing higher layer protocol performance.

Chapter 4

Fair Scheduling Algorithms for Single-Rate Transmission

Providing fair share of channel bandwidth among the different mobiles is one of the key issues in provisioning QoS in wireless data networks. The fairness is achieved by using a scheduling protocol to allocate all available resources (e.g., channel bandwidth) among the mobiles in proportion to their weights.

This chapter discusses fair scheduling algorithms under a single-rate transmission environment. Under this environment, a wireless channel can be modeled by using a GEC model (see Section 2.3.2). In principle, a mobile perceiving a *bad* channel should defer the transmission and let another mobile with a *good* channel transmit data. To provide fair service allocation, the scheduler should compensate for such the deferred services when the channel of the deferred mobile becomes *good* again. A mobile deferring the transmission is considered to be *lagging*, while a mobile receiving extra allocation is considered to be *leading*. General wireless fair queuing algorithms consist of error-free service, lead and lag model, compensation model, slot queue and packet queue, and channel monitoring and prediction [12]. A collection of fair scheduling algorithms which follow this framework are given in [12] and [47]. We observe that all these algorithms obtained the solution in a heuristic manner, and propose an optimization-based fair scheduling algorithms namely *Optimal Radio Channel Allocation for Single-Rate Transmission (ORCA-SRT)*.

The organization of this chapter is as follows. Fair scheduling algorithms proposed in the literature are outlined in Section 4.1. Section 4.2 discusses the proposed ORCA-SRT. Simulation environment and results are presented in Sections 4.3 and 4.4. Finally, the chapter summary is given in Section 4.5.

4.1 Fair Scheduling Algorithms for Wireless Networks

Well-known fair scheduling algorithms for wireless networks in the recent literature include *Channel State Dependent Packet Scheduling (CSDPS)* [48], *Idealized Wireless Fair Queuing (IWFQ)* [49], *Server Based Fairness Algorithm (SBFA)* [50], *Class Based Queuing with Channel State Dependent Packet Scheduling (CBQ-CSDPS)* [51], *Wireless Packet Scheduling (WPS)* [9], *Channel Independent Fair Queuing (CIF-Q)* [10], and *Wireless Fair Service Algorithm (WFS)* [11]. For the sake of brevity, we will presents only some of them.

4.1.1 Weighted Round Robin (WRR)

WRR allocates time slots to each mobile in proportion to the corresponding weight [12]. For example, for three mobiles with weights 1, 3 and 2, allocation of time slots for these mobiles in a scheduling frame would be $\{x_1, x_2, x_2, x_2, x_3, x_3\}$.

4.1.2 Weighted Fair Queuing (WFQ)

In WFQ [52], the scheduler maintains the following start tags (S_i^k) and finish tags (F_i^k) for all the mobiles:

$$S_i^k = max\left\{V(A_i^k), S_i^{k-1} + \frac{l_i^{k-1}}{w_i}\right\}, \tag{4.1}$$

$$F_i^k = S_i^k + \frac{l_i^k}{w_i}, \tag{4.2}$$

$$\frac{dV(t)}{dt} = \frac{C(t)}{\sum_{\text{all backlogged mobiles}} w_i}, \tag{4.3}$$

where A_i^k and l_i^k are packet arrival time and the length of packet k of mobile i, w_i is the weight of mobile i, $C(t)$ is the instantaneous channel capacity, and the function $V(t)$ can be calculated from (4.3).

After the calculation of start and finish tags, the time slot is assigned to the mobile with minimum finish tag. For the above example, the allocation would be $\{x_2, x_3, x_2, x_1, x_3, x_2\}$.

Similar to WRR, WFQ distributes resources among all the mobiles in proportion to their weights. WFQ also reduces the severity of burst errors by spreading out the allocation.

4.1.3 Wireless Packet Scheduling Protocol (WPS)

This algorithm employs WFQ as its error free service. By assuming perfect knowledge of channel condition, the scheduler will try to swap the allocated time slots, if the owner of a certain time slot perceives a *bad* channel. For example, if mobile i is allowed to transmit but it finds the current time slot *bad*, WPS will search in the forward direction for another time slot which is *good* for mobile i. If the owner of the newly found slot perceives the slot possessed by mobile i *good*, the allocation of both the mobiles will be swapped. If not, the current time slot will be given to another mobile whose channel condition during the current slot is *good*. The mobile relinquishing a time slot will be regarded as *lagging* while the mobile receiving an extra time slot will be regarded as *leading*. In the implementation, the lead and lag might be represented by positive and negative numbers. At the beginning of each time frame, before scheduling, a set of *effective weights* is calculated by subtracting the lead-lag counter from the original weights of each mobile. All the scheduling computations afterward will be performed using the effective weights rather than the original weights.

4.1.4 Channel-condition Independent Packet Fair Queuing (CIF-Q)

The error-free service of CIF-Q is based on *Start Time Fair Queuing* (STFQ) [53]. STFQ is very similar to WFQ. However, instead of calculating $V(t)$ from (4.3), STFQ sets the virtual time, $V(t)$ to the current time. CIF-Q simulates error free service allocation for all the mobiles and calculates corresponding lead-lag counters as the difference between real allocation and the error free service allocation. Unlike WPS, the mobile selected by the error free service and perceiving *good* channel condition in CIF-Q can be deferred with probability α. If the selected mobile is deferred, CIF-Q will allow another lagging mobile with minimum extra service to transmit. If the

Table 4.1. *Compensation sequence of WFS*

Sequence	Condition
1	A mobile from the error free service if it is either lagging or leading and rescheduled to transmit.
2	Another lagging mobile.
3	The leading mobile which was rescheduled to give up its time slot in sequence number 1.
4	Another leading mobile with the lead counter less than the maximum allowable lead.
5	Another in-sync mobile.
6	Any other mobile.

selected mobile cannot transmit, any other mobile with minimum extra service will be allowed to transmit.

4.1.5 Wireless Fair Service (WFS)

In order to support delay-bandwidth decoupling, WFS modifies WFQ by using the rate weight (w_i) and delay weight (ϕ_i). The start tag is calculated in the same way as WFQ. However, the finish tag is calculated as follows:

$$F_i^k = S_i^k + \frac{l_i^k}{\phi_i}. \tag{4.4}$$

The error free service selects the mobile with minimum finish tag which satisfies the constraint, $S_i^k \leqslant V + \varrho$, where V is the current virtual time and ϱ is a *lookahead* parameter. After obtaining the error free service solution, the scheduler will allocate each time slot to a mobile using the sequence specified in Table 4.1 only if the mobile finds the current time slot to be *good*. If not, the scheduler will select the next mobile in the sequence.

In the sequence number 1, the leading mobile is forced to give up the allocation for the $lead/lead_{max}$ fraction of time slots and is allowed to transmit for the $1 - (lead/lead_{max})$ fraction of time slots. Lagging mobiles in the sequence number 2 are selected to regain extra allocation by using their lag counters as weights in a weighted round-robin manner. If the mobile from the error free service is not scheduled to

transmit, its lead-lag counter is decreased, while the counter of the mobile which is scheduled to transmit in place of another mobile is increased.

4.2 Optimal Radio Channel Allocation for Single-Rate Transmission (ORCA-SRT)

ORCA-SRT is designed under a GEC model, where the channel state can be either *good* or *bad*. In this chapter, we assume that a data packet is properly delivered in a *good* time slot and is unsuccessfully transmitted when the time slot is in *bad* condition (i.e., $p_{err}^{(g)} = 0$ and $p_{err}^{(b)} = 1$). ORCA-SRT formulates a wireless scheduling problem as an *assignment problem*, where each individual can be regarded as a mobile and each job can be regarded as a time slot. ORCA-SRT uses the following three-step procedure:

- **Step 1:** Setup the problem as an assignment problem and formulate the total cost (C_T) in (4.5), by using *Single-Rate Transmission Cost Matrix* ($\mathbf{C}_{SRT}$).
- **Step 2:** Find the solution of the assignment problem (see Appendix F).
- **Step 3:** Minimize the useless transmission by means of an explicit compensation.

4.2.1 Assignment Problem

Given k individuals, k jobs, as well as a set of costs $c_{ij} \in \mathbb{I}^{0+}$ corresponding to the assignment of j^{th} job to i^{th} individual, the *assignment problem* is to assign jobs among the individuals such that each individual does exactly one job and each job is done by exactly one individual, and the assignment leads to total minimum cost [54]. Mathematically,

$$min \quad C_T = \sum_{i=1}^{k} \sum_{j=1}^{k} c_{ij} \cdot x_{ij} \tag{4.5}$$

$$subject\ to \quad x_{ij} = \begin{cases} 1, & \text{individual } i \text{ is assigned to job } j, \\ 0, & \text{otherwise,} \end{cases} \tag{4.6}$$

$$\sum_{i=1}^{k} x_{ij} = 1, \tag{4.7}$$

$$\sum_{j=1}^{k} x_{ij} = 1. \tag{4.8}$$

A solution to this problem can be obtained by using the *Hungarian Method* which is related only to the manipulation of the matrix $\mathbf{C}$ whose entries are c_{ij} [54]. Since the objective is to minimize the cost function, the solution is located where the cost is minimum, i.e., $x_{ij} = 1$ if $c_{ij} = 0$. Note that, if $c_{ij} = 0$, x_{ij} will not necessarily be equal to 1. The detail of the Hungarian Method is given in Appendix F.

4.2.2 Cost Function

The *single-rate transmission cost matrix* ($\mathbf{C}_{SRT}$) is established as follows:

1. Calculate scheduling frame size $T_F = \sum_{i=1}^{n} w_i$, where w_i is the weight of mobile i and n is the number of mobiles.
2. For mobile i, calculate the cost of allocating time slot j (c_{ij}) by using the following formula:

$$c_{ij} = 1 - m_{ij}, \quad j = \{1, 2, \cdots, T_F\}, \tag{4.9}$$

where

$$m_{ij} = \begin{cases} 0, & \text{if slot } j \text{ is } \textit{bad} \text{ for mobile } i, \\ 1, & \text{if slot } j \text{ is } \textit{good} \text{ for mobile } i. \end{cases}$$

3. Put w_i identical rows of mobile i into $\mathbf{C}_{SRT}$.
4. Repeat step 2 - step 3 for all the mobiles.

Observation 4.1 (COST FUNCTION FOR ORCA-SRT): *If the cost function in ORCA-SRT is defined as in (4.9), then the assignment solution minimizes the number of unsuccessful transmissions.*

PROOF: *Eq. (4.9) sets the cost of a* good *state to be lower than that of a* bad *state. In order to minimize total cost, the Hungarian method always assigns* $x_{ij} = 1$ *(in (4.5)) to the entry with lower* c_{ij}*. The mobile with* bad *state* ($c_{ij} = 1$) *will not be selected, if another mobile with* good *state* ($c_{ij} = 0$) *can be chosen. Therefore, the number of mobiles transmitting in the* bad *channel is minimized.* ■

4.2.3 Finding Solution of the Assignment Problem

Subject to the constraints in (4.7) and (4.8), the Hungarian method is utilized to find the optimal time slot assignment that minimizes C_T. In time slot j, mobile i will be allowed to transmit, only if $x_{ij} = 1$.

Definition 4.1 INTER-ACCESS DELAY OF MOBILE i ($\mathscr{D}_{acc_i}^{(k)}$) *is an interval between the transmission of* $(k-1)^{th}$ *and* k^{th} *packet of mobile* i. □

Observation 4.2 (PROPERTIES OF THE ASSIGNMENT SOLUTION):

1. *guarantees that only one mobile can transmit in a time slot,*
2. *guarantees fairness,*
3. *guarantees that inter-access delay is bounded to* $(2\sum_{\forall j \neq i} w_j + 1)$ *,*
4. *does not guarantee successful transmission.*

PROOF: *The above properties can be proven as follows:*

1. *From the constraint in (4.7), there is exactly one mobile transmitting in a time slot.*
2. *From the constraint in (4.8), each mobile transmits exactly one time slot per frame. Assuming that every mobile has the same weight, the fairness is ensured.*
3. *In two scheduling frames, mobile* i *transmits in exactly* $2w_i$ *time slots. Therefore, the maximum inter-access delay is bounded to* $(2\sum_{\forall j \neq i} w_j + 1)$.
4. *If all mobiles perceive the channel* bad *in a specific time slot, from (4.7), one of them must be selected and transmit unsuccessfully. Similarly, if one mobile*

does not find any good *channel in a scheduling frame, it must be allocated with one of the* bad *time slots and transmit unsuccessfully (eq. (4.8)).* ■

4.2.4 Lead-lag and Compensation Model

The assignment solution obtained in step 2 might lead to unsuccessful transmission. Similar to WFS, the third step utilizes explicit compensation specified in Table 4.1 to eliminate all the residual ineffectual transmissions. In this step, each mobile perceiving *good* channel condition is allocated according to the same process as that of WFS.

Observation 4.3 (VIOLATION OF DELAY BOUND AND FAIRNESS): *The compensation mechanism in step 3 of ORCA-SRT violates property 1 and 2 of the assignment solution.*

PROOF: *By giving a time slot allocated to one mobile to another, some mobiles might not get any allocation in a scheduling frame. The inter-access delay is no longer bounded to* $(2\sum_{\forall j \neq i} w_j + 1)$. *Fairness in that frame also degrades because of unequal amount of slot allocation.* ■

Note that, step 3 in ORCA-SRT results in unfairness. However, it should be implemented since the residual ineffectual transmission in step 2 is detrimental to both fairness and delay bound. Step 3 is necessary as long as the allocation obtained in step 2 causes transmission in *bad* channels.

4.2.5 Implementation in a TDMA-Based MAC Framework

In ORCA-SRT, scheduling frame size must be equal to $\sum_i w_i$ (i.e., number of rows or columns in the cost matrix) which may not be equal to the MAC layer frame size in a TDMA-based channel access scenario. In fact, the ORCA-SRT-based scheduling can be applied to any MAC frame size. If the scheduling frame size is larger than the MAC layer frame size, the MAC frame can be filled up using the ORCA-SRT-based allocation. In this case, some of the allocations which have not been used to fill the MAC frame, will be used to fill the beginning of the next MAC frame. On the other hand, if the scheduling frame size is smaller than the MAC frame, several rounds of ORCA-SRT-based scheduling will be needed to fill the MAC frame.

Note that, even if the weights of the mobiles are non-integer, they can be normalized to integer values without changing the proportion of the weights.

4.2.6 Complexity

The worst case time-complexity of the Hungarian method is $O(k^3)$ [55], where k, the number of columns and/or rows in the $\mathbf{C}_{SRT}$, is a function of the number of active mobiles and variations in the weights of the mobiles. The compensation model is similar to that of WFS. Therefore, its complexity is $O(n)$, where n the number of simultaneous mobiles. Note that other approaches designed to solve an assignment problem include *Successive Shortest Path Algorithm*, *Relaxation Algorithm*, *Cost Scaling Algorithm*, and *Stable Marriage Problem* [56]. Although the use of these approaches could lead to lower complexity, we leave this issue for future study.

4.3 Simulation Environment

4.3.1 Performance Measures

When all the mobiles have equal weights, the following performance matrics are defined:

- **Unfairness** (σ):

$$\sigma = \left(\frac{1}{n} \cdot \sum_{i=1}^{n} \left(\frac{K_i}{\sum_{j=1}^{n} K_j} - \frac{1}{n} \right)^2 \right)^{\frac{1}{2}}. \tag{4.10}$$

- **Expected inter-access delay** ($E[\mathscr{D}_{acc}]$):

$$E[\mathscr{D}_{acc}] = \frac{1}{n} \cdot \sum_{i=1}^{n} \left(\frac{\sum_{k=1}^{K_i} \mathscr{D}_{acc_i}(k)}{K_i} \right). \tag{4.11}$$

where K_i is the number of packets successfully transmitted by mobile i over a certain observation period, n is the number of mobiles, and $\mathscr{D}_{acc_i}(k)$ is inter-access delay of mobile i from packet $(k-1)^{th}$ to packet k^{th}.

Each mobile is assumed to have a buffer with the size of one packet and no new packet is generated until the packet in the buffer has been transmitted. Note that,

the unfairness (σ) is based on the average Euclidean distance between the actual allocation to the mobiles' weights. In the ideal situation (in the absence of loss), $\sigma = 0$ and $E[\mathscr{D}_{acc}] = n$.

4.3.2 Simulation Parameters and Methodology

Using *C++*, we run the simulation for 10^6 time slots. Each parameter is averaged over 1000 samples, each of which is the average value in a 200 time-slot observation window. Each wireless channel corresponding to each mobile is assumed to be independently varying. In all following simulations, we assume 5 simultaneous mobiles each with the weight of 1.

The single-rate fair scheduling algorithms considered here are the WPS, CIF-Q, WFS, and ORCA-SRT algorithms. We set $p_{err} = \{0.1, 0.15, 0.3\}$, and $\rho = \{0.5, 0.7, 0.85, 0.95\}$. Lead-lag counter is assumed to be bounded within $[-4, 4]$. For CIF-Q, we set $\alpha = 0$ as in [10]. For WFS, we set $\phi_i = w_i$ and $\varrho = \infty$ as in [11].

4.4 Simulation Results and Discussions

4.4.1 Delay Performance Under Single-Rate Transmission

Figure 4.1 shows that WPS incurs higher expected inter-access delay as compared to CIF-Q, WFS, and ORCA-SRT schemes. Also, for WPS the average delay increases with increasing packet error probability, while for the other schemes the average delay does not vary significantly as p_{err} changes. For CIF-Q, WFS, and ORCA-SRT, the average delay is very close to the ideal delay ($\mathscr{D}_{acc_i}(k) = n = 5, \forall i, k$).

As the channel errors become more correlated, the expected inter-access delay increases because each mobile has higher possibility to experience *bad* channels for a longer period of time, and might not get any allocation in a scheduling frame (Figure 4.1).

In case of WPS, increased channel error correlation may cause the lead-lag counter to exceed the maximum limit, and consequently, the deferred mobile will not be compensated. As a result, the delay performance deteriorates significantly. For the CIF-Q, WFS, and ORCA-SRT schemes each mobile is compensated before its lead-

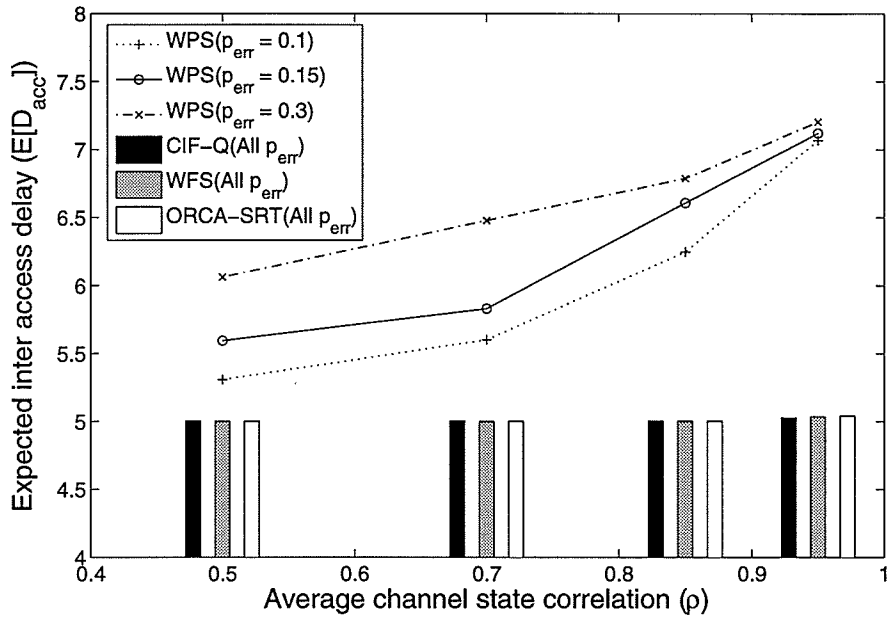


Figure 4.1. *Expected inter-access delay under single-rate transmission.*

lag counter exceeds the limit, and therefore, increased channel error correlation does not have significant impact on the average delay performance in this case.

4.4.2 Fairness Performance in Single-Rate Transmission

As the channel becomes more error prone and/or the channel errors become more correlated, in common with the average delay, unfairness increases (Figures 4.2 and 4.3).

CIF-Q, WFS, and ORCA-SRT schemes provide better fairness compared to WPS (Figure 4.2). This is due to the fact that WPS has the limitation of forward swapping only, while the other schemes allow both forward and backward swapping. Due to the optimization, ORCA-MRT performs slightly better than both CIF-Q and WFS.

4.5 Chapter Summary

Fair scheduling schemes such as WPS, WFS, or CIF-Q do not perform optimization, and therefore, the best solution might not be found in some cases. We have

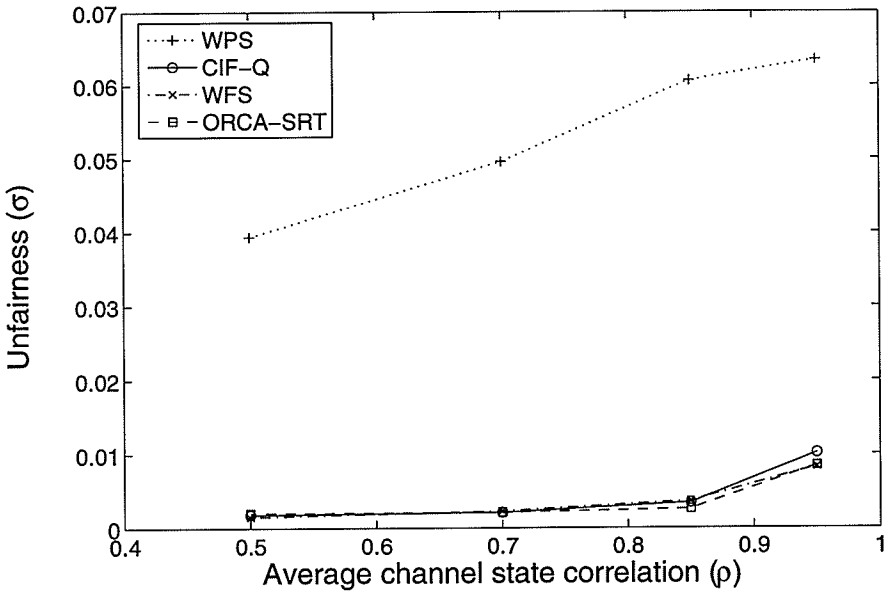


Figure 4.2. *Unfairness under single-rate transmission for* $p_{err} = 0.1$.

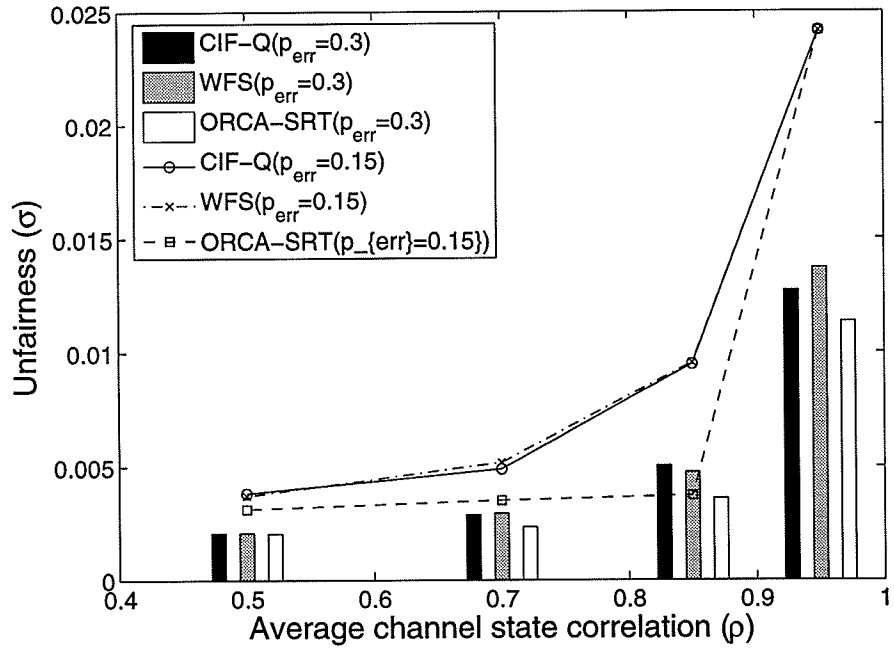


Figure 4.3. *Unfairness under single-rate transmission for* $p_{err} = 0.15$ *and* 0.3.

formulated the scheduling problem for fair bandwidth allocation among mobiles as an assignment problem. The optimization using the Hungarian method and lead-lag compensation constitute the *Optimal Radio Channel Allocation for Single-Rate Transmission (ORCA-SRT)* protocol. Simulation results have shown that, due to such optimization the ORCA scheme outperforms WPS, WFS, and CIF-Q schemes.

Chapter 5

Fair Scheduling Algorithms for Multi-rate Transmission

The problem of fair bandwidth allocation in TDMA-based wireless environment has been studied quite extensively in recent literature. [12] summarizes most of the proposed heuristic-based approaches for fair bandwidth allocation (e.g., *Wireless Packet Scheduling* (WPS), *Channel Independent Fair Queuing* (CIF-Q), *Wireless Fair Service* (WFS) algorithms). In Chapter 4, we propose an optimization-based approach, ORCA-SRT, and show that the proposed algorithm provides improved performance over the heuristic-based approaches in the literature. However, all the above scheduling algorithms including ORCA-SRT are based on the assumption that only one mobile can transmit at an instant and at one transmission rate only.

It is well known that in a wireless network the spectrum efficiency of the radio channels can be substantially increased by using dynamic rate adaptation based on the channel interference and fading conditions [33]. The dynamic transmission rate can be achieved, for example, through an *Adaptive Modulation and Coding (AMC)* technique in *Time Division Multiple Access (TDMA)*-based systems and through variable spreading gain and/or multi-code transmission in *Code Division Multiple Access* (CDMA)-based systems. Analysis of the fair-queuing problem under multi-rate transmission, therefore, reveals the interesting inter-relationship among the physical level transmission parameters and radio link level performance measures.

Several works on fair scheduling in a multi-rate system have been reported in the recent literature. The *Opportunistic Auto Rate* algorithm presented in [57] allows mobiles perceiving good channel condition to transmit several packets consecutively and ensures that all the mobiles acquire channel accesses for the same long-term time-

shares (i.e., *temporal fairness*[1]). The algorithm presented in [42] stochastically fixes a fraction of time slot allocation to each mobile and maximizes the overall throughput in a multi-rate TDMA cellular system. Neither of the two above algorithms provides *throughput fairness.* A *Multi-channel Fair Scheduler (MFS)* for a CDMA network, which maximizes system throughput while maintaining fairness among all the mobiles, was presented in [42]. In MFS, several mobiles are allowed to transmit at the same time as long as summation of the power of transmitted signals does not exceed a certain threshold. However, MFS does not guarantee temporal fairness.

In this chapter, we propose a framework, namely, *Optimal Radio Channel Allocation for Multi-Rate Transmission (ORCA-MRT)* to solve the combined *temporal-throughput fair scheduling problem* in a multi-rate TDMA network. ORCA-MRT ensures fair time slot allocation in each frame and maximizes overall throughput without deteriorating throughput-fairness. The fair scheduling problem is formulated as an *assignment problem* [58]. The properties of ORCA-MRT are analyzed by observing certain aspects of the modified assignment problem, and simulation results are presented in support of these mathematical observations. Also, two channel prediction methods are proposed to facilitate the optimal bandwidth allocation when channel state information cannot be perfectly known.

The remainder of this chapter is organized as follows. Section 5.1 presents the background and motivation. The architecture of the ORCA-MRT scheduling framework is presented in Section 5.2. Section 5.3 describes the simulation environment and the performance metrics. The simulation results are presented in Section 5.4. Chapter summary are given in Section 5.5.

5.1 Background and Motivation of the Work

5.1.1 Multi-channel Fair Scheduler (MFS)

Two variants of MFS, namely MFS-D (Deterministic fairness problem) and MFS-P (Probabilistic fairness problem), are designed to solve two fairness problems [42] . For MFS-D, the problem is defined as follows:

[1]See the definition in Chapter 1.

$$max \quad \sum_{i=1}^{n} E[X_i(k)] \tag{5.1}$$

$$subject\ to \quad \frac{E[X_i(k)]}{w_i} = \frac{E[X_j(k)]}{w_j}, \tag{5.2}$$

$$\sum_{i=1}^{n} c_i(k) \cdot X_i(k) \leqslant P, \tag{5.3}$$

$$X_i(k) \in \{0, R_i^1, \cdots, R_i^{M_i}\}, \tag{5.4}$$

where $X_i(k)$ is the transmission rate of mobile i in time slot k, w_i is the weight of mobile i, n is total number of mobiles, and $E[\cdot]$ is an expectation function. Possible values of transmission rate are given in (5.4), where mobile i has M_i+1 possible rates (0 means no transmission). The channel condition for mobile i at time t, $c_i(t)$, is calculated from

$$c_i(t) = 0.5 + d \cdot cos(2\pi f_i t + \theta_i) + X_{\sigma_i}(t), \tag{5.5}$$

where θ_i is a random variable uniformly distributed in $[0, 2\pi)$, $X_{\sigma_i}(t)$ models *Additive White Gaussian Noise* (AWGN) with variance σ_i^2, f_i expresses the channel correlation due to mobility over long time scale, and d is a scaling factor exhibiting the range of the channel variation.

MSF-P is devised to solve the probabilistic fairness problem in which the constraint in (5.2) was replaced by that in (5.6) below

$$Pr\left(\left|\frac{E[X_i(k)]}{w_i} - \frac{E[X_j(k)]}{w_j}\right| > \delta\right) \leqslant P_\delta, \tag{5.6}$$

where δ is the service discrepancy defining the tolerable deviation from ideal fairness.

Both MFS-D and MFS-P employ a *greedy algorithm* with the following preference list:

$$\frac{c_h(k)}{u_h(k)} \leqslant \frac{c_i(k)}{u_i(k)} \leqslant \cdots \leqslant \frac{c_j(k)}{u_j(k)}. \tag{5.7}$$

MFS sequentially chooses the next mobiles in the list until the maximum power limit (5.3) is reached. The control vector $\mathbf{u} = [u_1, u_2, \cdots, u_n]$ is selected by using

a *stochastic-approximation*-based iterative algorithm [59] so that the solution from the greedy algorithm ($X_i(k)$) satisfies (5.8) and (5.9) below for MFS-D and MFS-P, respectively.

$$\frac{w_i}{\sum_{j=1}^{n} w_j} = E\left[\frac{X_i(k)}{\sum_{j=1}^{n} X_j(k)}\right]. \tag{5.8}$$

$$Pr\left(\left|E\left[\frac{X_i(k)}{\sum_{j=1}^{n} X_j(k)}\right] - w_i\right| > 0.5\delta\phi_i\right) = P_\delta. \tag{5.9}$$

5.1.2 Motivation

ORCA-SRT is able to calculate the optimal channel allocation for a fair scheduling problem in a single-rate TDMA network. For a multi-rate TDMA network, the opportunistic scheduling algorithm in [42] was designed to ensure temporal fairness and to maximize overall throughput stochastically. However, both ORCA-SRT and the algorithm in [42] do not take throughput fairness into account.

MFS, on the other hand, was designed for throughput-fair scheduling in a multi-rate CDMA network. Nevertheless, MFS does not consider temporal fairness. Also, the greedy algorithm and the stochastic approximation used in MFS may result in a suboptimal solution and slow convergence rate for the algorithm. Again, MFS assumes that there are unlimited available codes and that they are perfectly orthogonal to each other. Therefore, the actual throughput might not always be as high as the throughput reported for MFS.

The primary objectives of the proposed ORCA-MRT framework are to ensure frame-based temporal fairness (i.e., all mobiles acquire temporal-fair share in every frame) and to bound the inter-access delay. Subject to these two constraints, the secondary objective is to maximize overall throughput without deteriorating throughput fairness. Such a combined temporal-throughput fair scheduling would be useful especially in situations where bounded delay is of utmost importance such as in the case of applications running *Transmission Control Protocol (TCP)* to avoid TCP timeouts, or in a real-time application where data packets become useless after some time.

In ORCA-MRT, the optimization problem is formulated by (4.5)-(4.8). The nature of a TDMA network and the primary objectives are realized by the hard constraints

(4.7) and (4.8) of the assignment problem. The secondary objective is achieved by designing appropriate cost function and minimizing the corresponding total cost (C_T) in (4.5).

5.2 System Model and Architecture of ORCA-MRT Scheduler

The general operation of ORCA-MRT[2] is illustrated in Figure 5.1.

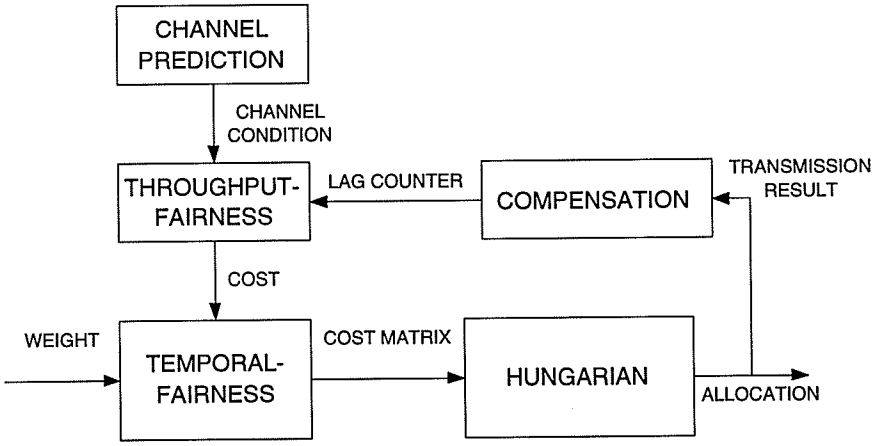


Figure 5.1. *Architecture of the ORCA-MRT scheduler.*

5.2.1 Channel Prediction Block

In a multi-rate transmission environment, it is assumed that the channel condition is quantized into M levels. The channel variation among these M levels can be modeled by an FSMC as explained in Section 2.3.3. Assuming that the parameters of FSMC are known, the predicted channel state (m'_{ij}) of mobile i in slot j can be calculated based on the following channel prediction models:

- *Perfect channel prediction:*

$$m'_{ij} = m_{ij}, \tag{5.10}$$

[2]Again, ORCA-MRT is implemented at the base station.

where m_{ij} is the real channel state.

- *Simulation-based channel prediction*: The last known state for mobile i, m_{i0} (i.e., the channel state of the last time slot in the previous frame) is utilized as an initial state in the FSMC to generate simulated channels $m'_{ij}(j = \{1, \cdots, T_F\})$, where T_F is scheduling frame size.
- *Expectation-based channel prediction*: The expected channel state for mobile i in time slot $k + t$ given that m_{ik} is known ($E[\mathscr{M}_t|m_{ik}]$) can be calculated as follows:

$$m'_{i,k+t} = E[\mathscr{M}_t|m_{ik}] = \sum_{l=1}^{M} l \cdot p^{(t)}_{(m_{ik})l}, \tag{5.11}$$

where $p^{(t)}_{(m_{ik})l}$ (calculated using (2.9)) is the probability that state m_{ik} will be in state l in the next t time slots. For ORCA-MRT, $k = 0$ and $t = \{1, \cdots, T_F\}$.

5.2.2 Throughput-Fairness Block

ORCA-MRT does not use explicit compensation to avoid delay and temporal-fairness degradation (see OBSERVATION 4.2 and 4.3). Instead, it makes use of the throughput-fairness block to give favor to mobiles which are lagging and/or perceiving good channel condition. This process is based on the cost matrix which is determined by exploiting the nature of the assignment problem, channel condition, and a lag counter.

5.2.2.1 Nature of the Assignment Problem

To minimize total cost (C_T) in (4.5), we set $x_{ij} = 1$ where c_{ij} is minimum. When each mobile does not experience its minimum cost in the same slot, the problem is fairly simple in that the scheduler selects the mobile with minimum transmission cost in each slot. Hereafter, we will consider only non-trivial cases where each mobile experiences equal minimum cost in the same slot. When several mobiles experience minimum transmission cost in the same slot (which is called the *contending slot*), they must contend for the possession of that slot (because of the constraint (4.7)).

Observation 5.1 (THE RELATIONSHIP BETWEEN COST MATRIX AND THE SOLUTION OF THE ASSIGNMENT PROBLEM): *Let j be the contending slot in which a pair*

of mobiles (h and i) perceives the same and minimum transmission cost. Flow i will acquire slot j, if $c_{ik} > c_{hk}, \forall k \neq j$.

PROOF: *Let row 1 and 2 represent transmission cost of mobile h and i, respectively. Let column* $j = 1$ *be the contending slot and let k be the second column in the sub-matrix* $\mathbf{C}'$ *residing in any cost matrix. Therefore,* $\mathbf{C}' = \begin{pmatrix} c & (c+H) \\ c & (c+I) \end{pmatrix}$, *where c, H, and I are positive integers. If* $I > H$, *the contending slot* $j = 1$ *will be given to mobile* $i = 2$ *and the solution* $\{x_{12} = 1, x_{21} = 1\}$ *will incur lower total cost* $(C_T = 2c + H)$ *by the amount of* $I - H$ *than* $\{x_{11} = 1, x_{22} = 1\}$ *will* $(C_T = 2c + I)$. ■

5.2.2.2 Channel Condition

A channel-aware the cost function could be obtained by generalizing the cost function of ORCA-SRT in (4.9) as follows:

$$c_{ij}^{CA} = M - m'_{ij}. \tag{5.12}$$

The channel-aware cost (c_{ij}^{CA}) is a linearly-decreasing function in m'_{ij}. When the channel becomes worse, m'_{ij} is smaller and c_{ij}^{CA} increases.

5.2.2.3 Lag Counter

A lagging mobile is allowed to transmit in a slot with good condition, even though other mobiles in the same slot perceive the same or better channel. Based on (5.12), we revise the cost function by taking into account the lag counter.

Observation 5.2 (ADDITIVE COST FUNCTION W.R.T. LAG COUNTER): *The cost function cannot favor any lagging mobile by just adding/subtracting a lag counter to all the elements in each row.*

PROOF: *Let K be an integer. By adding a constant K to all elements, the minimum value in row i becomes* $min_i + K$. *After step 1 in the Hungarian method,* $c_{ij} = c_{ij}^{CA} + K - (min_i + K) = c_{ij}^{CA} - min_i$. *Therefore, the transmission cost is not affected by adding a constant.* ■

The cost function intuitively revised by decreasing the cost of a lagging mobile to increase the probability to transmit at a faster rate,

$$c_{ij} = c_{ij}^{CA} - L_i, \tag{5.13}$$

where $L_i \in \mathbb{I}^{0+}$ is a lag counter of mobile i, will not work.

To prevent min_i from being subtracted, the cost function could be modified to obtain a constrained increasing cost function (w.r.t. the lag counter) as follows:

$$c_{ij} = c_{ij}^{CA} + L_i \qquad s.t. \qquad j \neq \arg_l \ min\{c_{il}^{CA}\}. \tag{5.14}$$

Observation 5.3 (CONSTRAINED INCREASING COST FUNCTION): *Let mobile h and i have lag counters of L_h and L_i, respectively. Eq. (5.14) will favor mobile i only in time slot j, if j is a contending slot and $\Delta L_{ih} > \Delta c_{hi}(k), \forall k \neq j$, where $\Delta L_{ih} = L_i - L_h$ and $\Delta c_{hi}(k) = c_{hk}^{CA} - c_{ik}^{CA}$.*

PROOF: *Let h and i are inserted in the first and second rows of the sub-matrix $\mathbf{C}'$ and let j and k represent the first and second columns in the matrix, respectively. After applying (5.14), $\mathbf{C}' = \begin{pmatrix} c & (c+H+L_h) \\ c & (c+I+L_i) \end{pmatrix}$, where H and I are positive integers. From* OBSERVATION *5.1, the first slot (j) is given mobile i, if $(c+I+L_i) > (c+H+L_h)$, i.e., $\Delta L_{ih} > \Delta c_{hi}(k), \forall k \neq j$.* ∎

Consider mobile g, h, i in $1^{st}, 2^{nd}, 3^{rd}$ row of the following sub-matrix (after applying (5.14)):

$\mathbf{C}' = \begin{pmatrix} c & \mathbf{c_{gk} + L_g} & \mathbf{c_{gl} + L_g} \\ c & \mathbf{c_{hk} + L_h} & \mathbf{c_{hl} + L_h} \\ c & c_{ik} + L_i & c_{il} + L_i \end{pmatrix}$. Assume $\Delta L_{ig} > \Delta c_{gi}(k)$ and $\Delta L_{ih} > \Delta c_{hi}(k), \forall k \neq 1$. The best slot with the cost of c will be given to mobile i, rather than g or h. After the first column is allocated to mobile i, the first column and third row can be removed from $\mathbf{C}'$. The problem now reduces to the upper-right sub-matrix of $\mathbf{C}'$ (marked by bold-face letters). All the elements in each row in the reduced sub-matrix is different from the channel-aware cost by an additive constant. Therefore, the scheduler does not favor any lagging mobile (see OBSERVATION 5.2).

Observation 5.4 (KEY FACTOR IN COST FUNCTION): *Only larger cost step size* ($\Delta^{(m')}$), *where*

$$\Delta^{(m')} = c(m'-1) - c(m'), \tag{5.15}$$

can favor lagging mobiles.

PROOF: *By applying (5.12), (5.14), and (5.15), the cost step size becomes*

$$\Delta^{(m')} = \begin{cases} (M-(m'_{ij}-1)) - (M-m'_{ij}+L) = L+1, & c^{CA}(m'_{ij}-1) = c^{CA}_{min}, \\ (M-(m'_{ij}-1)+L) - (M-m'_{ij}+L) = 1, & otherwise. \end{cases} \tag{5.16}$$

From OBSERVATION *5.3, the scheduler will favor a lagging mobile when* $\Delta^{(m')} \neq 1$, *or only in a contending slot where* $c^{CA}(m'_{ij}-1) = c^{CA}_{min}$ *and does not favor a lagging mobile with all other states, where* $(\Delta^{(m')} = 1)$. ■

From OBSERVATION 5.4, if $\Delta^{(m')} = L+1$ $(\forall m')$, the cost function would intuitively be able to favor a lagging mobile for all the states. To achieve this, the cost function is defined as follows:

$$c_{ij} = c^{CA}_{ij} \cdot (L_i + 1). \tag{5.17}$$

Observation 5.5 ($\mathbf{C}_{MRT}$): *The cost function defined in (5.17) is able to favor a lagging mobile in every state. The amount of favor is proportional to the lag counter* L_i.

PROOF: *From (5.15) and (5.17), we observe that*

$$\Delta^{(s)} = [(M-(m_{ij}-1)) - (M-m_{ij})] \cdot (L_i+1) = L_i + 1.$$

Since $\Delta^{(s)}$ *is a linear function in* L_i, *the amount of favor given to a lagging mobile is a linearly increasing function of the lag counter. Consider* $\mathbf{C}'$ *given previously in* OBSERVATION *5.3. Let us assume again that column 1 is given to mobile i in row 3. After applying (5.17) the upper-right sub-matrix will be*

$$\begin{pmatrix} c_{gk} + c_{gk}(1+L_g) & c_{gl} + c_{gl}(1+L_g) \\ c_{hk} + c_{hk}(1+L_h) & c_{hl} + c_{hl}(1+L_h) \end{pmatrix}.$$

Each row is not modified by just adding a constant, but a constant that is scaled with the current transmission cost. Therefore, the cost function in (5.17) favors a lagging mobile in all slots. ■

5.2.3 Temporal-Fairness Block

Like that of ORCA-SRT, scheduling frame size in ORCA-MRT is $T_F = \sum_i w_i$ time slots. For each mobile i with weight w_i, the temporal-fairness block calculates a row vector of c_{ij} (derived from (5.17) in the throughput-fairness block) with the length of T_F, and inserts into the cost matrix w_i identical rows where a set of rows belonging to mobile i is denoted by $\mathbf{R}_i = \{R_1, R_2, \cdots, R_{w_i}\}$.

Observation 5.6 (PROPERTIES OF TEMPORAL-FAIRNESS BLOCK): *Regardless of channel condition and/or lag counters, the temporal-fairness block ensures that*

1. *Flow i transmits in exactly w_i time slots in a scheduling frame.*
2. *The maximum inter-access delay for mobile i is bounded by*

$$\max_{\forall k}\{\mathscr{D}_{acc_i}(k)\} = 2\sum_{\forall j \neq i} w_j + 1. \tag{5.18}$$

3. *The maximum inter-access delay of each mobile increases by $2w$, when a mobile with the weight of w becomes active.*

PROOF: *ORCA-MRT utilizes the Hungarian method and therefore inherits all the properties of the assignment solution (see* OBSERVATION *4.2). By inserting w_i rows into the cost matrix, mobile i is allocated exactly w_i time slots. In two scheduling frames, mobile i receives $2w_i$ time slots. The maximum inter-access delay occurs when the allocation is clustered at the beginning of the first frame and at the end of the second frame. In such the case, the maximum inter-access delay of* $\max_{\forall k}\{\mathscr{D}_{acc_i}(k)\} = 2(T_F - w_i) + 1 = 2\sum_{\forall j \neq i} w_j + 1$. *It can easily be observed that* $\max_{\forall k}\{\mathscr{D}_{acc_i}(k)\}$ *increases by $2w$, if the sum of all the weights increases by w.* ∎

5.2.4 Hungarian Block

The Hungarian block receives the cost matrix from the temporal-fairness block and uses the algorithm presented in Appendix F to solve the assignment problem. In column (or time slot) j, mobile i will transmit with the rate of

$$r_{ij} = \begin{cases} r_{min} + m_{ij} - 1, & x_{hj} = 1 \; \forall h \in \mathbf{R}_i, \\ 0, & otherwise, \end{cases} \tag{5.19}$$

where r_{min} is the minimum number of packets that each mobile can transmit in a particular time slot (when the channel state is worst), and again $\mathbf{R}_i$ is the set of rows belonging to mobile i. We assume that channel state is always known right before the transmission. The transmission rate is therefore calculated based on the actual channel rather than the predicted channel.

5.2.5 Compensation Block

Due to the channel variation (as discussed in Section 2.3.3), each mobile might be allocated different transmission rates. The resulting unfairness is monitored by the compensation block via lag counters. The lag counter of mobile i (L_i) is calculated as follows:

$$L_i = \max_{\forall k} \left\{ \sum_{j=1}^{ts} \frac{r_{kj}}{w_k} \right\} - \sum_{j=1}^{ts} \frac{r_{ij}}{w_i}, \tag{5.20}$$

where ts is the current time slot and r_{ij} is the number of packets that mobile i transmits successfully in time slot j.

5.2.6 Summary

During each scheduling frame, the ORCA-MRT-based scheduler performs the following procedures:

- **Step 1:** Channel prediction block predicts the channel condition in the next scheduling frame.
- **Step 2:** Based on the predicted channel condition and the lag counter, the throughput-fairness block calculates transmission cost, by using (5.17).
- **Step 3:** Temporal-fairness block receives the transmission cost from the throughput-fairness block and constructs *Multi-Rate Transmission Cost Matrix* ($\mathbf{C}_{MRT}$). For mobile i, w_i identical rows (each with the size of T_F time slots) of transmission costs are inserted into the cost matrix.
- **Step 4:** Hungarian block solves the assignment problem based on the input cost matrix and gives the allocation to each mobile.
- **Step 5:** Compensation block observes the transmission result and calculates lag counters as a function of the observed transmission result.

5.3 Simulation Environment

5.3.1 Performance Measures

We define the following weight-independent performance metrics which are similar to those in [42]:

- **Normalized throughput for mobile** i ($\gamma_{mob}(i)$):

$$\gamma_{mob}(i) = \frac{1}{T_{ob}} \cdot \sum_{j=1}^{T_{ob}} \frac{r_{ij}}{w_i}, \tag{5.21}$$

- **Normalized inter-access delay for mobile** i ($\mathscr{D}_{acc_i}$):

$$\mathscr{D}_{acc_i} = \frac{1}{K_i} \cdot \sum_{k=1}^{K_i} \left(\frac{w_i}{\sum_{j=1}^{n} w_j} \cdot \mathscr{D}_{acc_i}(k) \right), \tag{5.22}$$

where r_{ij} is the transmission rate given to mobile i in time slot j, w_i is the weight of mobile i, K_i is the number of transmission opportunities of mobile i over an observation period (T_{ob}), and $\mathscr{D}_{acc_i}(k)$ is inter-access delay of mobile i from $(k-1)^{th}$ to k^{th} transmission opportunities. Note that, both $\gamma_{mob}(i)$ and $\mathscr{D}_{acc_i}$ are normalized and averaged over a period of T_{ob} such that they are independent of mobiles' weights.

We measure throughput and throughput-fairness by calculating the average value of $\gamma_{mob}(i)$, denoted by $E[\gamma_{mob}]$, and the *Standard Deviation (S.D.)* of $\gamma_{mob}(i)$, denoted by $\sigma(\gamma_{mob})$. Similarly, the delay and the temporal-fairness are measured by the average value of $\mathscr{D}_{acc_i}$, denoted by $E[\mathscr{D}_{acc}]$, and the standard deviation $\mathscr{D}_{acc_i}$, denoted by $\sigma(\mathscr{D}_{acc})$. Before proceeding to the simulation, the following observations are noteworthy:

1. $E[\gamma_{mob}] \leqslant M/T$, where M is the number of channel states. Higher $E[\gamma_{mob}]$ implies better performance in terms of throughput.
2. Under perfectly fair time slot allocation, $E[\mathscr{D}_{acc}] = 1$.
3. Smaller values of $\sigma(\gamma_{mob})$ imply better throughput-fairness performance and smaller values of $\sigma(\mathscr{D}_{acc})$ imply better temporal-fairness performance.

5.3.2 Simulation Parameters and Methodology

MFS allows several mobiles to transmit at the same time, while ORCA-MRT permits only one mobile to transmit. By ignoring the noise term and setting $d = 3$ as in

[42] , the channel condition calculated from (5.5) varies approximately in the range $[0.5-0.3, 0.5+0.3] = [0.2, 0.8]$. Due to the power constraint in (5.3) with $P = 2$, the overall transmission rate at an instant is in the set of $r = \{2, 3, \cdots, 10\}$ and has the average value (r_{avg}) of 4.

Assuming equally likely transmission probability in each state and setting r_{min} to 2 packets, we observe that a channel with $r = \{2, 3, 4, 5, 6\}$ (5-FSMC) has $r_{avg} = 4$ which is equal to r_{avg} of MFS. We run simulations for 2000 time slots (as suggested in [42]) for 3-FSMC ($r = \{2, 3, 4\}$), 5-FSMC ($r = \{2, 3, \cdots, 6\}$), and 7-FSMC ($r = \{2, 3, \cdots, 8\}$) with equally-likely steady state probability. We vary ρ in the range of $\{0.5, 0.6, 0.7, 0.8, 0.9\}$. Unless otherwise specified, perfect channel prediction is assumed to eliminate the effect of prediction inaccuracy on the performance evaluation.

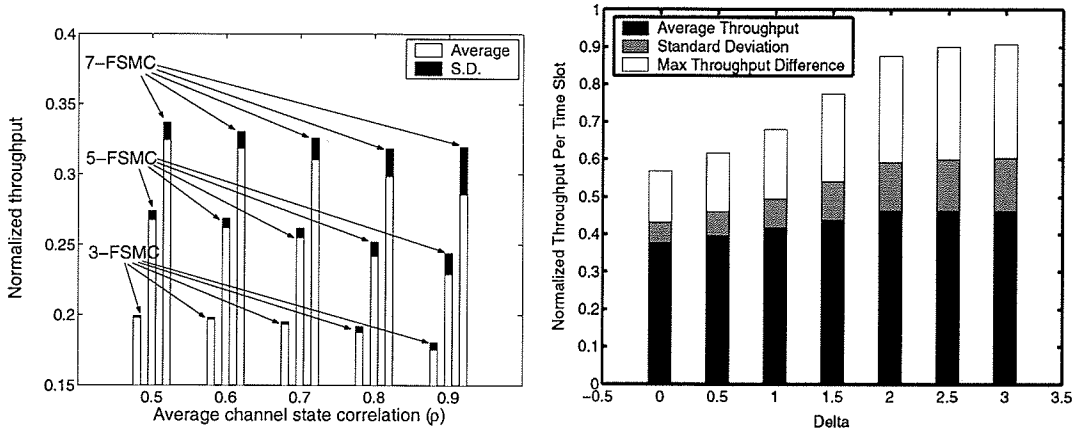
5.4 Simulation Results and Discussions

5.4.1 Performance of ORCA-MRT and MFS

As in MFS, the number of active mobiles is set to 16: 12 mobiles, each with weight of 1 and 4 mobiles, each with weight of 2. The average value and standard deviation of the normalized throughput among all the mobiles are shown in Figure 5.2.

Throughput of MFS ranges from 0.35 to 0.45 depending on the service discrepancy (δ), while ORCA-MRT has throughput in the range of [0.17,0.33]. However, with MFS the standard deviation of throughput $\sigma(\gamma_{mob})$ is in the range [0.05,0.11], while with ORCA-MRT $\sigma(\mathscr{D}_{acc})$ is in the range [0.006,0.03]. ORCA-MRT achieves better throughput-fairness performance at the expense of throughput degradation.

High throughput in MFS is achieved by allowing several mobiles to transmit simultaneously, under the assumption that interference among all the mobiles is negligible. As the transmission rate increases (e.g., increasing number of code channels in a CDMA system), the interference becomes more severe and cannot be ignored. ORCA-MRT, on the other hand, accommodates only one mobile per time slot. Therefore, the interference among all the mobiles in the same cell is alleviated. Obviously, the reduction in throughput occurs as a nature of a TDMA-based approach. Note that, although performing well in terms of throughput, MFS can neither ensure temporal



(a) (b)

Figure 5.2. *Average and standard deviation of normalized throughput for (a) ORCA-MRT and (b) MFS.*

fairness nor bound inter-access delay.

5.4.2 Channel State Correlation

As can be seen from Figure 5.2, throughput and throughput-fairness improve as channel states become less correlated (small ρ). When the channel-state correlation is small, a mobile does not to stay in the bad state for a long period of time. The scheduler is able to select the most suitable state and therefore is capable of improving performance in terms of both throughput and throughput-fairness.

Figure 5.3 reveals that delay and temporal-fairness performances are fairly good: $E[\mathscr{D}_{acc}] \approx 1$ and $\sigma(\mathscr{D}_{acc}) < 0.005$ under all channel conditions. ORCA-MRT shows delay robustness in that both $E[\mathscr{D}_{acc}]$ and $\sigma(\mathscr{D}_{acc})$ are not affected by the channel condition.

5.4.3 Channel Prediction

In this section, we investigate the effect of three channel prediction models on system performance.

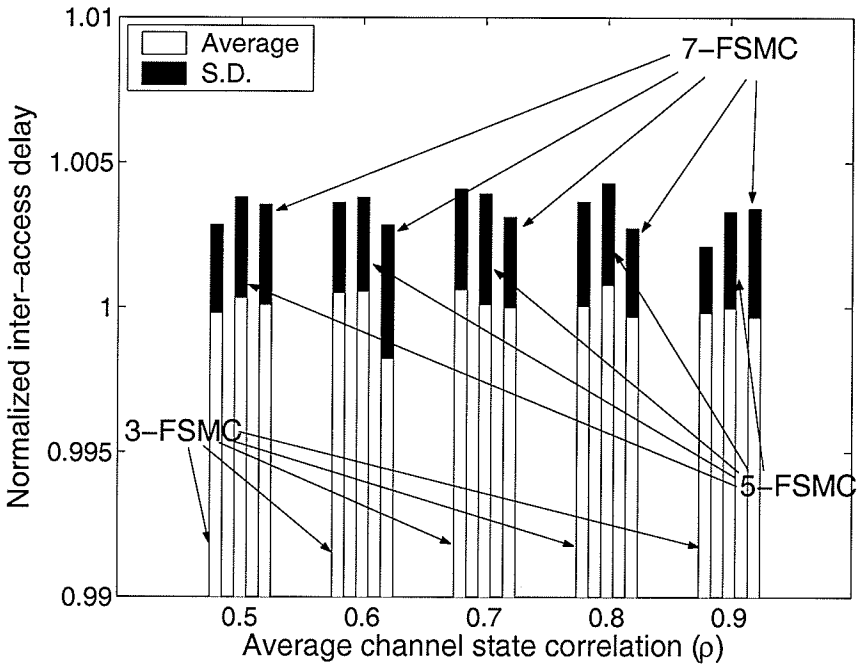


Figure 5.3. *Average and standard deviation of normalized inter-access delay.*

Definition 5.1 AVERAGE PREDICTION ERROR ($\Delta(t)$) *is the mean absolute difference between predicted channel and actual channel states in time slot t. Mathematically,*

$$\Delta(t) = \sum_{i=1}^{M} \pi_i \sum_{m'=1}^{M} \sum_{m=1}^{M} |m - m'| \cdot f^{(t)}_{\mathcal{M},\mathcal{M}'}(m, m'|i), \tag{5.23}$$

where $f^{(t)}_{\mathcal{M},\mathcal{M}'}(m, m'|i)$ is the joint probability that in time slot t the actual state is m and the predicted state is m' given that the channel state in time slot 0 is i. □

Theorem 5.1 (Average prediction error) *For an* FSMC *model where the channel state at $t = 0$ is known, average prediction error in time slot t for expectation-based and simulation-based prediction models can be calculated from (5.24) and (5.25), respectively,*

$$\Delta_E(t) = \sum_{i=1}^{M}\sum_{m=1}^{M} \pi_i \cdot |m - E[\mathscr{M}_t|i]| \cdot p_{i,m}^{(t)}, \tag{5.24}$$

$$\Delta_S(t) = 2\sum_{i=1}^{M}\sum_{\tau=1}^{M-1}\sum_{m=1}^{M-\tau} \pi_i \cdot \tau \cdot p_{i,m}^{(t)} \cdot p_{i,m+\tau}^{(t)}, \tag{5.25}$$

where the prediction length t is the interval (in terms of time slots) between the last known channel state and the predicted state, $E[\mathscr{M}_t|i]$ is the predicted channel state in time slot t obtained from the expectation-based prediction model (using (5.11)), and $p_{i,m}^{(t)}$ is the probability that state i will move to state m in t steps.

In an RSC, where the channel state depends only on $\pi_i(\forall i)$, the average prediction error for expectation-based and simulation-based models can be calculated from (5.26) and (5.27), respectively.

$$\Delta_E^{rand}(t) = \sum_{m=1}^{M} \pi_m \cdot |m - \sum_{i=1}^{M} i \cdot \pi_i|. \tag{5.26}$$

$$\Delta_S^{rand}(t) = 2\sum_{\tau=1}^{M-1}\sum_{m=1}^{M-\tau} \tau \cdot \pi_m \cdot \pi_{m+\tau}. \tag{5.27}$$

□

PROOF: *See Appendix G.* ■

Figure 5.4 shows the effect of prediction length on average prediction error when $M = 5$ and $\rho = \{0.5, 0.9\}$. We observe that in all the cases considered, the expectation-based channel prediction model always has smaller prediction error than the simulation-based model. Also, an RSC model leads to the upper-bound (worst-case) of prediction error, since the current channel state does not provide information about the channel state in the future. For small prediction length, the possible range of predicted states is limited (e.g., 3 possible states when $t = 1$) and the prediction is less likely to be erroneous. As the prediction length becomes larger, the possible values of the predicted states increase, the FSMC behaves more randomly, and the prediction becomes less accurate. When the prediction length is sufficiently long, the prediction of the FSMC converges to that of the RSC which provides the upper-bound for prediction error.

We observe from Figure 5.4 that the average prediction error for both expectation-based and simulated-based models converges to $\Delta_E^{rand}(t) = 1.2$ and $\Delta_S^{rand}(t) = 1.6$

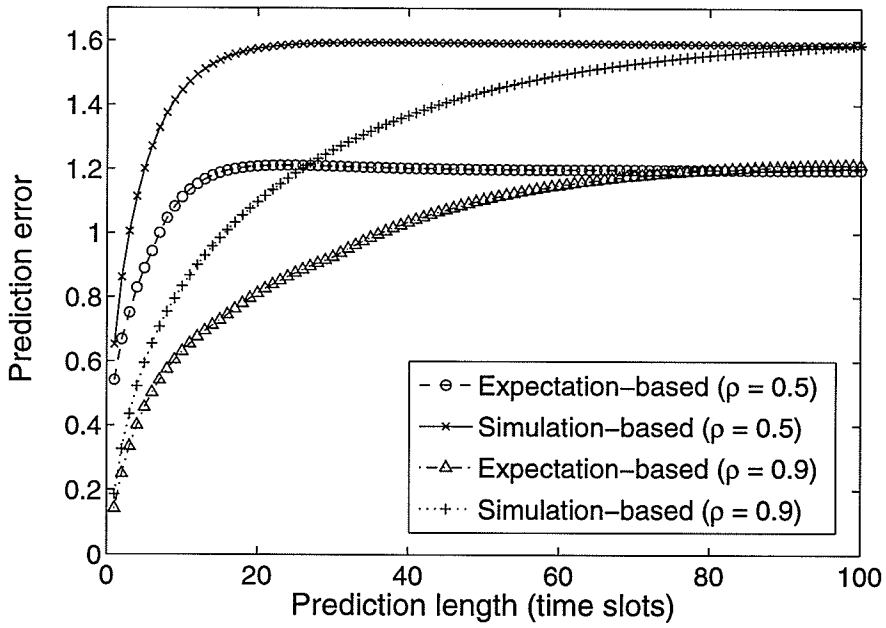


Figure 5.4. *Average prediction error vs. prediction length.*

calculated from (5.26) and (5.27), respectively. We also observe that increased channel state correlation leads to better prediction accuracy because the channel tends to stay in the same state and the predicted channel states become more similar to the actual channel states. Therefore, for high channel state correlation, the prediction error slowly converges to the worst-case error.

Figure 5.5 plots normalized throughput obtained from an ORCA-MRT scheduler under different channel prediction models with $M = 5$ and $\rho = \{0.5, 0.7, 0.9\}$. We observe that the simulation-based channel prediction model leads to inferior throughput and throughput fairness performances due to prediction inaccuracy. As channel states become more correlated, the prediction model performs better, and the performance of ORCA-MRT with the simulation-based model becomes closer to that with the perfect channel prediction model.

The expectation-based model performs fairly good in that it almost results in the same throughput and throughput fairness as those obtained when the channel condition is known ahead of time. The performances in terms of delay and temporal fairness are very similar for all the prediction models (the results are omitted for

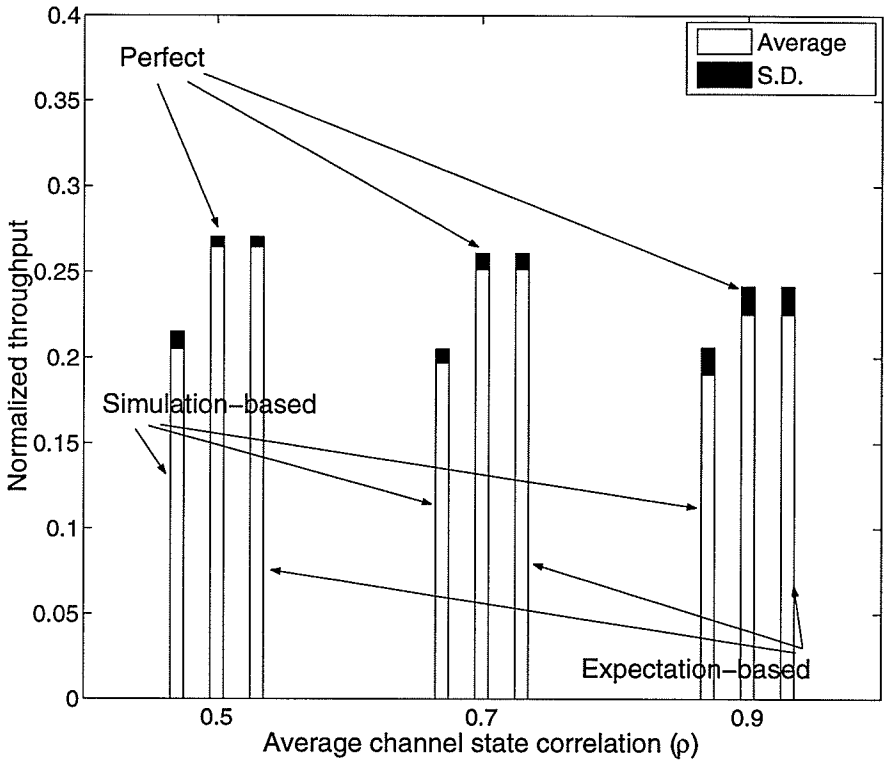


Figure 5.5. *Effect of imperfect channel prediction.*

brevity).

5.4.4 Upper-bound and Lower-bound for Throughput

Consider a situation where all mobiles have equal weights and equal lag counter, and experience the best channel state strictly in different slots. In this case, the solution is the same as that obtained from the channel-quality based opportunistic scheduling (see Chapter 3) where the mobile perceiving the best channel condition in each slot is allowed to transmit. This case therefore provides the upper-bound for overall throughput. However, if this is not the case, each mobile might not get its best allocation because of (4.7) and (4.8), and the overall throughput will drop accordingly.

Definition 5.1 TRANSMISSION STATE PROBABILITY $(f_{\mathcal{M}}(m))$ *is the probability that*

a particular mobile is allowed to transmit when channel is in state m. □

From THEOREM 3.1 and COROLLARY 3.1,

$$f_{\mathscr{M}}(m) = (F_m)^{T_F} - (F_{m-1})^{T_F}, \tag{5.28}$$

$$E[\gamma_{ub}] = \frac{1}{T_F} \cdot \left(r_{min} - 1 + M - \sum_{m=1}^{M-1} (F_m)^{T_F} \right), \tag{5.29}$$

where $(E[\gamma_{ub}])$ is an upper-bound throughput obtained from a channel-quality-based opportunistic scheduling, M is the number of channel states, π_m is steady state probability that the channel is in state m, and $F_m = \sum_{i=1}^{m} \pi_m$ is the *Cumulative Distribution Function (cdf)* of the channel state. In case of an equally likely channel model, where $\pi_m = 1/M, \forall m$,

$$f_{\mathscr{M}}(m)_{eq} = \frac{m^{T_F} - (m-1)^{T_F}}{M^{T_F}}, \tag{5.30}$$

$$E[\gamma_{ub,eq}] = \frac{1}{T_F} \cdot \left(r_{min} - 1 + M - \frac{\sum_{m=1}^{M-1} m^{T_F}}{M^{T_F}} \right). \tag{5.31}$$

Furthermore, in case of a round-robin-based scheduler, where each mobile acquires a channel access in order,

$$f_{\mathscr{M}}(m)_{\pi} = \pi_m, \tag{5.32}$$

$$E[\gamma_{\pi}] = \frac{1}{T_F} \cdot \left(r_{min} - 1 + \sum_{m=1}^{M} m \cdot \pi_m \right). \tag{5.33}$$

We plot both $E[\gamma_{ub,eq}]$ and $E[\gamma_{\pi,eq}]$ ($E[\gamma_{\pi}]$ when the channel states are equally likely) as well as the average throughput obtained from the simulation in Figure 5.6. All the parameter settings is the same as that in Figure 5.2, $\rho = 0.7$, and numbers of channel states are 3, 5, and 7. We can observe that $E[\gamma_{ub,eq}]$ and $E[\gamma_{\pi,eq}]$ provide upper-bound and lower bound for the average normalized throughput respectively.

We also run the simulation in 3-FSMC, 5-FSMC, and 7-FSMC equally-likely channels with $\rho = 0.7$ and plot the values of $f_{\mathscr{M}}(m)_{eq}$ in Figure 5.7. We compare the results from ORCA-MRT ($f_{\mathscr{M}}(m)_{eq}$:ORCA-MRT) with those from (5.30) ($f_{\mathscr{M}}(m)_{eq}$:UB). Figure 5.7 reveals that both upper-bound and simulated values of $f_{\mathscr{M}}(m)_{eq}$ are shifted

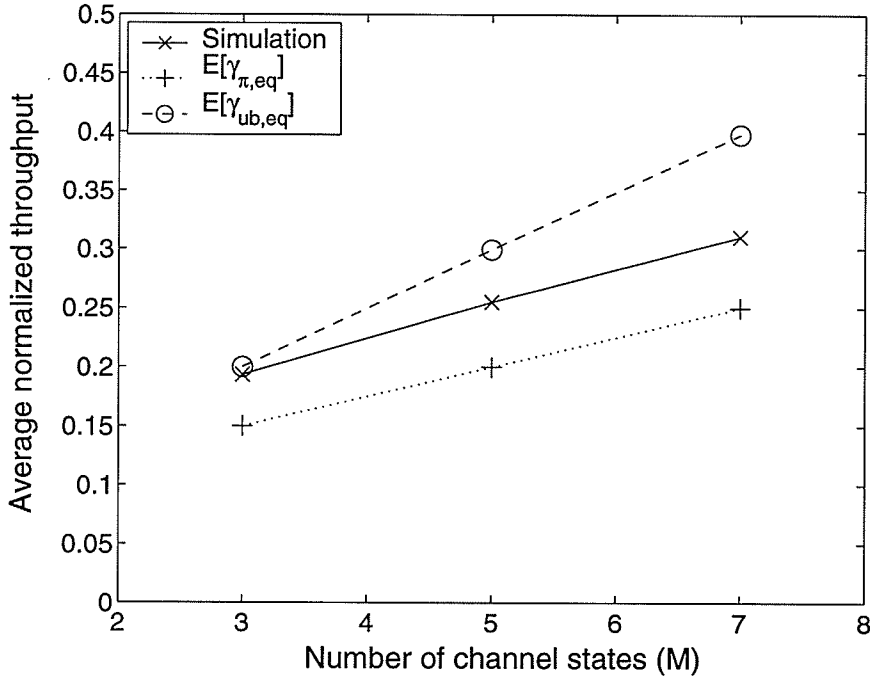


Figure 5.6. *Upper-bound and lower-bound of average normalized throughput.*

(from $\boldsymbol{\pi}$) towards the best channel state. Due to fairness and interference-free constriants in (4.7) and (4.8), some mobiles experiencing good channel conditions might not be able to transmit at good states. Therefore, the average throughput measured from ORCA-MRT is always lower than the upper-bound throughput in (5.29) and higher than the lower-bound in (5.33).

5.4.5 Number of FSMC States (M)

5.4.5.1 Throughput Performance

From Figure 5.7, $f_{\mathcal{M}}(m)$ tends to cluster around higher states. Therefore, increasing number of FSMC states increases average throughput, as can be observed from Figure 5.6, and in (5.29) and (5.33) with increasing M.

The shape of $f_{\mathcal{M}}(m)$ obtained from simulation becomes less similar to the upper-bound $f_{\mathcal{M}}(m)$, as the number of channel states increases. As a result, the performance in terms of throughput diverges from the upper-bound as the number of FSMC states

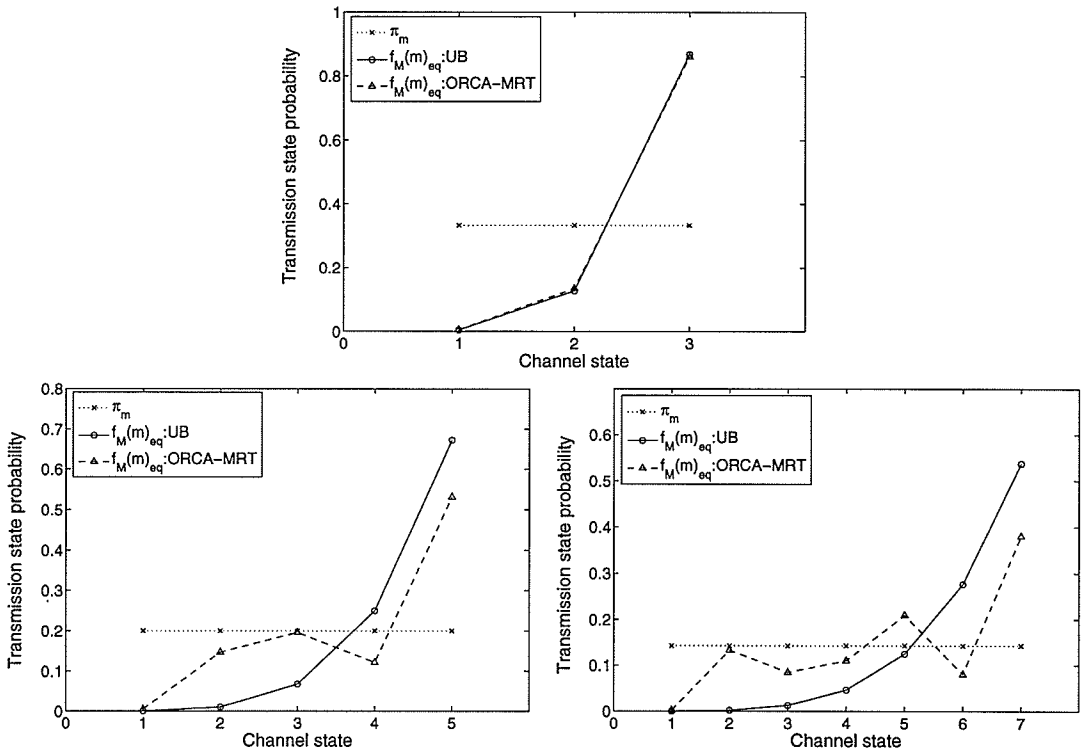


Figure 5.7. *Transmission state probability.*

increases (Figure 5.6).

5.4.5.2 Fairness Performance

There are two key factors related to throughput-fairness: equality in transmission rate and perfect compensation mechanism. When $f_{\mathcal{M}}(i) = 1$ and $f_{\mathcal{M}}(j) = 0$ $(\forall j \neq i)$, all mobiles transmit only at rate i. In this case, there will not be any unfairness. If there exists inequality in the transmission rate and the compensation mechanism is able to perfectly compensate for unequal allocation, there will be no unfairness.

In case of equally likely channels, each with M states, the probability that all n mobiles perceive the same channel state during a scheduling frame is

$$P_{\text{equal rate}}(M) = \sum_{m=1}^{M} n! \left(\frac{1}{M}\right)^n = n! \left(\frac{1}{M}\right)^{n-1}. \tag{5.34}$$

If the number of channel states is increased by a positive integer δ, $P_{\text{equal rate}}(M+\delta) =$

$n!\left(\frac{1}{M+\delta}\right)^{n-1}$. Therefore, the relative incremental probability is

$$P_{inc} = \frac{P_{\text{equal rate}}(M+\delta)}{P_{\text{equal rate}}(M)} = \left(\frac{M}{M+\delta}\right)^{n-1}. \tag{5.35}$$

Since $P_{inc} < 1$, increasing the number of channel states ($\delta = \{1, 2, \cdots\}$) reduces the probability that all mobiles will be allocated the same transmission rate during a scheduling time frame.

The compensation method in ORCA-MRT is non-aggressive, in which each mobile must transmit at least one packet per scheduling frame. The maximum number of compensation for a mobile with the weight one is limited to $M-1$ packets per scheduling frame. Compared to the compensation mechanism, the inequality in transmission rate has a stronger influence on fairness performance. Therefore, throughput-fairness in ORCA-MRT degrades as M increases (Figure 5.2). Again, average inter-access delay and temporal fairness are not affected by the channel parameters (Figure 5.3).

5.4.6 Weights of the Mobiles

To investigate the effect of the weights of mobiles on system performance, we assume 5 mobiles with the weights of 1, 2, 3, 4, and 5, and run the simulation on a 5-FSMC channel with $\rho = 0.7$. In this case, the scheduling frame size (T_F) is 15 time slots.

Table 5.1. *Effect of weights of the mobiles.*

mobile	1	2	3	4	5
w_i	1	2	3	4	5
$\max_{sim:\forall k}\{\mathscr{D}_{acc_i}(k)\}$	29	27	25	23	21
$\max_{\forall k}\{\mathscr{D}_{acc_i}(k)\}(5.18)$	29	27	25	23	21
$E[\mathscr{D}_{acc}]$	15.01	7.50	5.00	3.74	3.00
$\mathscr{D}_{acc_i}$	1.0	1.0	1.0	1.0	1.0
$w_i \cdot E[\gamma_{mob}(i)]$	0.228	0.527	0.671	1.104	1.23

Table 5.1 shows that the maximum inter-access delay measured from simulation ($\max_{sim:\forall k}\{\mathscr{D}_{acc_i}(k)\}$) is bounded by the value of $\max_{\forall k}\{\mathscr{D}_{acc_i}(k)\}$ calculated from (5.18). We can observe that the average inter-access delay of mobile i, $E[\mathscr{D}_{acc}]$ is proportional to w_i. Equivalently, the value of normalized inter-access delay approaches

that in the ideal fairness condition ($\mathscr{D}_{acc_i} = 1$). Also, we can observe that the actual throughput ($\gamma_{mob}(i) \cdot w_i$) is proportional to the weight of each mobile.

5.4.7 Number of Mobiles and Scheduling Frame Size

In this section, we assume that all the mobiles have equal weight. The number of mobiles is varied in the set of $\{5, 10, 15, 20, 25, 30\}$. We conduct the experiment on a 5-FSMC channel with $\rho = 0.7$. The scheduling frame size is calculated by $T_F = \sum_{i=1}^{n} w_i = \{5, 10, 15, 20, 25, 30\}$, where n is the number of mobiles.

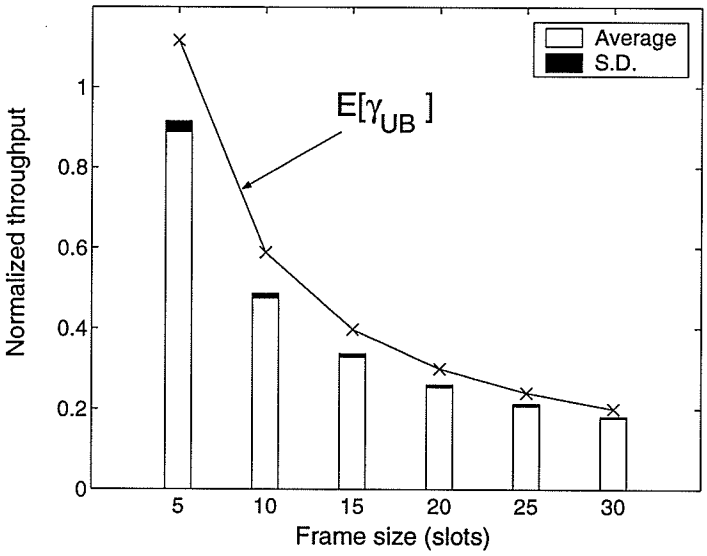


Figure 5.8. *Throughput vs. frame size.*

In ORCA-MRT, both number of mobiles and the corresponding weights do not have direct impact on system performance. Their increasing values only expand the scheduling frame size which in turn affect the performance.

From OBSERVATION 5.6, $\max_{\forall k}\{\mathscr{D}_{acc_i}(k)\} = 2\sum_{j \neq i} w_j + 1$. The delay increases as frame size becomes larger. Again, the $E[\mathscr{D}_{acc}]$ and $\sigma(\mathscr{D}_{acc})$ are not affected by frame size since they are normalized by w_i and T_F. Normalized throughput, on the other hand, decreases as frame size becomes larger because it is normalized by the frame size. We can observe in Figure 5.8 that the upper-bound of normalized throughput ($E[\gamma_{ub}]$) calculated from (5.31) decreases as frame size increases. The intuitive notion

behind this is that as the number of mobiles increases, each mobile has to wait longer for another transmission opportunity.

As frame size becomes larger, the scheduler has more choices to allocate higher rate to each mobile while maintaining fairness. The performance of ORCA-MRT with respective to the upper-bound becomes better for longer frame size.

5.5 Chapter Summary

This chapter extends ORCA-SRT to support multi-rate transmission environment. Based on the Hungarian method to solve the assignment problem, the ORCA-MRT scheduler utilizes optimization-based intra-frame rate allocation along with fairness compensation using lag counter. Also, to facilitate optimal bandwidth allocation we have proposed two methods for channel prediction. The proposed channel prediction methods perform almost as good as when the channel states are known ahead of time.

Simulation results have revealed that ORCA-MRT outperforms MFS in terms of throughput fairness and temporal fairness. Average throughput of ORCA-MRT is upper-bounded and lower-bounded by those obtained from the channel quality-based opportunistic and round-robin scheduling algorithms, respectively. ORCA-MRT ensures temporal fairness and bounds inter-access delay under a certain threshold. This feature would be useful to prevent end-to-end throughput (e.g., TCP throughput) degradation in wide-area wireless networks and/or to control QoS in real-time applications.

Chapter 6

Impact of Error Control on a Multi-hop Wireless Network: Part I – Single Packet Transmission

Multi-hop wireless networks are characterized by the lack of a direct communication link between source and destination nodes. End-to-end data transmission between a pair of nodes in this type of network requires intermediate nodes to forward data packets. For example, in a multi-hop cellular wireless network [60], the communications between a mobile and the base station can be carried out via relay nodes. The use of relay nodes helps increase service area and prevent network partitioning. In addition, short-range transmission improves spectral efficiency, increases spatial reuse, and leads to higher energy efficiency. A wireless backhaul network is another type of multi-hop network which consists of a collection of *Transit Access Points (TAPs)* [61]. These wireless TAPs forward traffic from mobiles to the internet gateway in a multi-hop manner. For successful end-to-end transmission of a packet, the packet needs to be successfully transmitted across all the links. Therefore, if the transmission fails (due to collision and/or channel fading) in one of the nodes *en route* the source and the destination nodes, retransmissions based on an ARQ (Automatic Repeat reQuest) policy will be necessary.

The end-to-end performance in a wireless multi-hop network depends strongly on the hop-level protocols and parameters. The study in [62] showed that the energy efficiency and end-to-end throughput (e.g., TCP throughput) depend greatly on

hop-level error probability, transmission range of each node, and maximum number of retransmissions at each node. However, only average values of the performance metrics such as energy efficiency or throughput were obtained.

Performance of end-to-end congestion and flow control mechanism in TCP over IEEE 802.11 *Distributed Coordination Function (DCF)* was investigated in [63] through simulations. In [64], the impact of TCP on end-to-end throughput performance in a multi-hop wireless network was analyzed and the optimal transmission window size was determined to be $H/4$ for a single TCP flow over a linear chain path with H hops. However, hop-level error control policies were not considered.

An analytical model for computing average steady state TCP throughput for a two-hop chain topology was presented in [65] assuming a collision-free and error-free wireless channel. Given the node density and the expected path length in a uniformly distributed ad hoc network, [66] computed optimal transmission distance which gives maximum network throughput and minimum energy consumption per data message. After all, a generalized analysis of the impacts of different hop-level error control policies on the end-to-end performance in a multi-hop network with an *arbitrary number of hops* has not been reported in the literature.

This chapter presents an analytical methodology to study the impact of radio link error and different hop-level ARQ policies on the end-to-end performance in a multi-hop wireless path. Specifically, by using *Phase-Type* (PH) distribution, we derive the *probability mass function (pmf)* of total required number of hop-level transmissions for successful end-to-end delivery of a packet in an H-hop chain topology. The usefulness of our analysis comes from the following facts. First, a general end-to-end transmission cost function can be defined in terms of number of hop-level transmissions. Based on this cost, the optimal routing paths (e.g., minimum energy paths [67]) can be determined for reliable communication in a multi-hop wireless network. Secondly, *Cumulative Distribution Function (cdf)* of the required number of transmissions can be utilized to quantify the reliability-energy tradeoff, since it is the probability to deliver a packet to the destination within a limited number of transmissions (and hence limited amount of transmission energy). Thirdly, statistics for end-to-end latency can be estimated if the information about queuing and channel access delay is available. This statistics can be used to set the transport control protocol timeout value at

the source node with a view to improve the end-to-end performance. Finally, since the proposed model is based on a generic link error process, the impact of different MAC and physical-layer parameters on the end-to-end performance under different hop-level error control strategies can be analyzed, and hence cross-layer design and engineering can be performed.

The rest of the chapter is organized as follows. Section 6.1 outlines the system model and assumptions. The Markov-based analytical model is presented in Section 6.2. The numerical and the simulation results as well as their useful implications are presented in Section 6.3. Finally, chapter summary are stated in Section 6.4.

6.1 System Model and Assumptions

6.1.1 Multi-Hop Network Model

We consider a data packet transmission scenario from a source node (A) to a destination node (D) over a multi-hop wireless path (Figure 6.1). We use a *chain topology* to represent the flow under consideration and regard all other active flows as background traffic[1]. We derive the statistics for total number of hop-level transmissions required for successful end-to-end delivery of a particular packet.

After transmitting a packet, each node can determine whether the transmission has been successful or not. The acknowledgement is transmitted over a different channel and is assumed to be error-free. If the transmission has failed, the node will invoke a retransmission procedure based on the ARQ policy at that node. If the packet is dropped (e.g., in case of *zero-retransmission policy*), the source node will retransmit the dropped packet.

6.1.2 Packet Error Model

Transmission of a packet in a link may fail due to data collision (when a distributed MAC protocol is used) and/or channel fading (independent or time-correlated). Data collision depends primarily on the underlying MAC layer. For example, the steady-state collision probability (p_{err_c}) for an IEEE 802.11 DCF-based MAC was analyzed

[1]Similar topology is used in a wireless backhaul network [61].

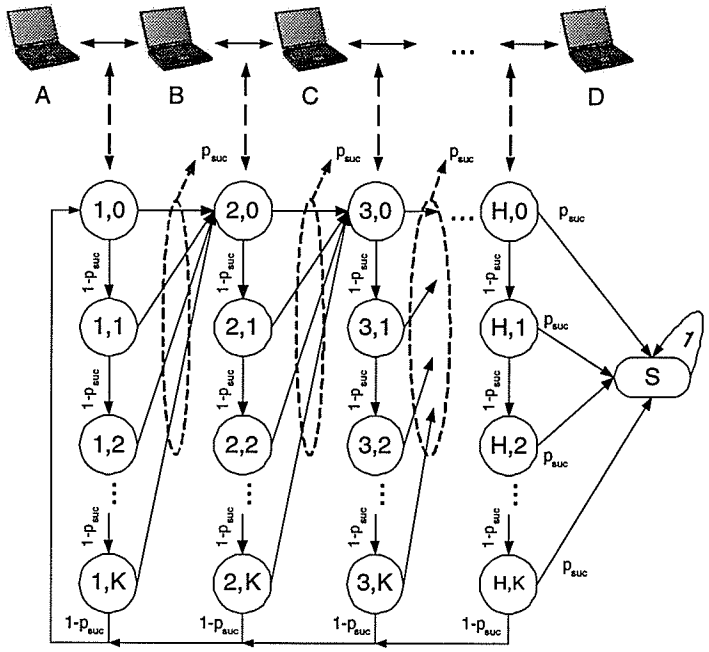


Figure 6.1. *An example of a chain topology and the corresponding Markov process for ARQ^F.*

in [68]. In a channel with independent channel fading (e.g., an RSC model in Section 2.3.1), packet transmission is assumed to be in error with probability p_{err_f}. Correlated channel fading can be modeled by the GEC model defined in Section 2.3.2.

In general, collision and fading are independent to each other. A methodology to calculate successful transmission probability (p_{suc}) for an IEEE 802.11 DCF-based MAC under Rayleigh fading channel was given in [69]. For a general MAC under independent fading channel, p_{suc} can be calculated from $(1 - p_{err_f}) \cdot (1 - p_{err_c})$. On the other hand, the successful transmission probability in a *good* and *bad* state of a correlated fading channel can be calculated as $p_{suc}^{(g)} = (1 - p_{err_f}^{(g)}) \cdot (1 - p_{err_c})$ and $p_{suc}^{(b)} = (1 - p_{err_f}^{(b)}) \cdot (1 - p_{err_c})$, respectively. By replacing $p_{err_f}^{(g)}$ and $p_{err_f}^{(b)}$ (packet error probability due to fading when the channel state is *good* and *bad*, respectively) in (2.24) with $1-p_{suc}^{(g)}$ and $1-p_{suc}^{(b)}$, respectively, the steady state packet error probability for a correlated fading channel can be determined.

6.1.3 Hop-Level Automatic Repeat ReQuest (ARQ) Model

If a node fails to deliver a packet to the next node, it will retransmit the packet according to one of the following hop-level ARQ policies:

- **Zero retransmission** (ARQ^0): If a transmission fails, the packet will be dropped immediately.
- **Infinite retransmission** (ARQ^∞): The packet is retransmitted repeatedly until the transmission is successful.
- **Finite retransmission** (ARQ^F): The maximum number of retransmissions allowed is K, and the packet will be dropped if the K^{th} retransmission fails.
- **Probabilistic retransmission with infinite persistence** (ARQ^P): If a transmission fails, the packet will be dropped with probability ξ and retransmitted with probability $(1-\xi)$.
- **Probabilistic retransmission with finite persistence** (ARQ^{FP}): This is similar to ARQ^P with the following constraints: $\xi < 1$ for first $K-1$ unsuccessful retransmissions and $\xi = 1$ for the K^{th} unsuccessful retransmission.

Definition 6.1 LINK[2] RELIABILITY *(r_{link}^{ARQ}) is the unconditional probability that a packet is successfully transmitted (before being dropped).*

Theorem 6.1 *For different ARQ policies, r_{link}^{ARQ} can be calculated as follows:*

$$r_{link}^{ARQ^0} = p_{suc}, \tag{6.1}$$

$$r_{link}^{ARQ^\infty} = 1, \tag{6.2}$$

$$r_{link}^{ARQ^F} = 1-(1-p_{suc})^{K+1}, \tag{6.3}$$

$$r_{link}^{ARQ^P} = \frac{p_{suc}}{p_{suc}+\xi-\xi p_{suc}}, \tag{6.4}$$

$$r_{link}^{ARQ^{FP}} = \frac{p_{suc}(1-(1-p_{suc})^{K+1}(1-\xi)^{K+1})}{p_{suc}+\xi-\xi p_{suc}}, \tag{6.5}$$

where p_{suc} is the probability of successful packet transmission over a link, K is the maximum number of allowable retransmissions in case of ARQ^F, and ξ is the packet dropping probability for ARQ^P. □

[2]We use the terms *link* and *hop* interchangeably in this dissertation.

PROOF:

1. *ARQ^0: Since the packet is dropped after one transmission,* $r_{link}^{ARQ^0} = p_{suc}$.
2. *ARQ^∞: Since the packet will never be dropped,* $r_{link}^{ARQ^0} = 1$.
3. *ARQ^F:* $r_{link}^{ARQ^F} = \sum_{i=1}^{K+1} p_{suc}(1-p_{suc})^{i-1} = 1-(1-p_{suc})^{K+1}$.
4. *ARQ^P:* $r_{link}^{ARQ^P} = \sum_{i=1}^{\infty} p_{suc}\left((1-p_{suc})(1-\xi)\right)^{i-1} = \frac{p_{suc}}{p_{suc}+\xi-p_{suc}\xi}$.
5. *ARQ^{FP}:* $r_{link}^{ARQ^{FP}} = \sum_{i=1}^{K+1} p_{suc}\left((1-p_{suc})(1-\xi)\right)^{i-1} = p_{suc}\frac{1-(1-p_{suc})^{K+1}(1-\xi)^{K+1}}{p_{suc}+\xi-p_{suc}\xi}$. ■

We observe that ARQ^0 and ARQ^∞ provide lower-bound and upper-bound for link reliability with $r_{link}^{ARQ^0} = p_{suc}$ and $r_{link}^{ARQ^\infty} = 1$, respectively. To achieve a target link reliability r_{link}^*, the minimum number of retransmissions K^* (in case of ARQ^F) or the maximum dropping probabilities ξ^* (in case of ARQ^P) can be obtained as follows:

$$K^* = \left\lceil \frac{\log(1-r_{link}^*)}{\log(1-p)} \right\rceil - 1, \quad \xi^* = \frac{p_{suc}(1-r_{link}^*)}{r_{link}^*(1-p_{suc})}. \tag{6.6}$$

In Section 6.3.7, we will show that THEOREM 6.1 would also be useful in estimating the end-to-end latency for a packet in a multi-hop route.

6.2 Markov Model and Analysis

6.2.1 Markov Modeling Under Independent Packet Error Process

We model a multi-hop wireless path by an absorbing DTMC (Figure 6.1). Again, we need to formulate TPM ,

$$\mathbf{P} = \left(\begin{array}{c|c} \mathbf{I} & \mathbf{0} \\ \hline \mathbf{t} & \mathbf{T} \end{array}\right), \tag{6.7}$$

as well as initial probabilities vectors $(\alpha_0, \boldsymbol{\alpha})$. We formulate these matrices only for ARQ^{FP} only, since it is the most general ARQ. Let $X^{(t)} \in \{sc, (h^{(t)}, k^{(t)})\}$ be the state of a tagged packet at service opportunity t. At opportunity t, the packet either reaches the destination ($X^{(t)} = sc$) or is being transmitted/waiting to be transmitted in one of the nodes along the route ($X^{(t)} = (h^{(t)}, k^{(t)})$), where the hop number ($h^{(t)} \in$

Table 6.1. *Implications of the sub-matrices*

Matrix	Event at time t	$(h^{(t+1)}, k^{(t+1)})$	Implication
Unsuccessful ($\mathbf{U}$)	$k^{(t)} < K$ and transmission fails.	$(h^{(t)}, k^{(t)}+1)$	Transmission fails and the current node retransmits the packet.
Successful ($\mathbf{S}$)	Successful Transmission	$(h^{(t)}+1, 1)$	Successful transmission. Start transmission in the next hop. Reset k to 1.
Restart ($\mathbf{R}$)	$k^{(t)} = K$ and transmission fails.	(1,1)	Packet is dropped. Retransmit from the hop $h = 1$.
Zero ($\mathbf{0}$)	Not possible	Not available	Not possible

$\{1, 2, \cdots, H\}$) corresponds to the link where the packet is being transmitted and $k^{(t)} \in \{0, 1, \cdots, K\}$ is the number of transmissions in the corresponding link.

The Markov process always starts when the packet is fed to the first node and finishes when the packet traverses the last hop. Therefore, we set $X^{(0)} = (1, 0)$ to be the initial state and $X^{(t)} = sc$ to be the absorbing state. Correspondingly, the initial probability (α_0) and initial probability vector ($\boldsymbol{\alpha}$) whose size is $1 \times H(K+1)$ are 0 and $(1, \mathbf{0})$, respectively.

We now construct transient TPM $\mathbf{T}$ and absorbing TPM $\mathbf{t}$ for the above DTMC in (6.8). Both of the matrices consist of blocks of sub-matrices. A transition among these blocks is equivalent to a change in the hop number (h), while a transition within a particular block represents the change in number of unsuccessful transmissions (k) in the same hop. The implications of all the sub-matrices are explained in Table 6.1. The elements in row i and column j of the sub-matrices $\mathbf{U}_h$, $\mathbf{S}_h$, $\mathbf{R}_h$, and $\mathbf{s}'$, whose sizes are $(K+1)^2$, $(K+1)^2$, $(K+1)^2$, and $(K+1) \times 1$, (i.e., $u_h(i,j), s_h(i,j), r_h(i,j), s'(i,1)$, respectively) can be calculated from (6.9)-(6.12) below

$$(\mathbf{t}|\mathbf{T}) = \left(\begin{array}{c|ccccc} \mathbf{0} & \mathbf{U}_1 + \mathbf{R}_1 & \mathbf{S}_1 & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{R}_2 & \mathbf{U}_2 & \mathbf{S}_2 & \mathbf{0} & \cdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \\ \mathbf{0} & \mathbf{R}_{H-1} & \mathbf{0} & \cdots & \mathbf{U}_{H-1} & \mathbf{S}_{H-1} \\ \mathbf{s}' & \mathbf{R}_H & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{U}_H \end{array}\right), \tag{6.8}$$

$$u_h(i,j) = \begin{cases} (1-p_{suc}(i,h))\cdot(1-\xi(i,h)), & i=\{1,\cdots,K\}, j=i+1, \\ 0, & otherwise, \end{cases} \quad (6.9)$$

$$s_h(i,j) = \begin{cases} p_{suc}(i,h), & i=\{1,\cdots,K+1\}, j=1, \\ 0, & otherwise, \end{cases} \quad (6.10)$$

$$r_h(i,j) = \begin{cases} (1-p_{suc}(i,h))\cdot\xi(i,h), & i=\{1,\cdots,K+1\}, j=1, \\ 0, & otherwise, \end{cases} \quad (6.11)$$

$$s'(i,1) = p_{suc}(i,H), i=\{1,\cdots,K+1\}, \quad (6.12)$$

where $p_{suc}(k,h)$ denotes probability of successful transmission corresponding to the k^{th} transmission in hop h, and $\xi(k,h)$ denotes dropping probability when the k^{th} transmission in hop h fails. By applying the formulated matrices (i.e, $\alpha_0 = 0, \boldsymbol{\alpha}, \mathbf{T}$, and $\mathbf{t}$) to (6.13)-(6.15), the *pmf* ($f_{\mathscr{T}}(t)$), *cdf* ($F_{\mathscr{T}}(t)$), and the expected value of the number of transmissions ($E[\mathscr{T}]$) required for successful end-to-end delivery can be calculated. In other words,

$$f_{\mathscr{T}}(t) = \boldsymbol{\alpha}\mathbf{T}^{t-1}\mathbf{t}, \quad (6.13)$$

$$F_{\mathscr{T}}(t) = 1-\boldsymbol{\alpha}\mathbf{T}^{t}\boldsymbol{e}, \quad (6.14)$$

$$E[\mathscr{T}] = \boldsymbol{\alpha}(\mathbf{I}-\mathbf{T})^{-1}\boldsymbol{e}. \quad (6.15)$$

6.2.2 Markov Modeling Under Correlated Packet Error Process

The analyzes above can be extended for the packet error model under correlated fading. At service opportunity t, a channel with correlated error (due to fading) can be in either *good* or *bad* state and the corresponding successful transmission probability is $p_{suc}^{(g)}$ and $p_{suc}^{(b)}$, respectively. For simplicity, we assume a homogeneous link condition across all the links. The analyzes for heterogeneous link conditions can be performed in a similar manner. Hereafter, we denote parameters for a correlated-error channel by superscript *corr*.

Assuming $p_{suc}(k,h) \in \{p_{suc}^{(g)}, p_{suc}^{(b)}\}, \forall k,h$, we modify the matrices $\boldsymbol{\alpha}$, $\mathbf{T}$, and $\mathbf{t}$ to support correlated error. We embed one more variable into the Markov process to keep track of the channel state. Therefore, the TPM becomes

$$(\mathbf{t}^{corr}|\mathbf{T}^{corr}) = \left(\begin{array}{c|cccc} \mathbf{0} & \mathbf{U}_1^{corr}+\mathbf{R}_1^{corr} & \mathbf{S}_1^{corr} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{R}_2^{corr} & \mathbf{U}_2^{corr} & \mathbf{S}_2^{corr} & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{s}^{corr'} & \mathbf{R}_H^{corr} & \mathbf{0} & \mathbf{0} & \mathbf{U}_H^{corr} \end{array}\right), \tag{6.16}$$

$$\mathbf{U}_i^{corr} = \frac{\mathbf{U}_i \otimes ((\mathbf{I}-\mathbf{G})\cdot\mathbf{P})}{1-p_{suc}}, \quad \mathbf{R}_i^{corr} = \frac{\mathbf{R}_i \otimes ((\mathbf{I}-\mathbf{G})\cdot\mathbf{P})}{1-p_{suc}}, \tag{6.17}$$

$$\mathbf{S}_i^{corr} = \frac{\mathbf{S}_i \otimes (\mathbf{G}\cdot\mathbf{P})}{p_{suc}}, \quad \mathbf{s}_i^{corr'} = \frac{\mathbf{s}_i' \otimes (\mathbf{G}\cdot\mathbf{1})}{p_{suc}}, \tag{6.18}$$

where $\otimes$ denotes Kronecker product, $\mathbf{G} = \begin{pmatrix} p_{suc}^{(g)} & 0 \\ 0 & p_{suc}^{(b)} \end{pmatrix}$, and $\mathbf{P}$ is the TPM for a GEC defined in (2.23). After constructing the matrices, the statistics for correlated-error channel can be obtained by using $\boldsymbol{\alpha}^{corr}$, $\mathbf{T}^{corr}$, and $\mathbf{t}^{corr}$ in (6.13)-(6.15).

6.2.3 Phase Type Representations for the Different ARQ Models

Assuming independent and homogeneous packet error process, we reduce entries of $\mathbf{T}$ in (6.8) for the following special cases with $K = 1$ to scalar values.

6.2.3.1 Zero Retransmission ARQ (ARQ^0)

In this case, $\xi = 1$ and $p_{suc}(k,h) = p_{suc}, \forall k,h$. Therefore,

$$(\mathbf{U}_h, \mathbf{S}_h, \mathbf{R}_h) = (0, p_{suc}, 1-p_{suc}), \forall h. \tag{6.19}$$

6.2.3.2 Infinite Retransmission ARQ(ARQ$^\infty$)

For ARQ$^\infty$, $\xi = 0$ and $p_{suc}(k,h) = p_{suc}, \forall k,h$. Therefore,

$$(\mathbf{U}_h, \mathbf{S}_h, \mathbf{R}_h) = (1-p, p, 0), \forall h. \tag{6.20}$$

Since $\mathbf{T}$ is diagonal dominant, a closed-form expression for the *pmf* ($f_{\mathscr{T}}(t)$) can be obtained as follows. From (6.13),

$$\begin{aligned} f_{\mathscr{T}}(t) &= \boldsymbol{\alpha}\mathbf{T}^{t-1}\mathbf{t} \\ &= (1,0,0,\cdots)(\mathbf{T}^{t-1}) \begin{pmatrix} 0 \\ 0 \\ \vdots \\ p_{suc} \end{pmatrix} = p_{suc} \cdot [\mathbf{T}^{t-1}]_{1,H}. \end{aligned} \tag{6.21}$$

Since $\mathbf{T}$ in ARQ$^{\infty}$ is a *bi-diagonal* matrix, a closed-form for $[\mathbf{T}^{t-1}]_{1,H}$ can be calculated from LEMMA E.1, and

$$f_{\mathscr{T}}(t) = \begin{cases} \begin{pmatrix} t-1 \\ H-1 \end{pmatrix} p_{suc}^{H}(1-p_{suc})^{t-H}, & t \geqslant H, \\ 0, & otherwise. \end{cases} \tag{6.22}$$

In (6.22), $f_{\mathscr{T}}(t)$ can be also regarded as a *negative binomial* or *Pascal* distribution which corresponds to the probability that there are $H-1$ successes among $t-1$ trials and another success at the t^{th} trial [19].

6.2.3.3 Probabilistic ARQ (ARQ^{P})

In this case, $p_{suc}(k,h) = p_{suc}, \forall k,h$, and

$$(\mathbf{U}_h, \mathbf{S}_h, \mathbf{R}_h) = ((1-\xi)(1-p_{suc}), p_{suc}, \xi(1-p_{suc})), \forall h. \tag{6.23}$$

6.3 Numerical and Simulation Results: Model Validation and Useful Implications

For brevity, we present results only for the independent packet error process, where packet error probability is fixed to p_{err}. The results for a correlated packet error process can also be generated by using the methodology provided in Section 6.2.2. Unless otherwise specified, we assume that all nodes implement the same type of ARQ and the probability of successful packet transmission is the same for all the links.

6.3.1 Model Validation

We verify the accuracy of our model by simulations using *ns-2* [70]. We establish a 10-hop chain topology and insert a link-loss module as well as ARQ models between each node and the connecting link. We run *File Transfer Protocol (FTP)* over TCP and plot the expected number of transmissions $E[\mathscr{T}]$ as a function of packet error probability in each link (Figure 6.2). Throughout this chapter, we set the maximum window size of TCP to 1. The payload of TCP is 500 bytes. The size of payload at the link layer is set to be the same as a TCP segment. During simulation, we measure the error probability in each link. The simulation terminates when the difference between input and measured link error probability is less than 10^{-6}. From the simulation, we calculate the expectation of measured hop-level transmissions associated with each TCP packet, and compare it with that obtained from the proposed framework.

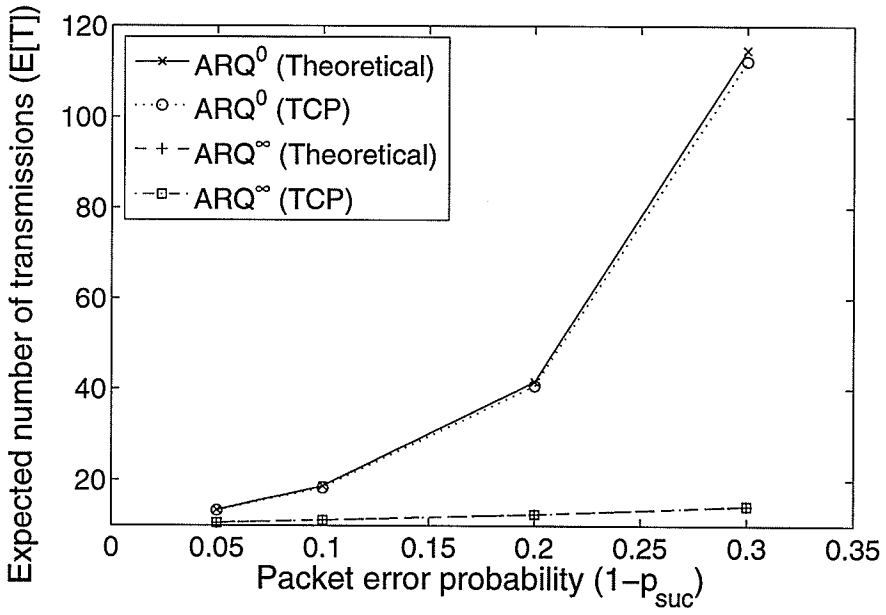


Figure 6.2. *Comparison between analytical and simulation results.*

As expected, $E[\mathscr{T}]$ increases as the link error probability increases and the results from the simulations are fairly close to those obtained from the analytical model (Figure 6.2). When the packet error probability is high, for ARQ^0, the packet rarely reaches the nodes closer to the destination node. In such a case, during simulation, the

link-loss module for the corresponding hops is rarely invoked. For this reason, at high link error rate, the analytical results on $E[\mathscr{T}]$ deviate slightly from the simulation results.

6.3.2 Impact of Hop-Level ARQ Policies on Expected Number of Transmissions

Figure 6.3 plots $E[\mathscr{T}]$ as a function of the number of hops (H) under different ARQ policies when the packet error probability in each link is 0.3. As expected, $E[\mathscr{T}]$ increases as the number of hops in the route increases. Since with ARQ^0 each unsuccessful transmission at any intermediate node requires retransmission from the source node, the upper bound of $E[\mathscr{T}]$ is observed for this ARQ policy. On the other hand, the lower bound of $E[\mathscr{T}]$ is achieved with ARQ$^\infty$ because in this case each intermediate node retransmits the lost packet until the transmission is successful.

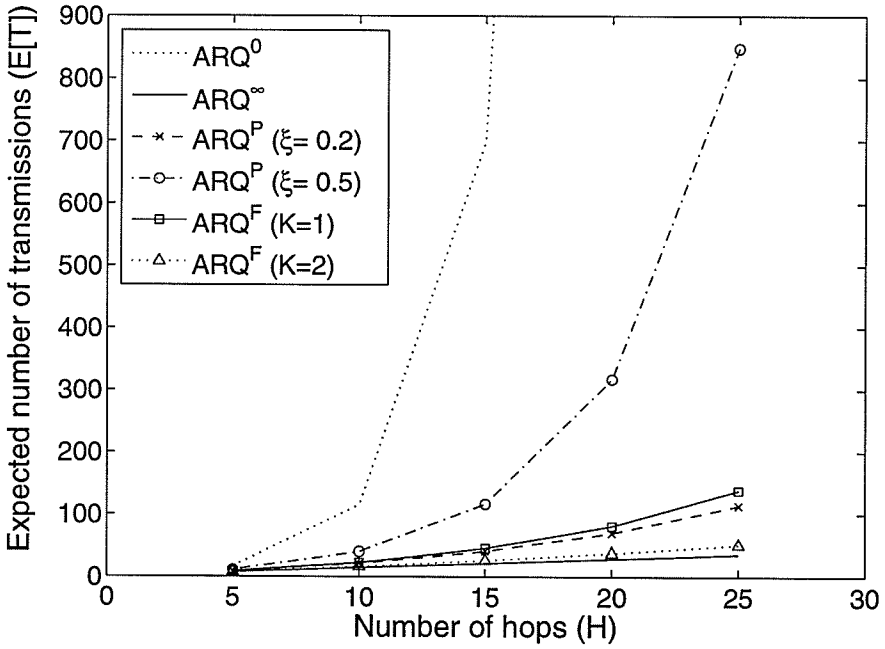


Figure 6.3. *Variations in the expected number of transmissions with number of hops in the route.*

The expected total number of transmissions increases as the maximum number of

allowable retransmissions in each hop (K) decreases (for ARQ^F) and/or the dropping probability (ξ) increases (for ARQ^P). In any case, the value of $E[\mathscr{T}]$ is bounded by those for ARQ^0 and ARQ$^\infty$. Both ARQ^F and ARQ^P policies are complementary to each other in that they yield the same $E[\mathscr{T}]$ when the parameters K and ξ are properly adjusted. Note that, $E[\mathscr{T}]$ for ARQ^F converges to the lower-bound fairly fast. From Figure 6.3, the required total number of transmissions becomes very close to the lower-bound when $K = 2$.

In general, the average amount of energy spent per hop-level transmission is a function of parameters such as transmission range, modulation techniques, and packet size. Given the average amount of energy consumption for a packet per hop-level transmission $\bar{e}$, the average energy spent for successful end-to-end delivery of a packet can be calculated as $E[\mathscr{T}] \times \bar{e}$. Since $E[\mathscr{T}]$ depends solely on the hop-level reliability and the number of hops, but neither on queuing delay nor on channel access delay, the expected number of transmissions required to deliver a window of N packets is $N \times E[\mathscr{T}]$. Also, the corresponding *pmf* for a window of N packets can be obtained by convoluting $f_{\mathscr{T}}(t)$ calculated from the above framework for N times.

6.3.3 Distribution of the Total Required Number of Transmissions

Figure 6.4 illustrates the *cdf* ($F_{\mathscr{T}}(t)$) of the required number of transmissions for a 10-hop connection when the packet error probability in each hop is 0.3. With the same argument, ARQ^0 and ARQ$^\infty$ converge to 1 at the lowest and highest rate. Again, the rate of convergence of ARQ^F and ARQ^P fall in between the above two.

We observe that the expected number of transmissions in ARQ^0 ensures less than 63.5% of packet delivery. Since $F_{\mathscr{T}}(t)$ represents the probability that a packet will be delivered to the destination within k transmissions, using $F_{\mathscr{T}}(t)$, the tradeoff between reliability and energy consumption can be analyzed.

6.3.4 Residual Improvement

Although ARQ$^\infty$ is the best ARQ policy in terms of $E[\mathscr{T}]$, it may cause head-of-line (HOL) blocking, and consequently, result in large hop-level delay and/or buffer over-

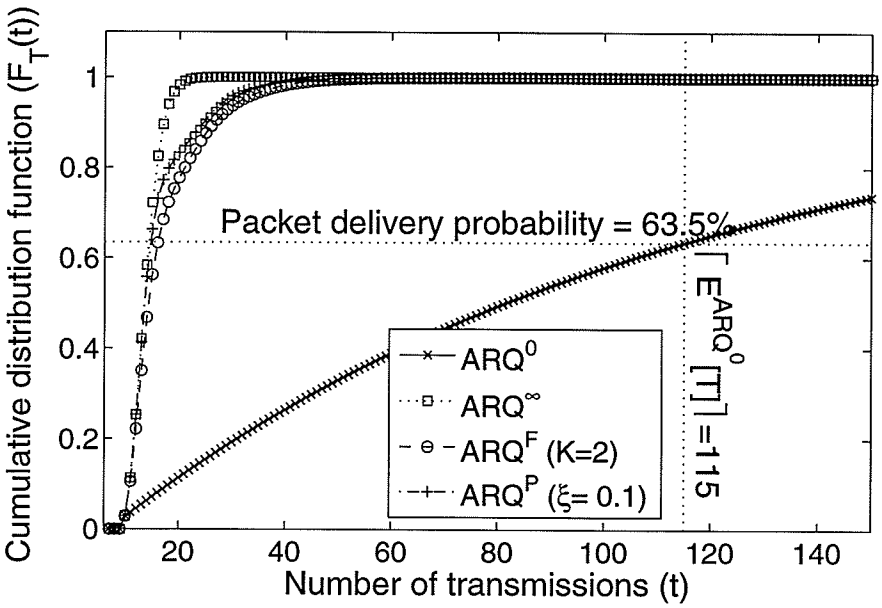


Figure 6.4. *Cumulative distribution function of the required number of transmissions ($F_{\mathcal{T}}(t)$) for different ARQ policies.*

flow. Therefore, ARQ^F and ARQ^P might be preferable to ARQ$^\infty$ in some scenarios. For these ARQ policies, there exist values of K or ξ further increase or decrease of which do not result in significant improvement in total required number of transmissions but would rather cause the HOL blocking problem.

To quantify the improvement for a particular ARQ policy (compared to the improvement from lower-bound to upper-bound corresponding to ARQ^{lb} and ARQ^{ub}, respectively), we define a parameter *Residual Improvement* (δ) as follows.

Definition 6.2 RESIDUAL IMPROVEMENT *($\delta^{ARQ}_{(lb,ub)}$) of an ARQ policy with respect to the improvement from upper-bound (ub) of the number of transmissions down to the corresponding lower-bound (lb) is defined in (6.24)*

$$\delta^{ARQ}_{(lb,ub)} \triangleq \frac{\sum_{t=0}^{\infty}\left(F_{\mathcal{T}}(t)^{lb} - F_{\mathcal{T}}(t)^{ARQ}\right)}{\sum_{t=0}^{\infty}\left(F_{\mathcal{T}}(t)^{lb} - F_{\mathcal{T}}(t)^{ub}\right)}. \tag{6.24}$$

□

The case with $\delta^{ARQ}_{(lb,ub)} = 1$ corresponds to an ideal ARQ, whose performance is the same as the lower bound of the number of transmissions. The worst case scenario, on the other hand, corresponds to an ARQ whose the number of transmissions is the

same as the upper-bound, where $\delta_{(lb,ub)}^{ARQ} = 0$. Therefore, small $\delta_{(lb,ub)}^{ARQ}$ implies good performance of an ARQ protocol. For two ARQ policies, namely, $ARQ1$ and $ARQ2$, it can be shown that (Appendix H)

$$\sum_{k=0}^{\infty} F_{\mathscr{T}}(t)^{ARQ1} - \sum_{k=0}^{\infty} F_{\mathscr{T}}(t)^{ARQ2} = E^{ARQ2}[\mathscr{T}] - E^{ARQ1}[\mathscr{T}], \tag{6.25}$$

where $F_{\mathscr{T}}(t)^{ARQ}$ and $E^{ARQ}[\mathscr{T}]$ refer to *cdf* and expectation of the number of transmissions for successful end-to-end delivery for a particular ARQ policy. Therefore,

$$\delta_{(lb,ub)}^{ARQ} = \frac{E^{ARQ}[\mathscr{T}] - E^{lb}[\mathscr{T}]}{E^{ub}[\mathscr{T}] - E^{lb}[\mathscr{T}]}. \tag{6.26}$$

In this section, we use ARQ^0 and ARQ^∞ as the upper-bound and the lower-bound, respectively. As can be observed from Figure 6.4, $\delta_{(lb,ub)}^{ARQ}$ is in fact the ratio of two areas–the area between the *cdf* for a certain ARQ and the *cdf* for ARQ^∞ and the area between the *cdf* for ARQ^0 and the *cdf* for ARQ^∞. Since $E[\mathscr{T}]^{ARQ^\infty} \leqslant E[\mathscr{T}]^{ARQ} \leqslant E[\mathscr{T}]^{ARQ^0}$, $\delta_{(\infty,0)}^{ARQ}$ lies between 0 and 1, where $\delta_{(\infty,0)}^{ARQ^0} = 1$, and $\delta_{(\infty,0)}^{ARQ^\infty} = 0$.

Figures 6.5-6.6 show typical variations in the residual improvement ($\delta_{\infty,0}^{ARQ^F}$ and $\delta_{\infty,0}^{ARQ^P}$) as a function of the maximum number of allowable retransmissions (K) and the dropping probability (ξ) in each hop, when the packet error probability in each hop is fixed to 0.3 and the number of hops varies from 5, 10, 15, to 20. As expected, for a particular H, $\delta_{\infty,0}^{ARQ^F}$ and $\delta_{\infty,0}^{ARQ^P}$ decrease with increasing K and decreasing ξ, implying that the corresponding *cdf* becomes closer to the *cdf* for ARQ^∞ (lower-bound). The performance of both ARQ^F and ARQ^P converge to the lower and the upper bounds when $K = \infty$ and 0 and $\xi = 0$ and 1, respectively.

We can achieve a certain level of residual improvement by adjusting K and ξ. For example, for a target value of $\delta_{\infty,0}^{ARQ^F} \leqslant 0.1$ and $\delta_{\infty,0}^{ARQ^P} \leqslant 0.1$, K must be chosen to be 2 for a 5-hop connection (Figure 6.5) and ξ must be set as the values located at the arrow ends in Figure 6.6. As H increases, both $E^{ub}[\mathscr{T}] - E^{lb}[\mathscr{T}]$ and $E^{ARQ}[\mathscr{T}] - E^{lb}[\mathscr{T}]$ increase. However, the rate of increase of $E^{ub}[\mathscr{T}] - E^{lb}[\mathscr{T}]$ (the denominator in (6.26)) is higher than that of $E^{ARQ}[\mathscr{T}] - E^{lb}[\mathscr{T}]$. Therefore, as H increases, the residual improvement relative to the $E^{ub}[\mathscr{T}] - E^{lb}[\mathscr{T}]$ becomes smaller. In effect, for larger H, smaller K and larger ξ can still satisfy the same constraint on residual improvement.

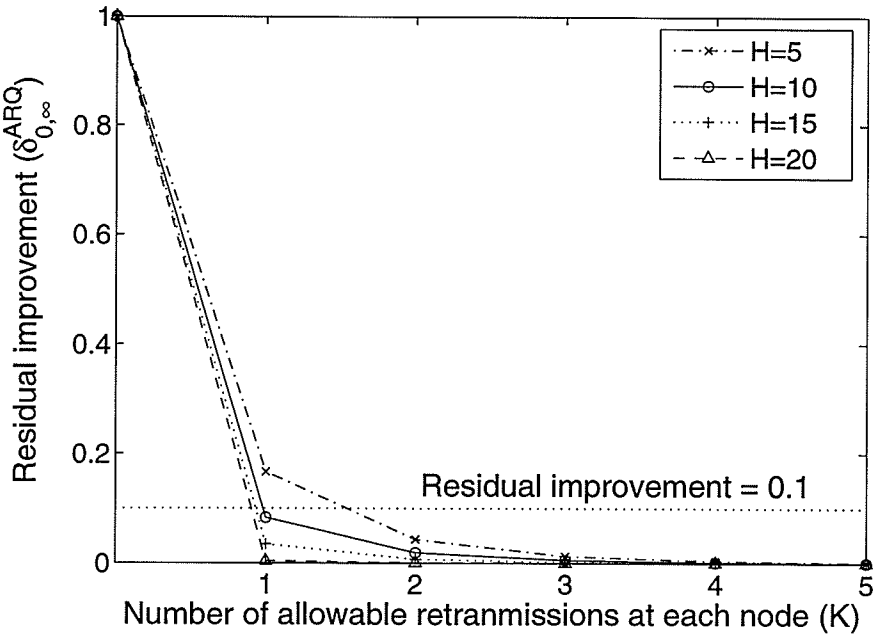


Figure 6.5. *Variations in residual improvement with maximum number of allowable retransmissions at each node.*

6.3.5 Heterogeneous Wireless Links

In general, packet error probability in each link in a multi-hop route can be different. This section presents typical numerical results under heterogeneous wireless links. We assume that each link can be either *good* with $p_{err} = 0.1$ or *bad* with $p_{err} = 0.3$, and show the *cdf* for the required number of transmissions for successful end-to-end delivery ($F_{\mathscr{T}}(t)$) under ARQ$^{\infty}$ in Figure 6.7. The legends in this figure correspond to cases where all links are *good*, all links are *bad*, *first bad* (where all except the first link is *good*), and *last bad* (where all except the last link is *good*).

With ARQ$^{\infty}$, since all the nodes persist on transmitting the packet until the transmission is successful, the *cdf* in this case depends only on the number of *bad* links but not on the location of the *bad* link(s). However, the location of a *bad* link does affect the performance of all other ARQ policies. Using the *all good* and the *all bad* cases for ARQ^0 as the lower-bound and the upper-bound, respectively, we denote the residual improvements in case of ARQ^0 for the *first bad* and the *last bad*

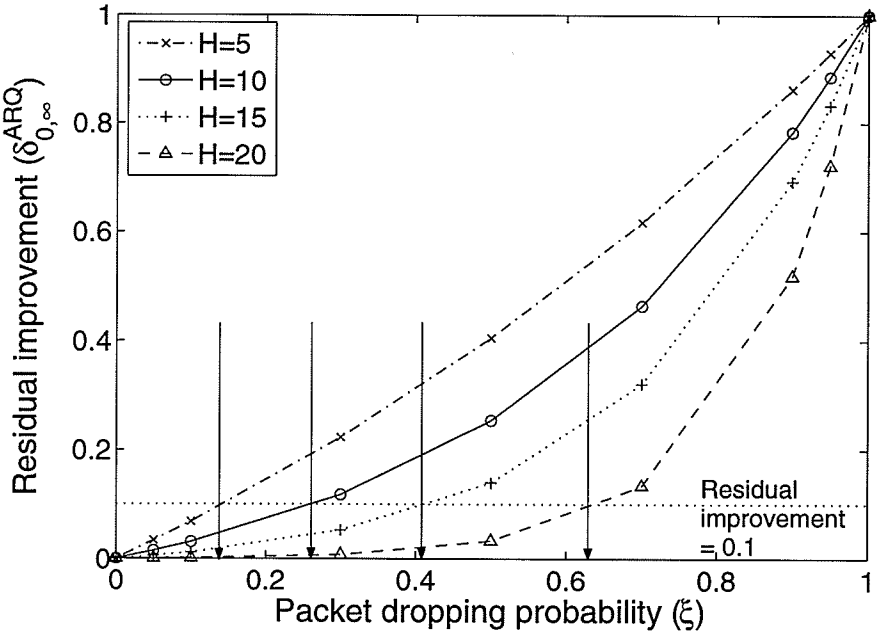


Figure 6.6. *Variations in residual improvement with packet dropping probability at each node.*

cases by $\delta_{(all\ good,all\ bad)}^{first\ bad}$ and $\delta_{(all\ good,all\ bad)}^{last\ bad}$, respectively. For the *last bad* case, it is more likely that the packet will be dropped at the last hop. If this happens, the transmissions which have already been succeeded earlier will be useless, since the source will have to retransmit the packet. We observe that $\delta_{(all\ good,all\ bad)}^{first\ bad}$ is always less than $\delta_{(all\ good,all\ bad)}^{last\ bad}$ (Figure 6.8). That is, the system performance drops when location of the *bad* link moves towards the destination.

As the number of hops increases, the required number of transmissions in all the cases increases. With respect to the *all good* case, the rate of relative increase in the *all bad* case is the highest because it has more number of *bad* links. In fact, this relative increase is the denominator in (6.26). Therefore, the residual improvement always decreases as H increases. Although $\delta_{(all\ good,all\ bad)}^{first\ good} \leqslant \delta_{(all\ good,all\ bad)}^{first\ bad}$, the difference becomes smaller as H increases because the denominator becomes more dominant.

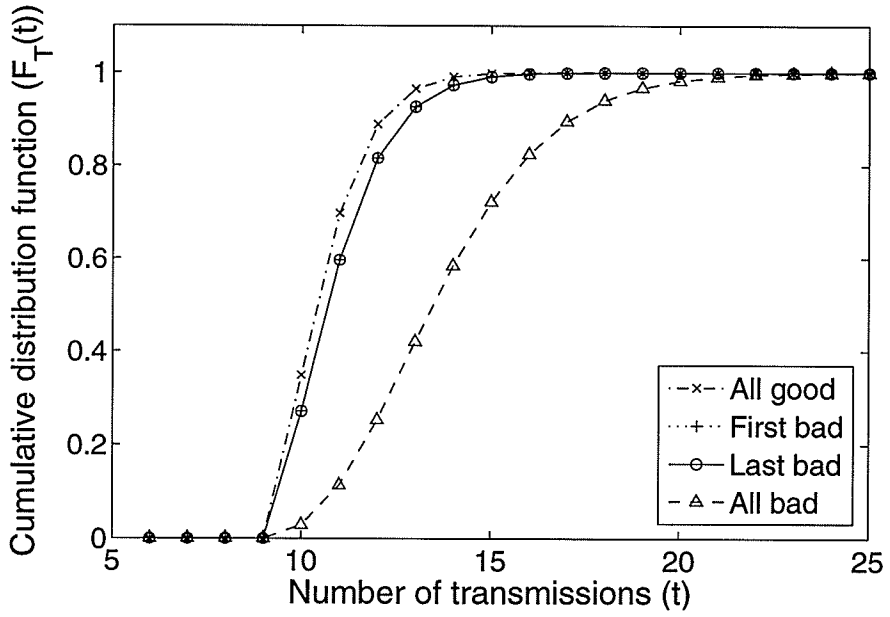


Figure 6.7. *Cumulative distribution function of the total required number of transmissions ($F_{\mathcal{T}}(t)$) for ARQ$^{\infty}$ under non-homogeneous wireless links.*

6.3.6 Impact of MAC Protocols: Typical Results for IEEE 802.11 DCF

Using *ns-2*, we run an FTP/TCP flow over a four hop linear chain topology. Each node implementing IEEE 802.11 MAC, and has a transmission range of 250 m. The distance between any two nodes is also 250 m (we will refer to these nodes as chain nodes, hereafter). We place background nodes (each with transmission range of 100 m) within a distance of r meters from each chain node, where r is randomly chosen between 0 to 100. We also set the retry limit in IEEE 802.11 DCF to ∞ and compare the results with ARQ$^{\infty}$. This setting is necessary to prevent route failure along the chain of nodes.

Under a two-ray ground reflection propagation model, every node generates CBR (Constant Bit Rate) traffic at the rate of 10, 20, 30, 40, and 50 *kbps*, destined to the closest chain node. Consisting of chain nodes as well as background nodes, this topology is very similar to that of wireless backhaul networks [61], where the chain nodes are analogous to the TAPs.

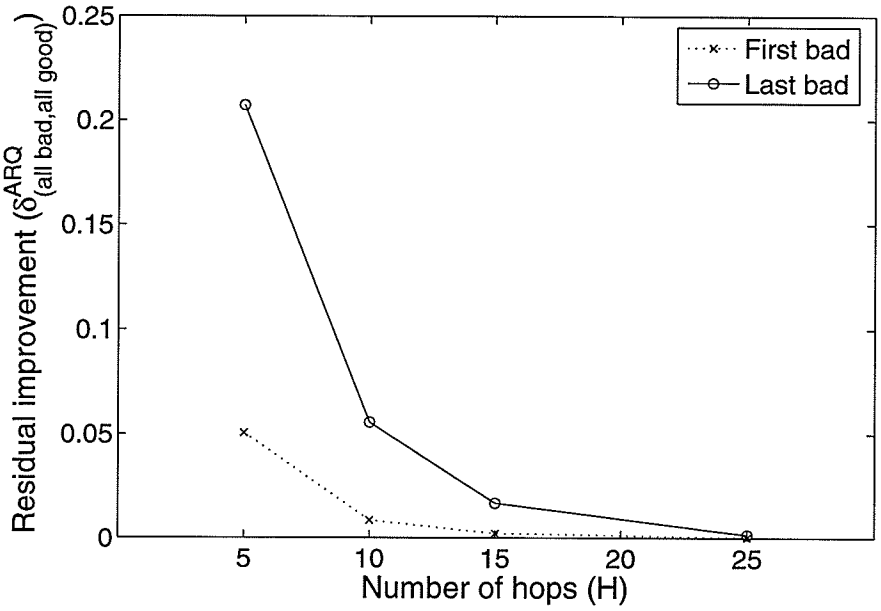


Figure 6.8. *Variations in residual improvement with number of hops under non-homogeneous wireless links.*

The simulation is run for 20 times. In each simulation, the first chain node sends out 1000 TCP segments destined to the last chain node. We measure the collision probability at each hop along the chain of nodes as well as the required number of hop-level transmissions for successful delivery of each TCP packet. Based on the measured collision probabilities, we calculate $E[\mathscr{T}]$ from the above framework and compare it with that obtained from simulation. Note that, this dissertation does not focus on modeling the collision probability p_{err_c}. Therefore, we use the measured value of p_{err_c} to calculate $E[\mathscr{T}]$. The value for p_{err_c} can be obtained by estimating the number of interfering mobiles following the approach in [71] and then using the method presented in [68].

In absence of channel fading, collision is the major cause of transmission failure. Figure 6.9 (a) shows typical variations in collision probability in the first hop of the chain route, where the number of interfering nodes refers to the number of mobiles which can cause collision at the chain node. Similar results have been observed in the other hops.

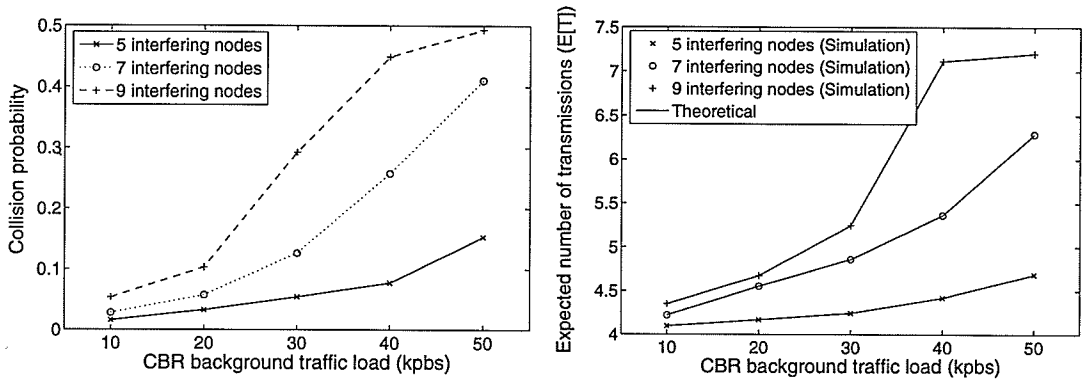


Figure 6.9. *Impact of 802.11 DCF MAC on* (a) *collision probability and on* (b) *expected number of transmissions.*

We observe that IEEE 802.11 suffers greatly from frequent data collisions due partly to the CBR background traffic, hence resulting in very low successful transmission probability. This result is in fact not surprising, since the collision probability for only 7 mobile nodes in the same neighborhood is expected to be greater than 20% [68]. In our simulations, this probability becomes even higher due to *hidden terminal jamming* problem [72].

Figure 6.9 (b) compares $E[\mathscr{T}]$ obtained from simulation (shown by symbols) with that obtained from the framework (shown by solid line). When collision probabilities are known, our model is extremely accurate in that all the symbols (simulation) fall very close to the line generated by the framework.

6.3.7 Estimation of End-to-End Transmission

The end-to-end latency ($\mathscr{L}$) for a packet depends primarily on queuing delay ($\mathscr{D}_q$), channel access delay ($\mathscr{D}_{acc}$) associated with each hop-level transmission, total number of hop-level transmissions ($\mathscr{T}$), and packet transmission time ($\mathscr{D}_{tx}$). In each hop, the queuing delay for a tagged packet is the time that the packet waits before it reaches the head of the queue. The channel access delay is the time required for a packet to be transmitted after it reaches the head of the queue. The queuing delay depends on the buffer management and scheduling policies, while the channel access delay

depends on the MAC scheme used by the nodes. The transmission time depends on the packet size and the link speed.

Assuming that the average queuing delay ($E[\mathscr{D}_q]$) and the average channel access delay ($E[\mathscr{D}_{acc}]$) are available, the expected end-to-end latency for a packet can be calculated from $E[\mathscr{L}] = E[\mathscr{N}_q] \cdot E[\mathscr{D}_q] + E[\mathscr{T}] \cdot (E[\mathscr{D}_{acc}] + \mathscr{D}_{tx})$, where $\mathscr{N}_q$ is the number of times the packet is enqueued in any of the queues in the multi-hop route before it reaches the destination node. Note that, the expected end-to-end throughput can be calculated as $1/E[\mathscr{L}]$.

The expected value of $\mathscr{N}_q$ ($E[\mathscr{N}_q]$) for the different hop-level ARQ policies can be calculated as follows. First, evaluate link reliability (r_{link}^{ARQ}) for the implemented ARQ (in each hop) using THEOREM 6.1. Secondly, formulate $\mathbf{U}_i$, $\mathbf{S}_i$, and $\mathbf{R}_i$ for ARQ^0 (6.19). Thirdly, replace p_{suc} in the formulated matrices with the calculated r_{link}^{ARQ}. This substitution is equivalent to the use of ARQ^0 at each hop along the route with the unconditional packet dropping probability at each node being equal to the one for the implemented ARQ. Finally, calculate $E[\mathscr{T}]$ by using the proposed framework, which in this case is equivalent to $E[\mathscr{N}_q]$.

6.4 Chapter Summary

We have presented a methodology for analyzing the impacts of hop-level ARQ policies on end-to-end performance statistics in a multi-hop wireless network. The number of hop-level transmissions is lower-bounded and upper-bounded by ARQ policies with infinite and zero retransmissions, respectively. Performances of all the ARQ policies, except that for ARQ with infinite retransmissions, degrade as the location of a weak wireless link moves towards the destination.

Simulation results have validated the analytical results. The proposed framework can be used to estimate the total amount of energy consumption and to analyze the tradeoff between reliability and energy for reliable end-to-end transmission. Also, it would be useful in estimating the end-to-end latency (hence throughput) and improving the end-to-end flow control mechanism. Consideration of wireless channel parameters and medium access control schemes would extend the use this model for analyzing cross-layer protocol performance. After all, the proposed model can be

used to effectively study the inter-relationship among link-level packet error probability, hop-level ARQ policy and parameters, and the end-to-end performance in a multi-hop wireless network.

Chapter 7

Impact of Error Control on a Multi-hop Wireless Network: Part II – Batch Transmission

In chapter 6, an analytical model for a multi-hop wireless network with an arbitrary number of hops was proposed. The main observation there was that the increasing number of hops significantly degrades system performances (e.g., Figure 6.3). Therefore, the number of hops in a path should be limited to a small number. For example, in a multi-hop cellular network ([16],[73]) or a *wireless mesh network* [14], short range communication leads to better link quality, higher transmission rate, less energy consumption, and improved load-balancing among base stations. Typically the network path in such a scenario consists of only few hops (e.g., two hops), and due to the presence of a central controller (e.g., a base station) the network is more controllable compared to a pure ad hoc network.

There is limited research regarding multi-hop cellular networks. An architecture, namely, the ad hoc global system for mobile communications (AGSM) was proposed in [74] which allows mobiles to switch to an ad hoc mode when the direct link to the base station is unavailable. Mobiles located far away from the base station could achieve higher data rate and better fairness with the aid of the relaying architecture proposed in [75]. Developed in [76], the multi-hop cellular network (MCN) architecture enabled short range communication and reduced transmission power for mobile nodes. An integrated cellular and ad hoc relay (iCAR) system to redirect traffic from a highly loaded base station to neighboring base stations was presented in [77]. For a cellular multi-hop network, [60] proposed two algorithms in which the base station participates

in packet relaying and each mobile adaptively switches its mode of operation (peer-to-peer or direct communication with the base station). Analytical models to evaluate network throughput under multi-hop relaying were developed in [78] and [79].

Again, the inefficiency of TCP in a multi-hop wireless network with IEEE 802.11-based MAC was revealed in [63] and [80]. With transmission window size of one, [81] investigated the effects of different hop-level ARQ protocols on end-to-end performance. The effects of number of hops and channel error on energy efficiency and throughput performance of TCP were studied in [62]. [65] presented a model for analyzing TCP performance over a two-hop chain topology under error-free wireless channels and a collision-free MAC protocol, assuming that only one mobile can transmit at any instant in time.

In the literature, all of the analytical works on multi-hop wireless transmission modeled only throughput performance and assumed a fixed transmission rate and/or only one certain type of ARQ. Since multi-rate transmission such as that achieved through AMC can significantly impact the transmission performance in a radio link, its impact on the end-to-end protocol performance need to be studied. To the best of our knowledge, a general analytical framework which captures the impact of multi-rate transmission, packet error probability, as well as ARQ policies (at each hop) on the end-to-end latency and reliability in a multi-hop wireless network was not reported in the literature.

In this chapter, we present an analytical model to evaluate end-to-end performance in a multi-hop and multi-rate wireless network. For a batch of packets, we determine the probability mass function (*pmf*) of the end-to-end latency (in terms of number of transmission intervals) and the number of packets successfully delivered to the destination under different AMC settings and hop-level ARQ policies. These statistics could be useful in many different ways.

By analyzing end-to-end latency and reliability, the hop-level ARQ parameters can be adjusted to achieve the optimal routing paths and/or desired latency-reliability tradeoff. Since an ARQ protocol increases link reliability at the expense of latency, different applications might choose different paths to fulfill their requirements. For example, delay sensitive application might opt for a low latency but less reliable path, while a more reliable path might be preferable for data traffic.

The statistics on end-to-end latency and reliability can be used to improve the performance of TCP flow-control mechanism or design even more efficient flow control protocols. Specifically, for an end-to-end flow and error control mechanism for a multi-hop wireless path, the results from our model would be useful to guide the source about when to transmit and when to withhold the transmission. In a window-based flow control protocol such as in TCP, the source node essentially transmits batches of packets, one after another. Therefore, data packets in the previous batches which have not yet reached the destination might affect the latency and the reliability performances for the current batch of transmission. Our model can capture this interaction, by analyzing the the end-to-end performance for a particular batch of packets and considering packets from the previous batch(es) which are still in the network path. For example, upon receiving an acknowledgement, the TCP sender prepares a new batch of packets for transmission. At this moment, the sender can estimate the number of packets in the network (e.g., by using the unacknowledged sequence numbers) based upon which it can estimate the throughput and delay statistics which could be utilized for adjustment of window size and/or the timeout value. Multi-hop batch transmissions also occur in sensor networks during network tasking [82].

The organization of this chapter is as follows. Section 7.1 outlines system model, assumptions, and methodology for the analysis. Section 7.2 presents the Markov-based model to evaluate end-to-end performance for a batch transmission across multi-hop wireless links. The numerical and simulation results are presented in Section 7.3. In Section 7.4, we demonstrate the usefulness of our model in predicting the performance of TCP in a multi-hop transmission scenario. Finally, the summary and the concluding remarks are given in Section 7.5.

7.1 System Model, Assumptions, and Methodology

7.1.1 System Description

Consider a static two-hop chain topology[1] where the transmission range of each node is just enough to reach its neighboring nodes (Figure 7.1). We assume that node 1 and node 3 are the source and the destination nodes, respectively. A batch transmission is initiated with N_1 and N_2 packets at node 1 and node 2, respectively. Due to limited transmission range, node 1 forwards these packets to node 2, which in turn forwards them to node 3. The process finishes when the buffers of node 1 and node 2 are empty (i.e., either delivered to the destination or dropped due to limited persistence of an ARQ protocol). For analytical tractability, we do not consider MAC-layer delay in this chapter. We also assume that communications in both the hops can occur at the same time[2].

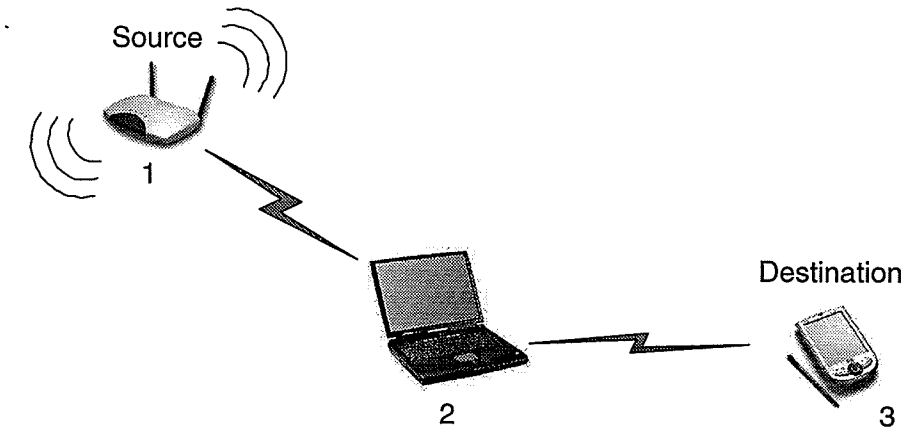


Figure 7.1. *A two-hop chain topology.*

[1]Although we consider a two-hop network here, the following analyzes can be performed for a general H-hop network in a similar way.

[2]Examples of MAC protocols which allow concurrent transmissions in both hops include the ODMA (opportunity driven multiple access) for UMTS networks [78], an interference-limited CDMA protocol [79], and a MAC protocol in a multi-radio multi-hop wireless mesh network where each mobile is equipped with multiple wireless interfaces [83].

7.1.2 Wireless Channel Model and Multi-Rate Transmission

The channel model in this chapter is assumed to follow an RSC model defined in section 2.3.1. During a transmission interval[3] where the channel state is m, a mobile can transmit at most $r(m)$ packets in order to satisfy the constraint on packet error probability (p_{err}). Each of $r(m)$ transmitted packets are assumed to be in error with probability p_{err}. The probability that exactly s packets are successfully transmitted by node n during a particular interval ($p_n(s_n)$) can be calculated from (7.1) below

$$p_n(s_n) = \sum_{\forall m: r(m) \leqslant x_n} p_n(s_n|r(m)) \cdot \pi_m + p_n(s_n|x_n) \cdot \left(\sum_{\forall m: r(m) > x_n} \pi_m \right), \quad (7.1)$$

$$p_n(s_n|r(m)) = \begin{cases} \binom{r(m)}{s_n} \cdot (1 - p_{err}(n))^{s_n} \cdot (p_{err}(n))^{r(m)-s_n}, & s_n \leqslant r(m), \\ 0, & s_n > r(m), \end{cases} \quad (7.2)$$

where x_n, $p_{err}(n)$, $r(m)$, and M are the number of packets in node n, the average packet error probability in the corresponding link, the transmission rate corresponding to channel state m, and the number of channel states, respectively. Although we assume $r(m) = m$ throughout this chapter, other channel-to-rate mapping functions (e.g., those in [23]) can be modeled in a similar manner.

7.1.3 Hop-Level Automatic Repeat Request (ARQ) Protocols

We define a *transmission failure* as an event where all of the transmitted packets during a transmission interval are lost. This may occur due to two main reasons: first, when all of the transmitted packets are corrupted due to channel fading, and secondly, when the intended receiver node is turned off or moves out of the transmission range of the transmitter node. In the latter case, the transmitter invokes a route discovery process, during which it does not receive any packet (corresponding to the tagged batch) from other nodes. To combat with transmission failure due to channel fading, each node implements a certain type of ARQ protocol, which retransmits only the lost

[3]A transmission interval is defined as the period for each node to transmit a burst of packets to its neighboring node and to determine which packets have been successfully transmitted.

Table 7.1. *Parameter adjustment of ARQ^{FP} to represent other ARQ protocols.*

	ARQ^0	ARQ^∞	ARQ^F	ARQ^P
$\xi_n(k), k < K_n$	1	0	0	ξ_n
$\xi_n(K_n)$	1	0	1	ξ_n
K_n	1	1	K_n	1

packets. We consider different ARQ protocols defined in section 6.1.3 which provide different levels of reliability-latency tradeoff.

Among those ARQ variants, ARQ^{FP} is the most general one. Table 7.1 shows different parameter settings for which ARQ^{FP} reduces to the other variants of ARQ. In this chapter, we develop the end-to-end model with ARQ^{FP} in each hop.

7.1.4 Methodology for Analysis

We derive complete statistics in terms of *pmf*, *cdf*, and expectation of the two following basic performance parameters: the *end-to-end latency* ($\mathscr{D}_{e2e}$) and the *number of packets successfully delivered to the destination* ($\mathscr{S}$). The end-to-end latency is defined as the number of *transmission intervals* required for all packets to reach the destination (when ARQ^∞ is used to provide reliable end to-end delivery) or to be transmitted through/dropped out of the network (when other types of ARQ is used). The end-to-end latency is the duration after which there is no packet (from the tagged batch) in the network. From these two basic parameters (i.e., $\mathscr{S}$ and $\mathscr{D}_{e2e}$), other performance metrics can also be calculated. For example, the *end-to-end throughput* (γ_{e2e}) and the *transmission reliability* can be calculated from $\mathscr{S}/\mathscr{D}_{e2e}$ and $\mathscr{S}/N$, respectively, where $N = N_1 + N_2$ is the total number of packets in the entire network path.

We model a batch transmission process by using an absorbing DTMC. This DTMC starts when N_1 packets are at the source node, while N_2 packets are in the intermediate node. The DTMC finishes when all $N = N_1 + N_2$ packets leave (either delivered or dropped) the network path, and the process is absorbed to a state in which $\mathscr{S} \in \{0, 1, \cdots, N\}$ packets are successfully delivered to the destination.

The *pmf* ($f_{\mathscr{D}_{e2e}}(d)$) and the expected value ($E[\mathscr{D}_{e2e}]$) of end-to-end latency, and

the $pmf\,(f_{\mathscr{S}}(s))$ and the expected value $(E[\mathscr{S}])$ of the number of packets successfully delivered to the destination can be evaluated as follows:

$$f_{\mathscr{D}_{e2e}}(d) = \boldsymbol{f}_{\mathscr{T}}(t) \cdot \boldsymbol{e}, \tag{7.3}$$

$$f_{\mathscr{S}}(s) = [\boldsymbol{f}]_{1,s}, \tag{7.4}$$

$$E[\mathscr{D}_{e2e}] = \boldsymbol{E}[\mathscr{T}] \cdot \boldsymbol{e}, \tag{7.5}$$

$$E[\mathscr{S}] = \sum_{s=0}^{N} s \cdot f_{\mathscr{S}}(s), \tag{7.6}$$

where $\boldsymbol{f}_{\mathscr{T}}(t)$, $\boldsymbol{f}$, and $\boldsymbol{E}[\mathscr{T}]$ can be calculated from (2.16)-(2.18). In the next section, we formulate the initial probability vector as well as the transient and absorbing TPMs for the above DTMC. All relevant results can then be calculated by using the formulated matrices in (7.3)-(7.6).

7.2 Analysis of Batch Transmission Under Different Hop-Level ARQ Policies

In this section, we first assume that the number of packets in the intermediate node N_2 is zero (i.e., $N = N_1$), and formulate the key matrices for ARQ$^{\infty}$. Then, we use the formulated matrices to obtain the matrices for ARQ^{FP}. Towards the end of this section, we also generalize the derived results for the case where $N_2 \neq 0$.

At each transmission interval t, we keep track of the number of packets (from the tagged batch) in the buffer of node n by using $X_n^{(t)}$ $(n \in \{1,2,3\})$[4]. We use an absorbing DTMC $\mathbf{X}^{(t)} = (X_1^{(t)}, X_2^{(t)}, X_3^{(t)})$ to model the above transmission process. The process starts with $\boldsymbol{X}^{(0)} = (N, 0, 0)$, and finishes at interval t where $X_1^{(t)} + X_2^{(t)} = 0$ (absorbing states).

7.2.1 Infinite Retransmission ARQ (ARQ$^{\infty}$)

Since with ARQ$^{\infty}$, each lost packet is retransmitted until it is successfully received by the next node in the path, the total number of packets in the network path is always

[4]In fact, $(X_1^{(t)}, X_2^{(t)})$ is sufficient to represent the process, but we also include $X_3^{(t)}$ into $\mathbf{X}^{(t)}$ to facilitate understanding of the process.

Table 7.2. *All possible transitions from* $\mathbf{X}^{(t)}$ *to* $\mathbf{X}^{(t+1)}$.

$(X_1^{(t+1)}, X_2^{(t+1)})$	$P(\mathbf{X}^{(t+1)}\|\mathbf{X}^{(t)})$
$((X_1^{(t)} - s_1), (X_2^{(t)} - s_2 + s_1)), s_1 < X_1^{(t)}, s_2 < X_2^{(t)}$	$p_1(s_1)p_2(s_2)$
$((X_1^{(t)} - s_1), s_1), s_1 < X_1^{(t)}, s_2 \geqslant X_2^{(t)}$	$p_1(s_1)p_2(X_2)$
$(0, (X_2^{(t)} - s_2 + X_1^{(t)})), s_1 \geqslant X_1^{(t)}, s_2 < X_2^{(t)}$	$p_1(X_1)p_2(s_2)$
$(0, X_1^{(t)}), s_1 \geqslant X_1^{(t)}, s_2 \geqslant X_2^{(t)}$	$p_1(X_1)p_2(X_2)$

N, or mathematically

$$\sum_{n=1}^{3} X_n^{(t)} = N, \quad \forall t. \tag{7.7}$$

The states which violate the constraint (7.7) are unreachable and can be removed from the DTMC. In other words, all reachable states of $\boldsymbol{X}^{(t)}$ can be obtained by enumerating all reachable buffer states of node 1 and node 2, i.e., $X_1^{(t)} = \{0, 1, \cdots, N\}$ and $X_2^{(t)} = \{0, 1, \cdots, N - X_1^{(t)}\}$. The number of reachable states in the DTMC is, therefore, $\sum_{x=0}^{N}(N + 1 - x) = (N + 1)(N + 2)/2$. Due to (7.7), all N packets will reach the destination and there exists only one possible absorbing state $(0, 0, N)$.

Let s_n be the number of packets successfully transmitted by node n with the corresponding probability $p_n(s_n)$ defined in (7.1). All the possible transitions from $\boldsymbol{X}^{(t)}$ to $\boldsymbol{X}^{(t+1)}$ and the corresponding conditional probabilities $(P(\boldsymbol{X}^{(t+1)}|\boldsymbol{X}^{(t)}))$ are given in Table 7.2. We arrange the states $(X_1^{(t)}, X_2^{(t)})$ in an ascending order. Therefore, the initial probability vector $(\boldsymbol{\alpha}_N)$, the transient TPM $(\mathbf{Q}_N)$, and the absorbing TPM $(\mathbf{R}_N)$ for ARQ$^\infty$ with batch size N can be formulated in (7.8)-(7.11).

$$\boldsymbol{\alpha}_N = (\mathbf{0}, 1). \tag{7.8}$$

$$\mathbf{P}_N = \left(\begin{array}{c|c} 1 & \mathbf{0} \\ \hline \mathbf{R}_N & \mathbf{Q}_N \end{array}\right), \quad \text{where} \tag{7.9}$$

$$[\mathbf{Q}_N]_{i,j} = \begin{cases} \mathbf{A}'_N, & i = j = 1, \\ (\mathbf{0}, p_1(i-j) \cdot \mathbf{A}_{N-i+2}), & i = \{2, \cdots, N+1\}, j = \{1, \cdots, i\}, \end{cases} \tag{7.10}$$

$$[\mathbf{R}_N]_{i,1} = \begin{cases} p_2(i), & i = \{1, \cdots, N\}, \\ 0, & \textit{otherwise}. \end{cases} \tag{7.11}$$

Here, $\mathbf{Q}_N$ and $\mathbf{R}_N$ are a square matrix and a column vector, respectively, with size

$N(N+3)/2$. Inherited in each row/column of $\mathbf{Q}_N$, $\mathbf{A}'_N$ and $\mathbf{A}_k$ are square matrices with size N and k, and their entries (i,j) are defined in (7.12) and (7.13), respectively.

$$[\mathbf{A}'_N]_{i,j} = \begin{cases} p_2(i-j), & i=\{1,\cdots,N\}, j=\{1,\cdots,i\}, \\ 0, & \textit{otherwise.} \end{cases} \tag{7.12}$$

$$[\mathbf{A}_k]_{i,j} = \begin{cases} 1, & (i,j)=(1,1), \\ p_2(i-j), & i=\{2,\cdots,k\}, j=\{1,\cdots,i\}, \\ 0, & \textit{otherwise.} \end{cases} \tag{7.13}$$

For better understanding, examples of TPM for $N=3$ are given in Appendix I both in matrix and scalar forms.

Two major components of $\mathbf{Q}_N$ are successful transmission probabilities at node 1 and at node 2. In (7.9)-(7.10), we explicitly specify the probability at node 1, while representing the probability at node 2 with matrices $\mathbf{A}'_N$, $\mathbf{A}_k$, and $\mathbf{R}$. A transition from $(X_1^{(t)}=a, X_2^{(t)}=b)$ to $(X_1^{(t+1)}=a-s_1, X_2^{(t+1)}=b-s_2)$ occurs with probability $p_1(s_1)\cdot[\mathbf{A}_k]_{i,i-s_2}$, $[\mathbf{A}'_N]_{i,i-s_2}$, or $[\mathbf{R}]_{s_2,1}$ depending on the values of $X_1^{(t)}$ and $X_2^{(t)}$. In the matrix form, row $k+1$ and/or column $k+1$ of $\mathbf{Q}_N$ stand for a set of states for which $X_1^{(t)}=k$. Due to the constraint in (7.7), the size of $\mathbf{A}_k$ becomes smaller as $X_1^{(t)}$ becomes larger and vice versa. Correspondingly, first s_1 and s_1-1 columns of $[\mathbf{Q}_N]_{k+1,k+1-s_1}$ and $[\mathbf{Q}_N]_{k+1,1}$ are $\mathbf{0}$. Conditioned on $X_1^{(t)}=0$, the DTMC process will and will not finish at interval $t+1$ with probabilities $\mathbf{R}_N$ and $\mathbf{A}'_N$, respectively, whose entries depend only on the probability of successful transmission from node 2.

After obtaining all the matrices, we calculate all the main statistics, by using (7.3)-(7.6), where

$$\boldsymbol{f}_{\mathscr{T}}(t) = \boldsymbol{\alpha}_N \mathbf{Q}_N^{t-1} \mathbf{R}_N, \tag{7.14}$$

$$\boldsymbol{F}_{\mathscr{T}}(t) = \sum_{i=0}^{t} \boldsymbol{f}_{\mathscr{T}}(t), \tag{7.15}$$

$$\boldsymbol{E}[\mathscr{S}] = \boldsymbol{\alpha}_N (\mathbf{I}-\mathbf{Q}_N)^{-1} \mathbf{R}_N, \tag{7.16}$$

$$\boldsymbol{E}[\mathscr{T}] = \boldsymbol{\alpha}_N (\mathbf{I}-\mathbf{Q}_N)^{-2} \mathbf{R}_N. \tag{7.17}$$

7.2.2 Probabilistic Retransmission with Finite Persistence ARQ (ARQ^{FP})

In this section, we divide the analysis into two parts. First, we model the evolution of the transmission counter[5] and the route failure declaration process. Then, we incorporate the buffer dynamics into the model and formulate all relevant matrices.

7.2.2.1 Modeling Transmission Counter

Consider an absorbing DTMC with transient states being the transmission counter of node n and an absorbing state representing route failure declared by node n. According to ARQ^{FP}, we formulate the initial probability vector ($\boldsymbol{\beta}$), the transient TPM ($\mathbf{T}_n$), and the absorbing TPM ($\mathbf{F}_n$) as follows:

$$\boldsymbol{\beta} = (1, \mathbf{0}), \tag{7.18}$$

$$\mathbf{T}_n = \sum_{s=0}^{M} \mathbf{T}_n(s), \tag{7.19}$$

$$[\mathbf{T}_n(0)]_{i,j} = \begin{cases} p_n(0) \cdot (1 - \xi_n(i)), & i = j-1, j = \{2, \cdots, K_n+1\}, \\ 0, & otherwise, \end{cases} \tag{7.20}$$

$$[\mathbf{T}_n(s)]_{i,j} = \begin{cases} p_n(s), & i = \{1, \cdots, K_n+1\}, j = 1, s = \{1, \cdots, M\}, \\ 0, & otherwise, \end{cases} \tag{7.21}$$

$$[\mathbf{F}_n]_{i,1} = \begin{cases} p_n(0) \cdot \xi_n(i), & i = \{1, \cdots, K_n\}, \\ 1, & i = K_n + 1, \\ 0, & otherwise, \end{cases} \tag{7.22}$$

where M is the number of channel states, K_n is the maximum number of allowable retransmissions specified by ARQ^{FP} for node $n \in \{1, 2\}$, and $\xi_n(i)$ is the route failure declaration probability when node n experiences i consecutive transmission failures. The TPM $\mathbf{T}_n(s)$ with size $(K_n+1)^2$ represents the change in the transmission counter when s packets are successfully transmitted and node n does not declare a route failure. With TPM $\mathbf{F}_n$ whose size is $(K_n + 1) \times 1$, all transmitted packets do not reach the next node in that hop and node n declares a route failure. Since the transmission counter always starts from 0, entries of $\boldsymbol{\beta}$ with size $1 \times (K_n + 1)$ is one

[5]The transmission counter is greater than the retransmission counter by one.

in the first column and zero elsewhere, regardless of n. Since the counters at both the nodes are independent, their evolution follows $\mathbf{Y}_1 \otimes \mathbf{Y}_2$, where $\otimes$ denotes Kronecker's product, and $\mathbf{Y}_n$ is $\mathbf{T}_n$ when node n does not declare a route failure and is $\mathbf{F}_n$ when node n declares a route failure.

7.2.2.2 Modeling ARQ^{FP}

If a route failure is declared by node 1, the buffer contents at node 1 (corresponding to the tagged batch) will be flushed and the process will continue with $\boldsymbol{X}^{(t+1)} = (0, X_2^{(t+1)}, X_3^{(t+1)})$. If a route failure occurs at node 2, the packets from node 1 and node 2 will not be able to reach node 3, and the process will finish in interval $t+1$ with $\boldsymbol{X}^{(t+1)} = (0, 0, X_3^{(t+1)})$.

Let $\mathcal{T}_N$ be a state space containing all the reachable states in the DTMC for ARQ$^{\infty}$ with batch size N, where (7.7) is satisfied. The DTMC for ARQ^{FP} starts off at the last state of $\mathcal{T}_N$ $(N, 0, 0)$, and stays in the space $\mathcal{T}_N$ until a route failure is declared. Assuming that x packets are dropped, the process moves to $\mathcal{T}_{N-x}$, where $\sum_{n=1}^{3} X_n^{(t)} = N - x$. Therefore, the entire state space for ARQ^{FP} is $\mathcal{T}_0 \bigcup \mathcal{T}_1 \bigcup \mathcal{T}_2 \bigcup \cdots \bigcup \mathcal{T}_N$.

With ARQ^{FP}, the batch transmission process finishes when the buffers of node 1 and node 2 are empty, or in the interval t where $\boldsymbol{X}^{(t)} = (0, 0, X_3^{(t)}), X_3^{(t)} \in \{0, 1, \cdots, N\}$. We model ARQ^{FP} by using a DTMC with $N+1$ absorbing states, each of which corresponds to a certain value of $X_3^{(t)}$ when the process finishes. We arrange the states by grouping all absorbing states as first $N+1$ states and concatenate them with $(\mathcal{T}_1', \mathcal{T}_2', \cdots, \mathcal{T}_N')$, where $\mathcal{T}_y' = \mathcal{T}_y - (0, 0, y)$. In effect, ARQ^{FP} is represented by a DTMC with state space $\mathcal{T}_N^{FP} = ((000), (001), \cdots, (00N), \mathcal{T}_1', \mathcal{T}_2', \cdots, \mathcal{T}_N')$.

In general, to incorporate the transmission counters of both nodes, we need to inherit $(K_1+1)(K_2+1)$ states into each state in $\mathcal{T}_N^{FP}$. However, when $X_n^{(t)} = 0$, node n does not need to track the transmission counter. Therefore, each buffer state in ARQ^{FP} consists of $\prod_{X_n^{(t)}>0}(K_n+1)$ sub-states. The formulation of initial probability vector ($\boldsymbol{\alpha}_N^{FP}$), transient TPM ($\boldsymbol{\Omega}_N^{FP}$), and absorbing TPM ($\boldsymbol{\omega}_N^{FP}$) for ARQ^{FP} is similar to that for ARQ$^{\infty}$, except that $p_n(s)$ is replaced with $\mathbf{T}_n(s)$ or $\mathbf{F}_n$. Mathematically,

$$\boldsymbol{\alpha}_N^{FP} = (0,0,\cdots,0,1), \tag{7.23}$$

$$\mathbf{P}_N^{FP} = \left(\begin{array}{c|c} \mathbf{I} & \mathbf{0} \\ \hline \boldsymbol{\omega}_N^{FP} & \boldsymbol{\Omega}_N^{FP} \end{array}\right)$$

$$= \left(\begin{array}{c|ccc} \mathbf{I} & \mathbf{0} & \cdots & \\ \hline \mathbf{U}_1 & \mathbf{Q}_1^{FP} & \mathbf{0} & \cdots \\ \mathbf{U}_2 & \mathbf{V}_{21} & \mathbf{Q}_2^{FP} & \ddots \\ \vdots & \vdots & \vdots & \ddots \\ \mathbf{U}_N & \mathbf{V}_{N1} & \mathbf{V}_{N2} & \cdots \quad \mathbf{Q}_N^{FP} \end{array}\right), \quad \text{where} \tag{7.24}$$

$$[\mathbf{Q}_k^{FP}]_{i,j} = \begin{cases} \mathbf{A}_k^{FP'}, & i=j=1, \\ \mathbf{T}_1(0) \otimes \mathbf{A}_{k-i+2}^{FP''}, & i=j=\{2,\cdots,k+1\}, \\ (\mathbf{0}, (\mathbf{T}_1(i-j)\cdot \boldsymbol{e}) \otimes \mathbf{A}_{k-i+2}^{FP}), & i=\{2,\cdots,k+1\}, j=1, \\ (\mathbf{0}, \mathbf{T}_1(i-j) \otimes \mathbf{A}_{k-i+2}^{FP}), & i=\{2,\cdots,k+1\}, j=\{2,\cdots,i\}, \end{cases} \tag{7.25}$$

$$[\mathbf{A}_k^{FP'}]_{i,j} = \begin{cases} \mathbf{T}_2(i-j), & i=\{1,\cdots,k\}, j=\{1,\cdots,i\}, \\ 0, & \textit{otherwise}, \end{cases} \tag{7.26}$$

$$[\mathbf{A}_h^{FP''}]_{i,j} = \begin{cases} 1, & (i,j)=(1,1), \\ \mathbf{T}_2(i-j)\cdot \boldsymbol{e}, & i=\{2,\cdots,h\}, j=1, \\ \mathbf{T}_2(i-j), & i=\{2,\cdots,h\}, j=\{2,\cdots,i\}, \\ 0, & \textit{otherwise}, \end{cases} \tag{7.27}$$

$$[\mathbf{A}_h^{FP}]_{i,j} = \begin{cases} \boldsymbol{\beta}, & (i,j)=(1,1), \\ \mathbf{T}_2(i-j), & i=\{2,\cdots,h\}, j=\{1,\cdots,i\}, \\ 0, & \textit{otherwise}, \end{cases} \tag{7.28}$$

$$[\mathbf{V}_{kl}]_{i,j} = \begin{cases} \mathbf{F}_1 \otimes \mathbf{T}_2(x_2-j), & i=\frac{k(k+3)-l(l+3)}{2}+1+x_2, \\ & x_2=\{1,\cdots,l\}, j=\{1,\cdots,x_2\}, \\ \mathbf{0}, & \textit{otherwise}, \end{cases} \tag{7.29}$$

$$\mathbf{U}_k = \left(\begin{array}{c|c} \begin{array}{c} \mathbf{u}_k' \\ \mathbf{u}_k \\ (\mathbf{u}_{k-1}, \mathbf{0}) \\ \vdots \\ (\mathbf{u}_2, \mathbf{0}) \\ (\mathbf{u}_1, \mathbf{0}) \end{array} & (\mathbf{R}_k^{FP}, \mathbf{0}) \end{array}\right), \tag{7.30}$$

$$[\mathbf{R}_k^{FP}]_{i,1} = \begin{cases} \mathbf{T}_2(i)\cdot \boldsymbol{e}, & i=\{1,\cdots,k\}, \\ 0, & \textit{otherwise}, \end{cases} \tag{7.31}$$

$$\mathbf{u}_k(i,j) = \begin{cases} \mathbf{F}_1, & (i,j) = (1,k), \\ \mathbf{F}_1 \otimes \mathbf{T}_2(i-1)\cdot \boldsymbol{e}, & i = \{2,\cdots,k\}, j = k, \\ \mathbf{F}_2, & i = \{2,\cdots,k\}, j = k+1-i, \\ \mathbf{0}, & otherwise, \end{cases} \tag{7.32}$$

$$\mathbf{u}'_k(i,j) = \begin{cases} \mathbf{F}_2, & i = \{1,\cdots,k\}, j = k+1-i, \\ 0, & otherwise, \end{cases} \tag{7.33}$$

The implications of the above matrices are as follows. First, $\mathbf{Q}_k^{FP}$ consists of probabilities that the process will stay in $\mathcal{T}_k$. The structure of $\mathbf{Q}_k^{FP}$ is similar to that of $\mathbf{Q}_k$, but $p_n(s)$ is replaced by $\mathbf{T}_n(s)$. We use a Kronecker's product operation to combine transmission counters of both node 1 and node 2. Again, we do not need to keep track of the counter for node n when $X_n^{(t)} = 0$. If $s = X_n^{(t)}$, the buffer size of node n in time $t+1$ will be zero. For node 1 and 2, this case corresponds to the first column of $\mathbf{Q}_k^{FP}$ and $\mathbf{A}_h^{FP''}$, respectively. For these entries, we multiply $\boldsymbol{e}$ to the right of $\mathbf{T}_n(s)$ to integrate all possibilities into one column. The first rows of these columns are the place where the process also starts from zero-buffer. Therefore, we also need to multiply $\boldsymbol{e}^T$ to the left of $\mathbf{T}_n(s)$. Correspondingly, the entries (1,1) of these matrices are one. When the packet first arrives, node 2 sets the transmission counter to zero. Accordingly, entry (1,1) of $\mathbf{A}_h^{FP}$ is $\boldsymbol{\beta}$.

Secondly, entries (i,j) of $\mathbf{V}_{kl}$ stand for a route failure at node 1, and j packets successfully transmitted by node 2. Therefore, $[\mathbf{V}_{kl}]_{i,j} = \mathbf{F}_1 \otimes \mathbf{T}_2(X_2^{(t)} - j)$. With $\mathbf{V}_{kl}$, $X_1^{(t)} = k - l$ packets are dropped from node 1, and the process moves from $\mathcal{T}_k$ to $\mathcal{T}_l$. Since all packets in node 1 are dropped, the non-zero entries of $\mathbf{V}_{kl}$ are located at the rows with $X_1^{(t)} = k - l$, which span over $\frac{k(k+3)-l(l+3)}{2} + 1 + X_2^{(t)}, X_2^{(t)} = \{1,\cdots,l\}$. Note that, the range of $X_2^{(t)}$ excludes 0, since the process will move to the absorbing states if $X_2^{(t)} = 0$.

Thirdly, given $X_1^{(t)} = 0$, $\mathbf{R}_k^{FP}$ is the TPM corresponding to the case that all packets in node 2 are successfully transmitted. Otherwise, if a route failure occurs at node 2, the process will finish with TPM $\mathbf{u}'_k$. Note that, since in the absorbing states $X_1^{(t)} = X_2^{(t)} = 0$, we need to multiply $\mathbf{T}_n(s_n)$ with $\boldsymbol{e}^T$ and/or $\boldsymbol{e}$. Finally, for $X_1^{(t)} \neq 0$, $\mathbf{u}_k$ represents three possibilities to finish the process: a route failure at node 2 ($\mathbf{F}_2$), a route failure at node 1 while $X_2^{(t)} = 0$ ($\mathbf{F}_1$), and a route failure at node 1 and all packets in node 2 are successfully transmitted ($\mathbf{F}_1 \otimes \mathbf{T}_2(i-1)\cdot \boldsymbol{e}$).

Similar to those in case of ARQ$^{\infty}$, examples of $\mathbf{\Omega}_N^{FP}$ and $\boldsymbol{\omega}_N^{FP}$ with $N = 3$ are given in Appendix I. With ARQ^{FP}, sizes of transient TPM $\mathbf{\Omega}_k^{FP}$ and absorbing TPM $\boldsymbol{\omega}_k^{FP}$ expand to $(L_N^{FP})^2$ and $L_N^{FP} \times (N+1)$, where $L_N^{FP} = N(N+1)[3(K_1+K_2+2)+(K_1+1)(K_2+1)(N-1)]/6$ (see Appendix J). Now, we replace $\boldsymbol{\alpha}_N$, $\mathbf{\Omega}_N$, and $\boldsymbol{\omega}_N$ in (7.14)-(7.17) with the formulated $\boldsymbol{\alpha}_N^{FP}$, $\mathbf{\Omega}_N^{FP}$, and $\boldsymbol{\omega}_N^{FP}$ respectively, and derive all main statistics from (7.3)-(7.6).

7.2.3 Non-zero Initial Packets in the Network

The model for ARQ^{FP} developed above is modified to support the case where the source and intermediate nodes have N_1 and N_2 packets and their current transmission counters are k_1 and k_2, respectively. Again, we need to derive three major matrices. We first observe that the initial condition $X_1^{(0)} = N_1$ and $X_2^{(0)} = N_2$ is one of the states in $\mathcal{T}_{N_1+N_2}^{FP}$. Therefore, the transient and the absorbing TPMs in this case are just $\mathbf{\Omega}_{N_1+N_2}^{FP}$ and $\boldsymbol{\omega}_{N_1+N_2}^{FP}$, respectively. Since the process starts with parameters N_1, N_2, k_1, and k_2, we set the entry of the initial probability vector ($\boldsymbol{\alpha}_{N_1+N_2}^{FP}$) in the state with the corresponding parameters to one and set all other entries of $\boldsymbol{\alpha}_{N_1+N_2}^{FP}$ to zero.

7.2.4 Computational Complexity

Calculation of (7.14)-(7.17) involves matrix multiplications and inversions which in general incur computational complexity of $O(L^3)$, where L is the size of the matrix. In our model, the worst case complexity is determined by $\mathbf{\Omega}_N^{FP}$ whose size is $L_N^{FP} = N(N+1)[3(K_1+K_2+2)+(K_1+1)(K_2+1)(N-1)]/6$ (see Appendix J). For brevity, we do not explicitly quantify the computational complexity for each step in the calculation. Rather, we show how the complexity of the calculation reduces compared to that for the worst case due to the special structure of the above formulation.

First, the initial probability vector ($\boldsymbol{\alpha}_N^{FP}$) contains only one non-zero entry. Therefore, the result of the multiplication ($\boldsymbol{\alpha}_N^{FP} \cdot \mathbf{A}$) is simply the row of $\mathbf{A}$ which corresponds to the non-zero entry of $\boldsymbol{\alpha}_N^{FP}$. Next, since $\mathbf{\Omega}_N^{FP}$ is a lower-triangular matrix, the complexity of self-multiplication of $\mathbf{\Omega}_N^{FP}$ and inversion of $(\mathbf{I} - \mathbf{\Omega}_N^{FP})$ (by means of *back substitution*) is reduced by half. Finally, since non-zero entries are sparsely populated in the lower triangular part of $\mathbf{\Omega}_N^{FP}$, the complexity can be reduced further

by using sparse matrix operations [84].

7.3 Numerical and Simulation Results

This section presents numerical results mainly for the two basic performance metrics – end-to-end latency ($\mathscr{D}_{e2e}$) and number of successfully delivered packets ($\mathscr{S}$) – from which other metrics such as throughput or reliability can be calculated. We assume that the wireless channel states are equally-likely and are the same for both the hops (i.e., $\pi_m = 1/M, m \in \{1, \cdots, M\}$ and $p_{err}(1) = p_{err}(2) = p_{err}$). Unless otherwise specified, we assume that $N_1 = N$ and $N_2 = 0$. Each node in the network is assumed to implement the same variant of ARQ protocol. For brevity, we only consider ARQ^0, ARQ$^\infty$, and ARQ^P with $\xi_1 = \xi_2 = \xi$. Numerical results for other cases can also be obtained by adjusting the parameters in the above ARQ models accordingly.

7.3.1 Model Validation

We validate our model by simulating a batch transmission with N packets provided to node 1. We measure the end-to-end latency ($\mathscr{D}_{e2e}$) and the number of successfully delivered packets ($\mathscr{S}$), and average them over 10^6 samples. In each transmission interval, the channel state is simulated based on $(\pi_1, \pi_2, \cdots, \pi_M)$, where M is the number of channel states. When the channel state is m, each of the m transmitted packets is assumed to be in error with probability p_{err}.

We set $p_{err} = 0.1$ and $N = \{1, 2, \cdots, 10\}$, and plot expected latency ($E[\mathscr{D}_{e2e}]$) and expected number of successfully delivered packets ($E[\mathscr{S}]$) in Figure 7.2 (for $M = 1$) and in Figure 7.3 (for $M = N$). Also, we plot $E[\mathscr{D}_{e2e}]$ for ARQ$^\infty$ in Figure 7.4, where $N = M$ and $p_{err} = \{0, 0.1, 0.2, 0.3\}$. We observe that the simulation results closely follow the numerical results obtained from the developed analytical model.

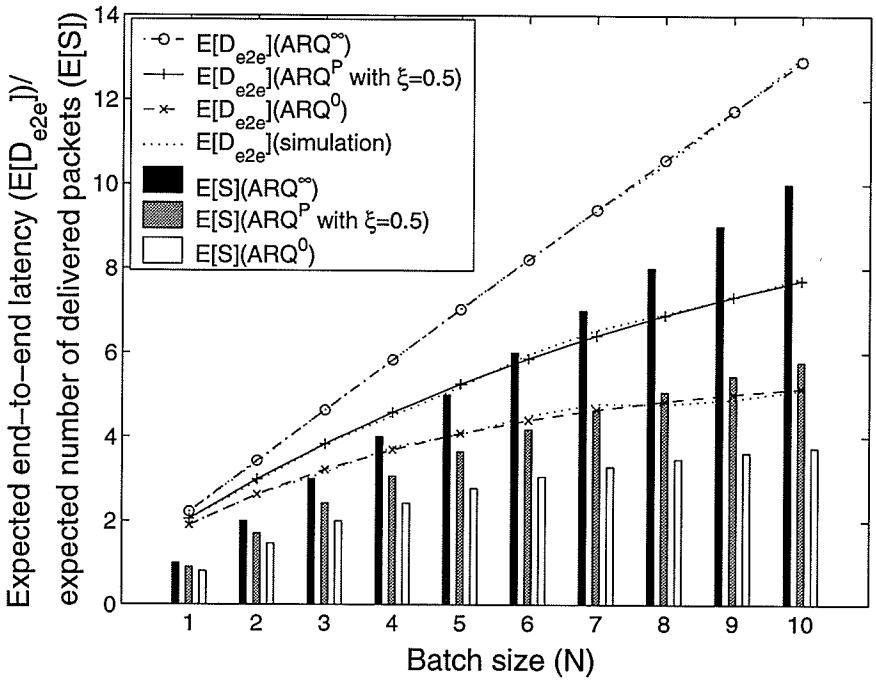


Figure 7.2. *Variations in expected end-to-end latency ($E[\mathscr{D}_{e2e}]$) and expected number of successfully delivered packets ($E[\mathscr{S}]$) when $M = 1$.*

7.3.2 Expected End-to-End Latency and Number of Successfully Delivered Packets

7.3.2.1 Performance Bounds

For ARQ$^\infty$, $E[\mathscr{D}_{e2e}]$ is the time required for all the packets to reach the destination. Therefore, the minimum latency for $M = 1$ is $N + 1$ transmission intervals. With all other ARQ policies, batch transmission can finish earlier due to a possible route failure. For example, if node 1 implements ARQ^0 and if node 2 does not receive any packet by the end of the first interval, the transmission will finish at the end of the first interval. For a general ARQ^P policy with $0 \leqslant \xi \leqslant 1$, ARQ$^\infty$ (i.e., $\xi = 0$) and ARQ^0 (i.e., $\xi = 1$) provide upper bound and lower bound, respectively, for end-to-end latency. Intuitively, the expected number of successfully delivered packets is the largest for ARQ$^\infty$ (where $S = N$) and the smallest for ARQ^0 (Figures 7.2-7.3).

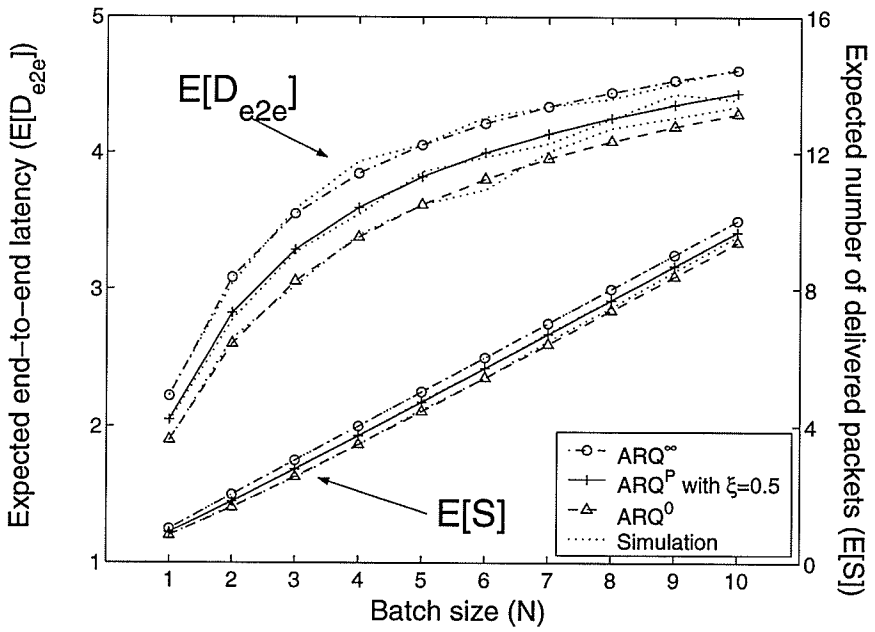


Figure 7.3. *Variations in expected end-to-end latency ($E[\mathscr{D}_{e2e}]$) and expected number of successfully delivered packets ($E[\mathscr{S}]$) when $M = N$.*

7.3.2.2 Effect of Batch Size

With ARQ$^{\infty}$, when $M = 1$, each packet incurs the same amount of end-to-end latency. Therefore, increasing batch size causes a linear increase in the expected end-to-end latency (Figure 7.2). For ARQ^0, packet i is dropped not only because it is not received by the next node in the path but also because packet k (where $k < i$) was dropped. Essentially, packet i has higher unconditional dropping probability and incurs less latency (per packet) than packet k ($k < i$) does. As a result, the rate of increase in both $E[\mathscr{D}_{e2e}]$ and $E[\mathscr{S}]$ for ARQ^P with $\xi > 0$ is a decreasing function of N.

7.3.2.3 Effect of Packet Error Probability

With $p_{err} = 0$ and fixed transmission rate of $(M + 1)/2$, each node requires two transmission intervals to transmit all packets in its buffer and the end-to-end latency is three transmission intervals. For an equally-likely channel with a small number of channel states (e.g., 1 or 2), both node 1 and node 2 might be able to transmit several

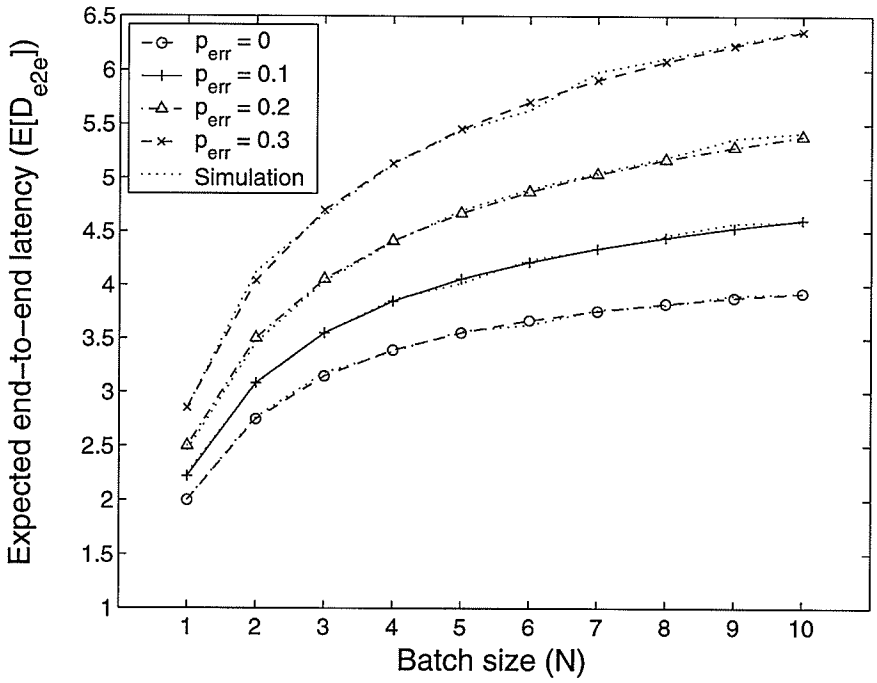


Figure 7.4. *Variations in expected end-to-end latency ($E[\mathscr{D}_{e2e}]$) for ARQ^{∞} with $M = N$.*

packets when their buffers are full, and might experience *bad* channel condition when their buffers are empty. We observe in Figure 7.4 (ARQ^{∞} with $M = N$) that with $p_{err} = 0$ the expected latency can be less than that obtained when the transmission rate is fixed to the average transmission rate. For example, $E[\mathscr{D}_{e2e}] = 2$ for $M = 1$ and $E[\mathscr{D}_{e2e}] < 3$ for $M = 2$. However, increasing M increases the possibility that a node will not transmit at the highest rate. Therefore, some packets might be left in the buffer of node 1 after the first transmission interval. In subsequent intervals, node 1, node 2, or both might have fewer packets than it can actually transmit (i.e., $X_n^{(t)} < m$, where m is the current transmission rate). Since the node cannot fully utilize the available transmission rate, we observe that the actual average end-to-end latency is greater than three transmission intervals for $M > 2$. During these transmission intervals, the node may adjust the AMC mode such that the transmission rate does not exceed the number of available packets. This adjustment would reduce packet error probability without trading off the number of packets transmitted.

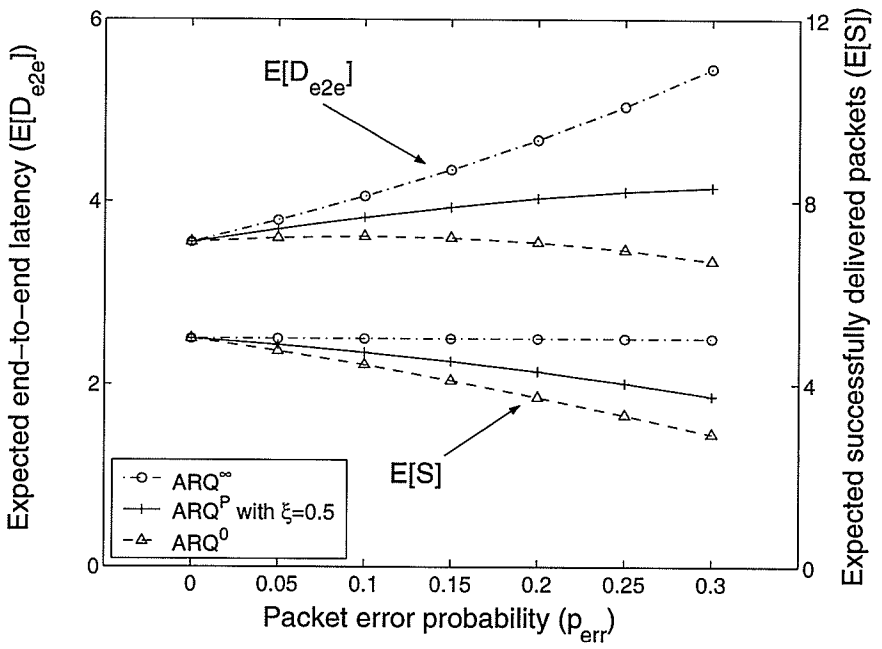


Figure 7.5. *Expected end-to-end latency ($E[\mathscr{D}_{e2e}]$) and expected number of successfully delivered packets ($E[\mathscr{S}]$) vs. packet error probability (p_{err}).*

Next, we set $M = N = 5$, vary p_{err} from 0 to 0.3, and plot the resulting $E[\mathscr{D}_{e2e}]$ and $E[\mathscr{S}]$ in Figure 7.5. We observe that $E[\mathscr{D}_{e2e}]$ and $E[\mathscr{S}]$ for all the ARQ policies are equivalent when $p_{err} = 0$. While for ARQ$^{\infty}$ $E[\mathscr{S}]$ is independent of p_{err}, $E[\mathscr{S}]$ is a decreasing function of p_{err} for the two other ARQ policies. For ARQ$^{\infty}$, increasing p_{err} always increases $E[\mathscr{D}_{e2e}]$. However, with ARQ^{P}, especially for large ξ (e.g., $\xi = 1$), increasing p_{err} increases $E[\mathscr{D}_{e2e}]$ only initially, and further increase in p_{err} decreases the end-to-end latency due to more frequent route failures.

7.3.2.4 Effect of Number of Channel States

From Figures 7.2-7.3, we notice that increasing the maximum transmission rate (M) decreases $E[\mathscr{D}_{e2e}]$, since packets can be delivered faster. It also prevents route failure and increase reliability, because more packets are transmitted during a transmission interval. However, from Figure 7.6 ($N = 10$, $p_{err} = 0.1$, and $M = \{1, \cdots, 10\}$), we realize that this may not always be true.

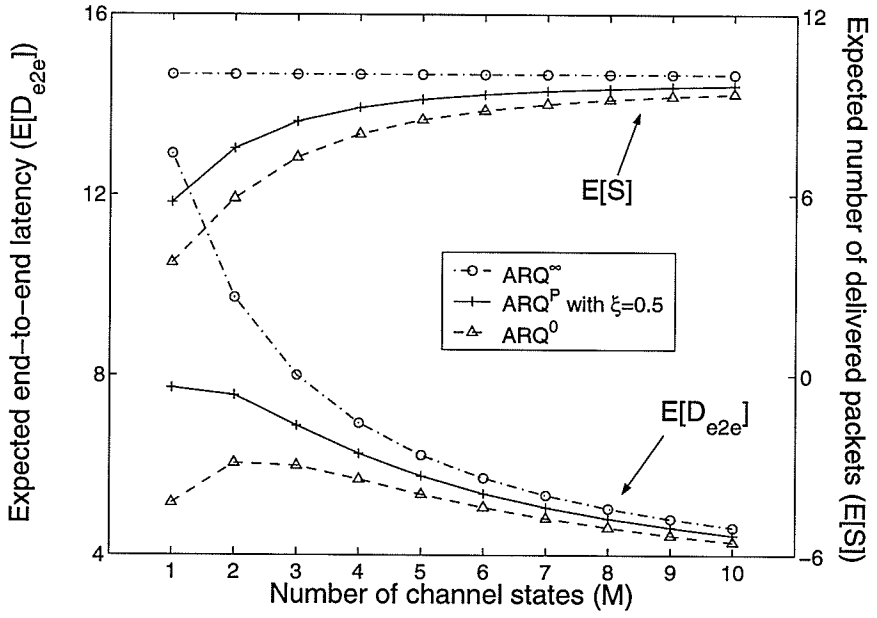


Figure 7.6. *Variation in expected end-to-end latency ($E[\mathscr{D}_{e2e}]$) and expected number of successfully delivered packets ($E[\mathscr{S}]$) for $N = 10$.*

For ARQ$^{\infty}$, $E[\mathscr{S}]$ is not affected by M, while $E[\mathscr{D}_{e2e}]$ decreases exponentially with increasing M. For large M, the transmission rate is under-utilized, and the rate of decrease of $E[\mathscr{D}_{e2e}]$ is not as fast as the rate for small M. Similarly, for ARQ^0 and ARQ^P, increasing M increases $E[\mathscr{S}]$. For small ξ (e.g., $\xi = 0.5$), the same trend (as of ARQ$^{\infty}$) for $E[\mathscr{D}_{e2e}]$ is observed for ARQ^P. With large ξ (e.g., $\xi = 1$, or ARQ^0), increasing M initially increases the latency, because the process is less likely to finish earlier due to less frequent route failures. As M increases further, $E[\mathscr{D}_{e2e}]$ begins to decrease because all the packets tend to be delivered faster. In Figure 7.6, we can observe that both low latency and high-reliability can be achieved if M is increased beyond 2. Nevertheless, if the range of M is confined to $M \leqslant 2$, higher number of channel states will lead to improved end-to-end reliability at the expense of increasing latency which might not be preferable to delay-sensitive applications.

7.3.2.5 Initial Number of Packets in the Network Path

With fixed N_1, the effect of N_2 is similar to that of batch size. In particular, increasing N_2 increases $E[\mathscr{D}_{e2e}]$ and $E[\mathscr{S}]$ with decreasing slope. However, the slope is less than that due to increasing batch size, since the packets (in the intermediate node) have to traverse only one hop. This result is quite expected.

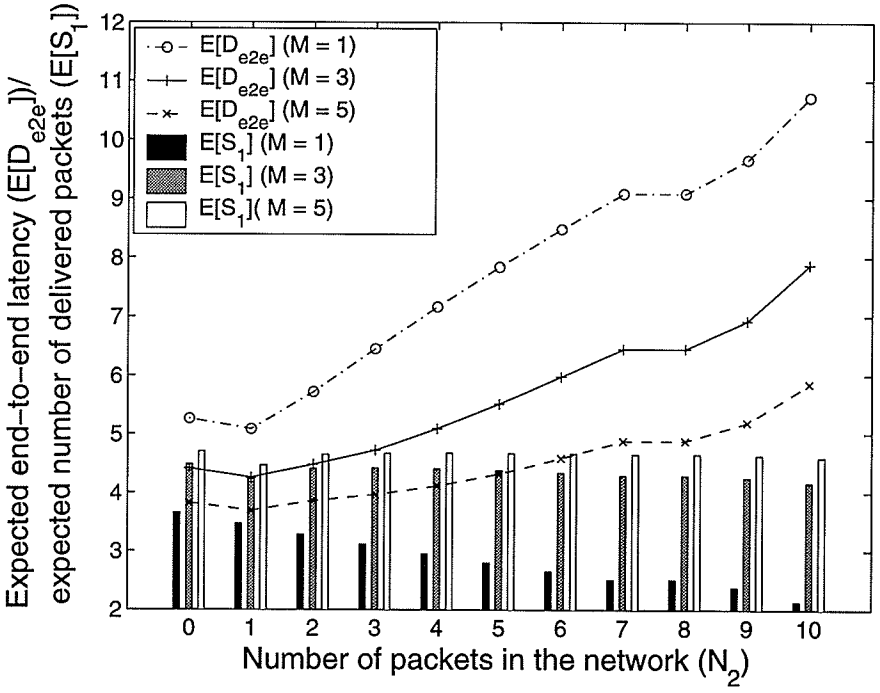


Figure 7.7. *Variation of expected end-to-end latency ($E[\mathscr{D}_{e2e}]$) and expected number of successfully delivered packets ($E[\mathscr{S}_1]$) as a function of initial number of packets in the network path (N_2) for $N_1 = 5$.*

For a tagged batch of size N_1 packets, we define the expected number of successfully delivered packets as $E[\mathscr{S}_1] = \sum_{s=N_2+1}^{N_2+N_1}(s - N_2)f_{\mathscr{S}}(s)$. The expected latency in this case is still the same, since it is the time to get all the packets out of the network path. In Figure 7.7, we plot $E[\mathscr{D}_{e2e}]$ and $E[\mathscr{S}_1]$ for the current batch of $N_1 = 5$ packets as a function of $N_2 = \{0, 1, \cdots, 15\}$ and $M = \{1, 3, 5\}$ when $p_{err} = 0.1$. We observe that the effect of N_2 is opposite to that of the number of channel states. Initially, increasing N_2 increases route failure probability at node 2 and decreases

batch latency. Further increase in N_2 leads to an increase in latency since there are more number of packets to be delivered. For $M = 1$, increasing N_2 always results in a decrease in reliability due to route failure. However, for $M > 1$, the reliability for $N_2 = 2$ is greater that that for $N_2 = 1$, since node 2 might transmit more than one packet during a transmission interval, and one of the transmitted packets might be delivered successfully. In this case, the route failure is less likely to happen, and the reliability improves. The reliability degrades again as N_2 increases beyond M. In this case, although there are enough packets for transmission, node 2 cannot transmit all of them, and the left-over packets might cause route failure in subsequent transmission intervals.

7.3.3 Expected End-to-End Throughput

We define expected end-to-end throughput as $E[\gamma_{e2e}] = E[\mathscr{S}]/E[\mathscr{D}_{e2e}]$. With $p_{err} = 0.1$, we plot $E[\gamma_{e2e}]$ in Figure 7.8 for different ARQ variants for different batch sizes and transmission rates. For ARQ$^{\infty}$, with $M = 1$ and $p_{err} = 0$, $E[\gamma_{e2e}]$ and the rate of increase of $E[\gamma_{e2e}]$ with respect to batch size N are given by $N/N+1$ and $(N+1)^{-2}$, respectively. We can observe the same trend for $p_{err} = 0.1$ and $M = 1$ in Figure 7.8, where $E[\gamma_{e2e}]$ is an increasing function (with decreasing slope) of N.

Although increasing instantaneous transmission rate beyond $X_n^{(t)}$ does not decrease end-to-end latency, increasing maximum transmission rate increases the chance to transmit more packets per interval, therefore decreasing route failure probability and increasing $E[\gamma_{e2e}]$. In Figure 7.8, we observe that, with fixed M (where $M = 1$, 5, or 10) and $N < M$, $E[\gamma_{e2e}]$ is larger than that with variable M (where $M = N$).

When M is small, ARQ$^{\infty}$ does not under-utilize the radio resource and has higher $E[\gamma_{e2e}]$ than ARQ^0 which causes frequent packet drops. ARQ$^{\infty}$ and ARQ^0 provide upper-bound and lower-bound, respectively, for $E[\gamma_{e2e}]$ when M is small. For a system with large M (e.g., $M = 10$), when N is also large, ARQ$^{\infty}$ suffers from transmission-rate under-utilization, while ARQ^0 and ARQ^P with large ξ avoid this situation by dropping only few packets to terminate the process earlier. We can observe that for $M = 10$ and $N \geqslant 4$, $E[\gamma_{e2e}]$ for ARQ$^{\infty}$ is not as high as that for ARQ^0 or ARQ^P. Clearly, this trade-off between reliability ($E[\mathscr{S}]$) and latency ($E[\mathscr{D}_{e2e}]$) implies that we can adjust ξ in ARQ^P to maximize the expected throughput.

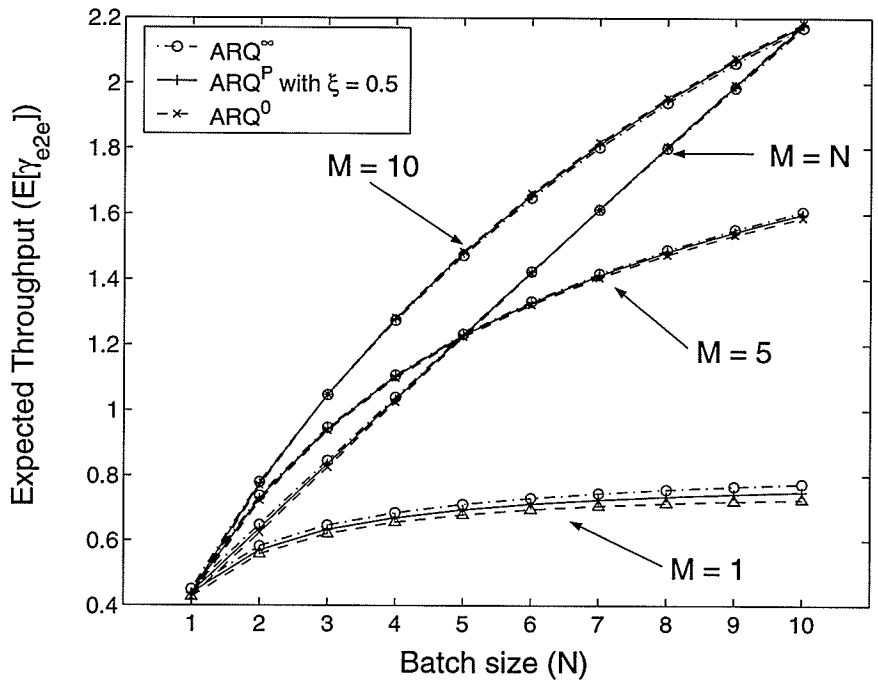


Figure 7.8. *Typical variations in expected end-to-end throughput.*

7.3.4 Probability Mass Function of End-to-End Latency

Figure 7.9 plots the *cdf* of end-to-end latency, $F_{\mathscr{D}}(d) = \sum_{i=1}^{d} f_{\mathscr{D}}(i)$, for $N = 5$, $M = 1$, and $p_{err} = 0.1$ for the above three ARQ policies. For ARQ$^{\infty}$, it is impossible for the process to complete before $N + 1$ intervals (i.e., $f_{\mathscr{D}_{e2e}}(d) = 0, d \leqslant N$). Therefore, it provides an upper-bound for the end-to-end latency. For both ARQ^0 and ARQ^P, the *cdf* starts to increase when $d \geqslant 1$. When $d \leqslant N$, the process can finish only because all the packets are dropped, while for $d > N$ the process can finish because all packets are dropped and/or delivered to the destination node. Due to the small packet error rate, the *cdf* for $d > N$ converges to 1 at a much faster rate than it does when $d \leqslant N$.

After $\lceil E[\mathscr{D}_{e2e}] \rceil$ (= 5 for ARQ^P in Figure 7.9) intervals, the probability that the batch transmission process with ARQ^P has finished is only 35.7%. Therefore, there is a very high probability that some packets are still in the network path after $\lceil E[\mathscr{D}_{e2e}] \rceil$. In other words, the expected end-to-end latency may not be a *good* metric to measure the quality of a network path.

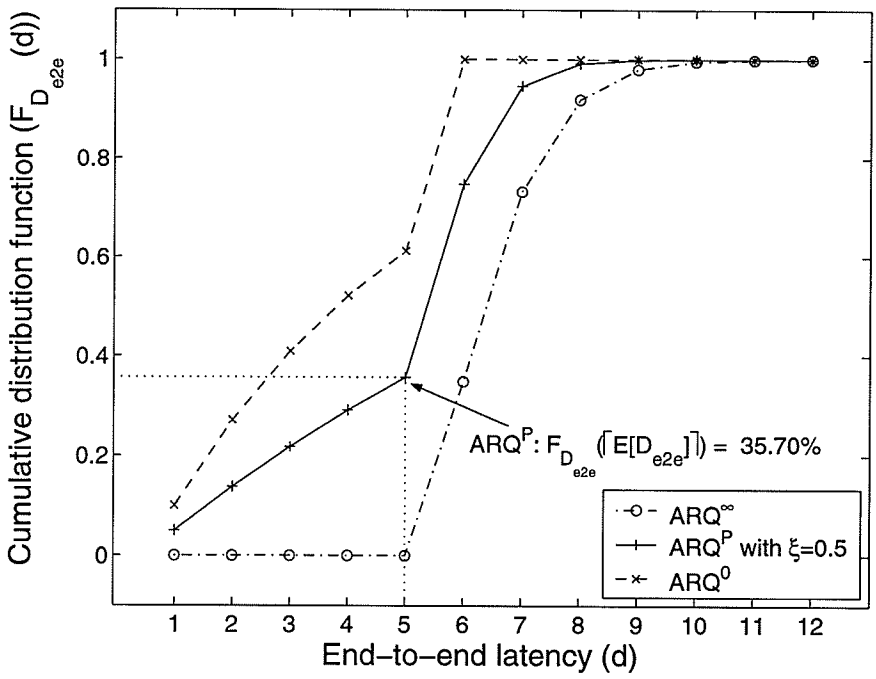


Figure 7.9. *Cumulative distribution function of end-to-end latency for* $M = 1$.

Next, we set $M = N$ and plot the *cdf* of end-to-end latency in Figure 7.10. All other parameter settings are the same as those in Figure 7.9. We observe that for all the ARQ policies the *cdf* increases sharply when $d > 2$ because the route failure probability decreases due to higher transmission rate. Compared to the case with $M = 1$, the probability that there is no packet left in the network after $\lceil E[\mathscr{D}_{e2e}] \rceil$ intervals is as high as 78.85%. This observation implies that multi-rate transmission not only improves $E[\mathscr{S}]$, $E[\mathscr{D}_{e2e}]$, and $E[\gamma_{e2e}]$, but also helps avoid congestion in the network path.

Clearly, the expected latency per se would not be a very useful metric for an end-to-end flow control mechanism. Instead, the *cdf* of the end-to-end latency can be exploited in controlling the end-to-end transmission (e.g., setting up the timeout value at the sender). For example, at the sending node a timeout may be triggered after all the packets have left the network with probability 95%. With $M = N$, the minimum number of transmission intervals (which is, in general, larger than $E[\mathscr{D}_{e2e}]$) that ARQ$^\infty$ requires to achieve this probability is plotted in Figure 7.11 for different

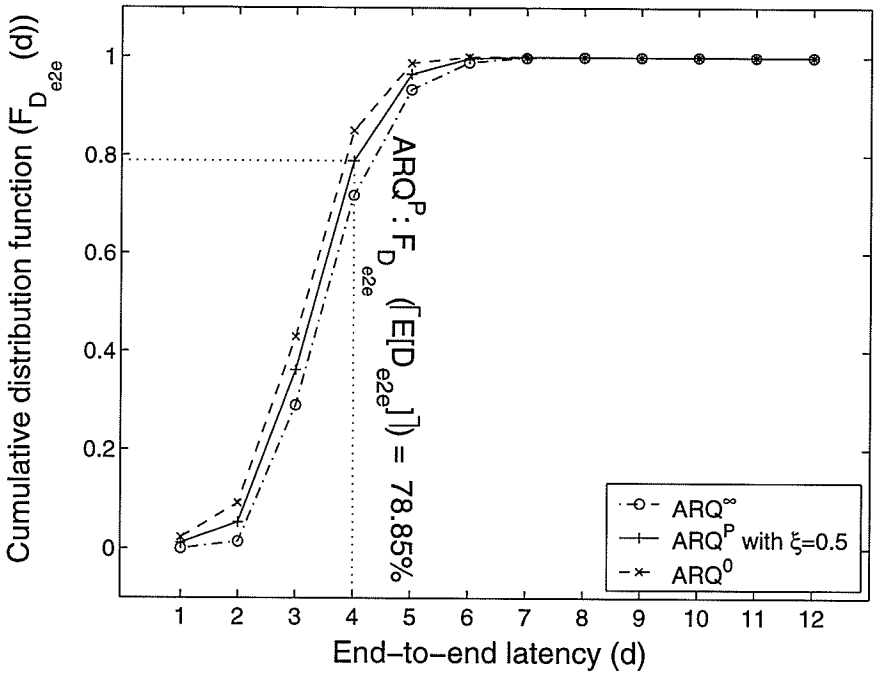


Figure 7.10. *Cumulative distribution function of end-to-end latency for $M = N$.*

packet error rate.

7.3.5 Probability Mass Function of the Number of Delivered Packets

For ARQ^0 with $M = 1$, s ($s < N$) packets will be successfully delivered to the destination node with probability $f_{\mathscr{S}}(s) = (1-p_{err})^{2s}(1-(1-p_{err})^2)$. Literally, first s packets must be successfully transmitted in both the hops and no more packets are successfully delivered to the destination. For $s = N$, transmission of all packets must again be successful in both the hops, and the corresponding probability for this case is $f_{\mathscr{S}}(N) = (1-p_{err})^{2N}$. Note that, the term $(1-(1-p_{err})^2)$ vanishes because the buffers at both the nodes are empty after N packets are successfully transmitted. For ARQ^0, $f_{\mathscr{S}}(s)$ is convex in s as can be observed in Figure 7.12 where $N = 5$. For small p_{err}, the transmission of the entire batch is successful with high probability and $f_{\mathscr{S}}(s)$ is strictly convex ($f_{\mathscr{S}}(N) > f_{\mathscr{S}}(N-1)$) because the term $1-(1-p_{err})^2$ appreciably

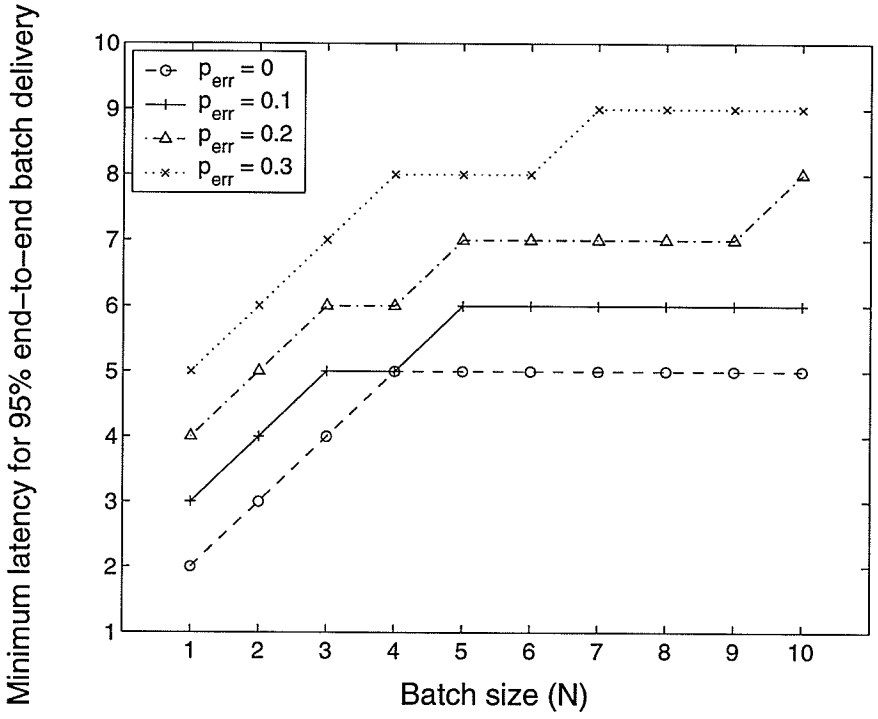


Figure 7.11. *Minimum latency for 95% end-to-end batch delivery in a reliable batch transmission (ARQ^{∞}).*

scales down $f_{\mathscr{S}}(s)$ for $s < N$. When $p_{err} = 0$, the *pmf*s corresponding to all ARQs converge to each other, where $f_{\mathscr{S}}(N) = 1$ and $f_{\mathscr{S}}(s) = 0, s < N$.

For $p_{err} > 0$, if ARQ^{∞} is employed, all N packets will be delivered to the destination node. Therefore, ARQ^{∞} provides the upper-bound for $E[\mathscr{S}]$, while the corresponding lower-bound can be obtained when ARQ^{0} is used. For ARQ^{P}, the effect of increasing M and/or decreasing ξ on $f_{\mathscr{S}}(s)$ is similar to that due to decreasing p_{err} because both result in fewer number of route failures and higher number of delivered packets. We omit these intuitive results for brevity.

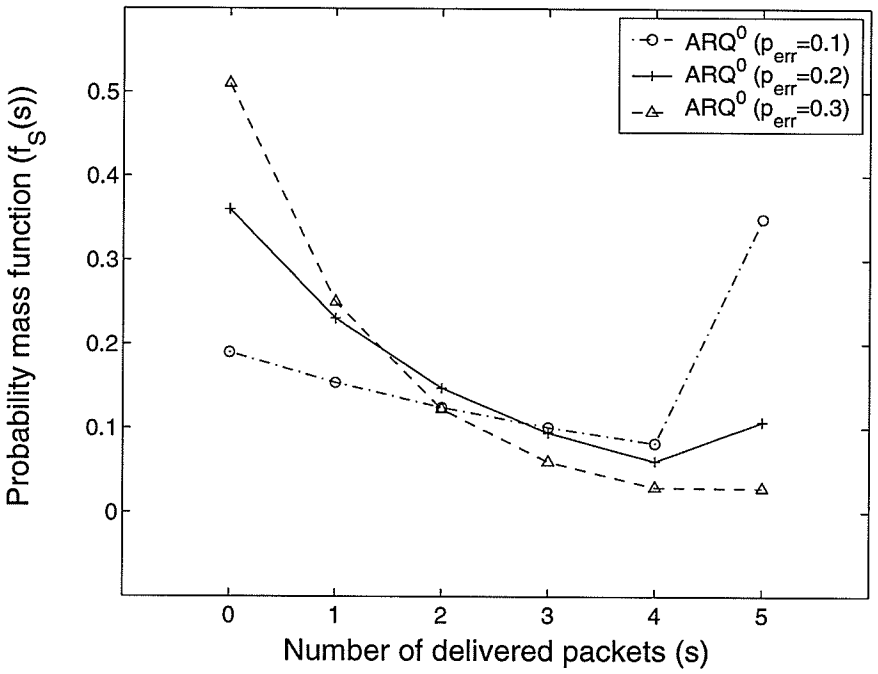


Figure 7.12. *Probability mass function of the number of packets successfully delivered to the destination node ($f_{\mathcal{S}}(s)$) for $N = 5$ and $M = 1$.*

7.4 Application of the Model: Estimating TCP Performance in a Multi-hop Wireless Network

In this section, we demonstrate the usefulness of the proposed analytical model in estimating the performance of TCP across a multi-hop wireless path. Since TCP is a window-based flow control protocol, after transmitting a window of packets, the TCP sender waits for an acknowledgement (ACK) from the destination. Each acknowledgement opens the congestion window, and allows the sender to transmit another batch of packets. In general, packets from the previous transmission window might still be in the network path when an ACK arrives at the sender. At this point, the sender begins a new batch of transmission with size N_1, while there are N_2 packets in the network path.

We define *batch reliability* as the percentage of delivered packets from a batch of N_1 packets. The batch reliability is equivalent to transmission reliability (i.e.,

$\mathscr{S}/N$) when $N_2 = 0$. For $N_2 \neq 0$, when the packets are delivered in order[6], the batch reliability is just $(\mathscr{S} - N_2)/N_1$. If $N_2 \neq 0$ and an in-sequence packet delivery cannot be guaranteed, we will need to determine which batch each of the successfully transmitted packets belongs to. Analysis of this case with out-of-sequence packet delivery is left for future studies. In the following experiments, we use TCP Reno [85] with $M = 1$, and therefore, the batch reliability is calculated from $(\mathscr{S} - N_2)/N_1$.

Using *ns-2* [70], we set up a three-node chain topology (as in Figure 7.1), and run one TCP flow over a two-hop wireless path. We assume that each mobile in the path implements ARQ^0 and has maximum transmission rate $M = 1$. Again, if a route failure occurs, the node will flush its buffer and will not receive any packet from other nodes. In the simulation, we vary the data rate (i.e., link capacity) in each hop from 10 to 500 packets per second (*pps*), and vary p_{err} from 0.01 to 0.3.

Due to TCP window dynamics, we measure average TCP packet latency, defined as $E[\mathscr{D}_{e2e}]/N_1$, rather than batch latency. We measure batch reliability as the ratio of the number of delivered TCP packets and the batch size. From the simulations, we observe that TCP timeouts occur fairly often. Under these circumstances, there is no buffer accumulation in the intermediate node, and the buffer size of node 2 at any time is zero or one. Based on the knowledge about average congestion window ($N_1 = \overline{W}$) (measured from the simulation) and buffer dynamics at node 2 (N_2), we predict the end-to-end latency ($E[\mathscr{D}_{e2e}]$) and batch reliability ($E[\mathscr{S} - N_2]/N_1$) by using our model. Since the average congestion window ($\overline{W}$) is not a whole number, the predicted value is interpolated between $N_1 = \lceil \overline{W} \rceil$ and $N_1 = \lfloor \overline{W} \rfloor$, and averaged over $N_2 = 0$ and $N_2 = 1$.

Figure 7.13 plots TCP packet latency and batch reliability obtained from the measurement and from our model. We plot batch reliability for data rate of 10 *pps* only, since the results for other data rate values are very close to each other. We observe that in most cases, the results from the analytical model (shown by symbols) provide fairly good approximations for both latency and batch reliability. However, at high error probability, route failures cause more TCP timeouts, which in turn causes more fluctuations in the TCP window size. In these cases, the average window size is

[6]For example, if each node always transmits one packet per transmission interval, only the head of line packet will be transmitted, and the entire batch will be delivered in order.

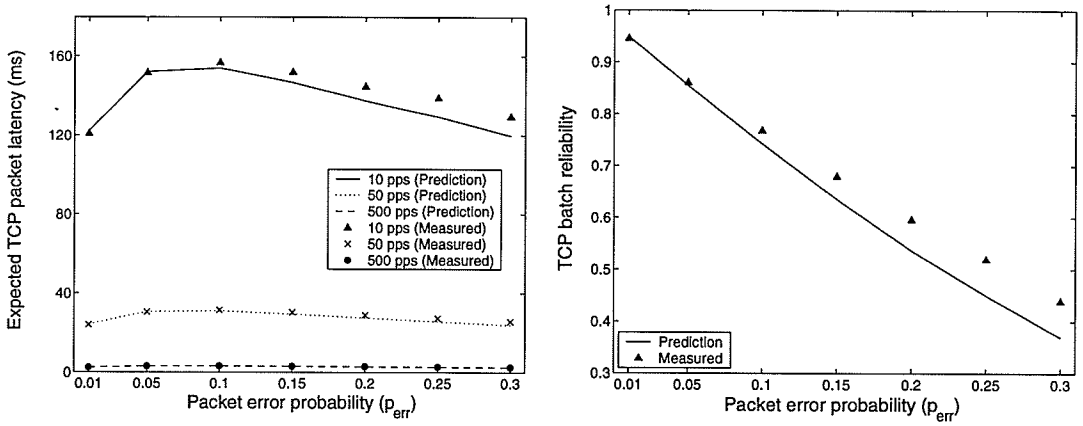


Figure 7.13. *Prediction of (a) expected TCP packet latency and (b) TCP batch reliability.*

not a valid representation for batch size. Therefore, predicted values deviate slightly from the simulation results. This discrepancy is scaled by hop-level delay, which is larger when the data rate is lower. Accordingly, the difference between the predicted and the measured values becomes smaller as data rate increases. To alleviate this discrepancy, we can predict the latency at every window adjustment instant. Since, at each instant, we have exact knowledge about the size of the congestion window, the predicted latency matches with those obtained from simulations in all the cases (the results are omitted for brevity).

In Figure 7.13, the concavity of packet latency in p_{err} results from two major factors. First, when operating over high p_{err}, TCP experiences frequent timeouts, and the corresponding average batch size (which is the denominator of packet latency) tends to be small. Secondly, an initial increase in p_{err} results in a smaller TCP window size, and therefore, an increase in packet latency. However, higher p_{err} also leads to reduction in latency due to more frequent route failures. The overall trend of packet latency depends on which factor is dominant for a particular p_{err}. In our particular case, increasing p_{err} initially increases packet latency since the denominator decreases. When p_{err} is relatively high, the average window size becomes somewhat constant (e.g., 1) and the packet latency decreases due to higher route failure probability. Again, when the data rate increases, the relative change with respect to hop-level

delay becomes less noticeable. Therefore, we observe rather monotonic change in packet latency for higher data rate (Figure 7.13).

7.5 Chapter Summary

We have presented an analytical model for batch transmission in a multi-hop and multi-rate wireless network. We have derived the *pmf* of the end-to-end latency as well as the number of packets successfully delivered to the destination. The following provides a summary of the key results obtained from the developed analytical model:

- The upper-bound and lower-bound for both expected end-to-end latency ($E[\mathscr{D}_{e2e}]$) and expected number of successfully delivered packets ($E[\mathscr{S}]$) can be obtained from ARQ$^{\infty}$ and ARQ^{0}, respectively. A certain level of trade-off between end-to-end latency and reliability can be achieved by adjusting route failure declaration probability (ξ) in ARQ^{P}.
- Route failure probability is an increasing function of ξ, packet error probability (p_{err}), and the number of packets in the network (N_2), and is a decreasing function of the number of channel states (M).
- $E[\mathscr{S}]$ is an increasing function of batch size (N_1), and is a decreasing function of the route failure probability, while $E[\mathscr{D}_{e2e}]$ is an increasing function of N_1. Low end-to-end latency can be achieved either when the route failure probability is very high where all the packets are dropped quickly, or when it is very small where all the packets are delivered within a short period of time.
- To maximize expected throughput ($E[\gamma_{e2e}] = E[\mathscr{S}]/E[\mathscr{D}_{e2e}]$), the optimal route failure declaration probability ξ^* can be obtained by solving $\xi^* = arg_{0 \leqslant \xi \leqslant 1} \max E[\gamma_{e2e}]$. The optimal ξ^* should neither be too small to cause rate under-utilization (as in ARQ$^{\infty}$) nor should it be too large to cause excessive packet drop (as in ARQ^{0}).
- For an end-to-end batch transmission, the probability that some packets remain in the network after $\lceil E[\mathscr{D}_{e2e}] \rceil$ intervals could be significantly high. To avoid the congestion, flow control mechanism should be based on the *cdf* rather than the expectation.
- Multi-rate transmission increases $E[\mathscr{S}]$ and $E[\gamma_{e2e}]$, decreases $E[\mathscr{D}_{e2e}]$, and helps avoid congestion.

- The *pmf* of the number of delivered packets is convex, and is strictly convex when the route failure probability is greater than zero but sufficiently small.

The developed model reveals an interesting inter-relationship between hop-level parameters and end-to-end performance metrics. As an example, we show that our model can predict TCP packet latency as well as batch reliability fairly accurately, especially under high data rate (i.e., link capacity) and low packet error probability in each hop.

Chapter 8

Summary and Directions for Future Research

8.1 Summary

A Markov-based model for a channel-quality-based opportunistic scheduling was proposed in Chapter 3 [7]. This model captures independent and correlated channel variation as well as rate adaptation at the link layer. It also incorporates the use of an ARQ mechanism to counteract possible transmission failures. Using this model, the *pmf* of the inter-access, inter-success, and connection-reset delay can be obtained. The main observation from this chapter is that the opportunistic scheduling aiming at maximizing system throughput could lead to severe unfairness among all mobiles.

Even under *Fixed Modulation and Coding (FMC)*, providing fairness among all mobiles is a complicated job. Most of fair scheduling algorithms in the literature resorted to different heuristic-based approaches. In Chapter 4, an optimization-based fair scheduling algorithm, *Optimal Radio Channel Allocation for Single-Rate Transmission (ORCA-SRT)*, was developed [86]. ORCA-SRT formulates a fair scheduling problem as an *assignment problem*, where each mobile and time slot are viewed as an individual and a job, respectively. ORCA-SRT solves the assignment problem, by using the *Hungarian Method*, and employs lead-lag compensation mechanism to minimize the transmission that might not be successful and to maintain fairness among all mobiles.

With multi-rate transmission, the fair scheduling problem becomes more complicated due to different fairness notions: *temporal fairness* and *throughput fairness*. An extension of ORCA-SRT, namely, *Optimal Radio Channel Allocation for Multi-*

Rate Transmission (ORCA-MRT) [41] was presented in Chapter 5. Subject to hard constraints on temporal fairness, ORCA-MRT employs cost-function manipulation to provide throughput fairness among mobiles. One of the main assumptions for both ORCA-SRT and ORCA-MRT is the perfect knowledge of channel states. In Chapter 5, two channel prediction models were also proposed. With these prediction models, ORCA-MRT performs as good as it does with perfect channel knowledge.

For a multi-hop wireless network, a mathematical model for deriving the required number of transmissions for a successful packet delivery was developed in Chapter 6 [87]. In this model, a packet is forwarded via several hops until it reaches the destination node. Each hop is subject to (independent) packet error probability. This probability could be due to channel fading or data collision in a contention-based MAC protocol. If the transmitted packet is in error, the transmitter will invoke retransmission procedures based on the underlying ARQ policy. If the packet is dropped due to limited-persistent ARQ, the source node will retransmit again after some time. The statistics for the required number of transmissions is an indication for both energy usage and end-to-end latency for a successful delivery of a single packet. The results from the above model revealed the performance degradation as the number of hops increases.

Multi-rate transmission improves network performance by using techniques like AMC. Nevertheless, it is still unclear how much improvement rate adaptation yields at the end-to-end level in a multi-hop wireless network. To answer this question, a Markov-based model was formulated for multi-rate transmission across a multi-hop link in Chapter 7, and the *pmf* of the required number of transmission intervals for successful transmission of a batch of packets (N packets) from a source node to a destination node was derived [88]. In this model, different types of hop-level ARQ policies are considered to combat with the error-prone wireless channel. If the packets are dropped due to limited persistence of the ARQ policy, the transmitting node will initiate a route discovery process. This situation can occur due to the absence of the receiving node. Therefore, all N transmitted packets may finally reach the destination, or be dropped during a route discovery process. End-to-end latency in this case is defined as the time until all N packets are no long in the network path. The trade-off between the reliability and latency in such a transmission scenario was

investigated thoroughly. Also, the usefulness of the analytical model in predicting the performance of end-to-end flow control mechanism such as TCP was illustrated.

8.2 Main Contributions and Insightful Results

For a cellular wireless network, an analytical framework was developed for a channel-quality-based opportunistic scheduling. In general, the scheduling algorithm operates better in less correlated wireless channels, due to higher time-diversity gain. We derive the *pmf* of the delay and throughput, and show that the *pmf* could provide QoS levels (e.g., the percentage of data delivery for delay-sensitive services) to the customers. This delivery percentage represents reliability-latency trade-off in that it decreases as the bounded delay becomes smaller. Although the use of multi-rate transmission (e.g., AMC) increases per-mobile throughput (and henece system throughput), it results in larger delay variation and unfairness among mobiles, since the only objective of the above scheduling algorithm is to maximize the throughput. To address the unfairness problem, we propose two optimization-based fair scheduling algorithms (ORCA-SRT and ORCA-MRT), and two channel prediction models. These algorithms outperform all other protocols proposed in the literature. In comparison to the channel-quality-based scheduling, these two protocols achieve better fairness at the expense of degrading throughput.

Two analytical models are developed for multi-hop wireless networks: one considering single-packet transmission and another for batch transmission. The developed models reveal interesting trade-off among reliability, latency, and energy, and would be useful in estimating end-to-end latency as well as energy consumption, and in the design of an efficient end-to-end flow control mechanism for multi-hop wireless networks.

The above trade-off can be manipulated by adjusting ARQ parameters. Under limited-persistent ARQ, packet dropping, especially at the hops closer to the destination node, is detrimental to both end-to-end latency and throughput. One solution to this problem is to increases persistence level of the underlying ARQ protocol. Nevertheless, this solution may lead to increasing latency for route failure detection[1]. In

[1]From this point of view, infinite-transmission ARQ is unable to detect route failure.

addition, when the channel condition is fairly poor, it might be better to reset the connection and find a new route with better channel quality.

In some cases, ARQ with high level of persistence also leads to throughput degradation. Consider a batch transmission under a multi-hop path. At the beginning, the transmitter uses all radio resource to transmit packets in a batch. As some packets are successfully transmitted, the transmitter might still try to transmit other few packets in its buffer, and might not be able to utilize all available radio resource. When a transmitter has less packets than it can transmit, the radio resource is under-utilized. In this case, the mobile should decrease transmission rate to decrease packet error probability. Since infinite-retransmission ARQ does not give up transmission until all packets in the buffer are successfully transmitted, resource under-utilization occurs for an extended period of time. The throughput in this case is, therefore, less than that with other ARQs with lower level of persistence. These ARQ protocols (e.g., ARQ^P or ARQ^F) give up retransmission and let the source node retransmit the dropped packets as well as newly arrived packets as a new batch. Therefore, they do not suffer much from wireless channel under-utilization.

Multi-rate transmission helps decreases route failure probability. Since a route failure can occur only if all transmitted packets are lost, a mobile transmitting more packets has less route failure probability. In some cases, packet drop under a multi-rate multi-hop wireless path with a small number of hops (e.g., two hops) could be less detrimental than the resource under-utilization. Under such circumstances, ARQ protocols with smaller persistence levels are preferable, since they yield higher throughput and less latency in route failure detection. The model proposed in this dissertation provides a means to quantify the trade-off between packet dropping and resource under-utilization. Therefore, the best ARQ policy for a certain scenario can be identified.

8.3 Work in Progress and Future Research Directions

The results obtained for cellular and multi-hop wireless networks would be useful for design and analysis of *multi-hop cellular networks*. However, the implementation of

this type of networks requires more research in many more aspects. The followings provide introductory descriptions for potential future research.

- **Medium Access Control (MAC) protocol**: The model in Chapter 7 assumes orthogonal channels for different hops. In some implementations, the channel access policy, especially between a relay node and a mobile, can be distributed. The incorporation of IEEE 802.11 MAC into the model would make the model more flexible, and more useful in some cases.
- **End-to-end flow control**: Another possible extension of the model in Chapter 7 is to identify the operation after all N packets have left the network path. For example, a new batch of packets can be transmitted immediately. The source node may adjust the batch size and/or defer the transmission of a new batch based on an acknowledgement received from the destination node. With a more completed model, flow control mechanism could be optimized, by experimenting with different environment and control parameters.
- **Multi-hop scheduling algorithms**: The multi-hop model considered in this dissertation assumes only single mobile at a source node. In multi-user environment, both source and intermediate nodes needs to schedule packet transmission for different mobiles. This dissertation reveals the unfairness and resource under-utilization of the current version of scheduling algorithm. Therefore, the newly-developed scheduling algorithms need to take channel condition, buffer variation, and other mobiles' constraints (e.g., fairness, delay, or energy consumption) into account.
- **Cooperative diversity**: An emerging concept, namely *cooperative diversity*, which exploits the principle of *Multi-Input Multi-Output* (MIMO) communications has shown improved performance for a broad class of wireless networks. Nevertheless, one of the major drawbacks of a MIMO system is the space and power requirements to accommodate multiple antennae. These requirements are very critical for a mobile with small size. Cooperative diversity employs the relay nodes as *virtual antennae*, and remove the space and power requirements at the mobile node. Since these relay nodes are not closely located, their channel fading characteristics tend to be independent, leading to higher space-diversity gain. Also, in a multi-hop cellular network, relay nodes can

be fixed and equipped with power supply, therefore eliminating the energy-consumption constraint. However, current research in this area does not take ARQ mechanism, buffer management, and scheduling algorithms into account . The research work in this dissertation can be extended for cooperative diversity as well.

Appendix A

Derivations of (2.19) and (2.20)

For a single absorbing state, $\mathbf{R} = \boldsymbol{e} - \mathbf{Q}\boldsymbol{e}$. Applying this relation to (2.16),

$$\begin{aligned} F_t &= \alpha_0 + \sum_{i=1}^{t} \boldsymbol{\alpha}\mathbf{Q}^{i-1}\mathbf{R}, \quad t \geqslant 1 \\ &= \alpha_0 + \sum_{i=1}^{t} \boldsymbol{\alpha}\mathbf{Q}^{i-1} \cdot (\boldsymbol{e} - \mathbf{Q}\boldsymbol{e}), \quad t \geqslant 1 \\ &= \alpha_0 + \boldsymbol{\alpha}\boldsymbol{e} - \boldsymbol{\alpha}\mathbf{Q}^{t}\boldsymbol{e}, \quad t \geqslant 1 \\ &= \alpha_0 + 1 - \boldsymbol{\alpha}\mathbf{Q}^{t}\boldsymbol{e}, \quad t \geqslant 1, \end{aligned} \tag{A.1}$$

which proves (2.19). Similarly, from (2.18)

$$\begin{aligned} E[t] &= \boldsymbol{\alpha}(\mathbf{I} - \mathbf{Q})^{-1}(\mathbf{I} - \mathbf{Q})^{-1}(\boldsymbol{e} - \mathbf{Q}\boldsymbol{e}) \\ &= \boldsymbol{\alpha}(\mathbf{I} - \mathbf{Q})^{-1}(\mathbf{I} - \mathbf{Q})^{-1}(\mathbf{I} - \mathbf{Q})\boldsymbol{e} \\ &= \boldsymbol{\alpha}(\mathbf{I} - \mathbf{Q})^{-1}\boldsymbol{e}, \end{aligned} \tag{A.2}$$

which proves (2.20).

Appendix B

Proof of Propositions 3.1, derivation of (3.30), and the extension for non-i.i.d. cases

For *Case I*,

$$\begin{aligned}
p_{tx}^{(t)}(m) &= Pr\{(\text{the channel of mobile } i \text{ is in state } m) \text{ and} \\
&\qquad (\text{mobile } i \text{ is selected})\} \\
&= \pi_m \sum_{k=0}^{n-1} \binom{n-1}{k} \frac{(\pi_m)^k (F_{m-1})^{n-k-1}}{k+1} \\
&= \frac{1}{n} \sum_{k=0}^{n-1} \binom{n}{k+1} (\pi_m)^{k+1} (F_{m-1})^{n-(k+1)} \\
&= \frac{1}{n} \sum_{k=1}^{n} \binom{n}{k} (\pi_m)^k (F_{m-1})^{n-k} \\
&= \frac{1}{n} \left(\sum_{k=0}^{n} \binom{n}{k} (\pi_m)^k (F_{m-1})^{n-k} \right) - (F_{m-1})^n \\
&= \frac{1}{n} (\pi_m + F_{m-1})^n - (F_{m-1})^n \\
&= \frac{(F_m)^n - (F_{m-1})^n}{n}, \qquad \text{(B.1)}
\end{aligned}$$

which proves PROPOSITION 3.1.

For *Case III*, when mobile 1 acquires a channel access, the number of eligible mobiles with FSMC and RSC would be $mult_{\mathbf{m}}(m_1)$ and k, respectively. Similar to (B.1), the probability that k out of n_r mobiles will be eligible is $\binom{n_r}{k} (\pi_{m_1})^k (F_{m_1-1})^{n_r-k}$.

Eq. (3.30) is then obtained, by summing all possible values of k weighted by probability $1/(k + mult_{\mathbf{m}}(m_1))$.

By setting $n_r = n - 1$,

$$\begin{aligned} g(m) &= \sum_{k=0}^{n-1} \binom{n-1}{k} \frac{(\pi_m)^k (F_{c-1})^{n-k-1}}{k+1} \\ &= \frac{p_{tx}^{(t)}(m)}{\pi_m} = \frac{f_{\gamma_{sys}}(m)}{n\pi_m}. \end{aligned} \tag{B.2}$$

When the channels for different mobiles are not i.i.d., $p_{tx}^{(t)}(m)$ becomes,

$$p_{tx}^{(t)}(m) = f_{\mathcal{M}_i}^{(t)}(m) \sum_{k=0}^{n-1} \frac{1}{k+1} \sum_{l=1}^{\binom{n-1}{k}} \cdot \left(\prod_{\forall j \in \mathbf{e}_l(n-1,k)} f_{\mathcal{M}_j}^{(t)}(m) \right) \cdot \left(\prod_{\forall j \notin \mathbf{e}_l(n-1,k)} F_{\mathcal{M}_j}^{(t)}(m) \right), \tag{B.3}$$

where $\mathbf{e}_l(n,k), l \in \{1, 2, \cdots, \binom{n}{k}\}$ consists of $\binom{n}{k}$ sets of k eligible mobiles when the total number of mobiles is n.

Appendix C

Proof of Corollary 3.1

Inter-access delay ($\mathscr{D}_{acc}$) is conditioned on that the mobile is granted channel access at time slot 0. Therefore, its *pmf* can be calculated by summing $f_{\mathscr{M},\mathscr{D}_{acc}}(m,d)$ in (3.2) over all possible channel states, that is, $f_{\mathscr{D}_{acc}}(d) = \sum_{\forall m} f_{\mathscr{M},\mathscr{D}_{acc}}(m,d)$. Now, $E[\gamma_{sys}]$ and $E[\mathscr{D}_{acc}]$ can be obtained by averaging γ_{sys} and $\mathscr{D}_{acc}$ over probabilities $f_{\gamma_{sys}}(m)$ and $f_{\mathscr{D}_{acc}}(d)$, respectively. Mathematically, from (3.2),

$$\begin{aligned}
E[\gamma_{sys}] &= \sum_{m=1}^{M} m \cdot ((F_m)^n - (F_{m-1})^n) \\
&= M(F_M)^n + \sum_{m=1}^{M-1} m \cdot (F_m)^n - \sum_{i=0}^{M-1} (i+1) \cdot (F_i)^n \\
&= M(F_M)^n + \sum_{m=1}^{M-1} m \cdot (F_m)^n - \sum_{i=1}^{M-1} (i+1) \cdot (F_i)^n + (F_0)^n \\
&= M + \sum_{m=1}^{M-1} m \cdot (F_m)^n - \sum_{i=1}^{M-1} i \cdot (F_i)^n + \sum_{i=1}^{M-1} (F_i)^n \\
&= M - \sum_{m=1}^{M-1} (F_m)^n. \qquad \text{(C.1)}
\end{aligned}$$

Similarly, from (3.5),

$$\begin{aligned} E[\mathscr{D}_{acc}] &= \sum_{d=1}^{\infty} d \cdot f_{\mathscr{D}_{acc}}(d) \\ &= \frac{1}{n} \cdot \sum_{d=1}^{\infty} d \left(\frac{n-1}{n}\right)^{d-1} \\ &= \frac{1}{n} \cdot \left(1 - \frac{n-1}{n}\right)^{-2} \\ &= n. \end{aligned} \tag{C.2}$$

Appendix D

Generalization of Channel-to-Rate Mapping Function

Let $E[\gamma_{sys}(M)]$ be the system throughput of a channel-quality-based opportunistic scheduling defined in Chapter 3, when the number of channel states is M. Also, let a general channel-to-rate mapping $r(m)$ returns transmission rate when the channel state is m. Following Appendix C,

$$\begin{aligned}
E[\gamma_{sys}(M)] &= \sum_{m=1}^{M} r(m) \cdot ((F_m)^n - (F_{m-1})^n) \\
&= r(M)(F_M)^n + \sum_{m=1}^{M-1} r(m) \cdot (F_m)^n - \sum_{i=0}^{M-1} r(i+1) \cdot (F_i)^n \\
&= r(M)(F_M)^n + \sum_{m=1}^{M-1} [r(m) - r(m+1)] \cdot (F_m)^n - r(1)\,(F_0)^n \\
&= r(M) - \sum_{m=1}^{M-1} [r(m+1) - r(m)] \cdot (F_m)^n \\
&= r(M) - \sum_{m=1}^{M-1} d(m) \cdot (F_m)^n, \qquad \text{(D.1)}
\end{aligned}$$

where $d(m) = r(m+1) - r(m)$.

Proposition D.1 *Let $\Delta(M) = E[\gamma_{sys}(M+1)] - E[\gamma_{sys}(M)]$. For an equally likely channel with $d(m) \geqslant 0$, $\Delta(M)$ is a decreasing function in n (total number of mobiles).*

PROOF First, expand $\Delta(M)$ as follows:

$$\begin{aligned}
\Delta(M) &= E[\gamma_{sys}(M+1)] - E[\gamma_{sys}(M)] \\
&= r(M+1) - \sum_{m=1}^{M} d(m) \cdot (F_m)^n - r(M) + \sum_{m=1}^{M-1} d(m) \cdot (F_m)^n \\
&= d(M) - \sum_{m=1}^{M} \frac{d(m)}{(M+1)^n} + \sum_{m=1}^{M-1} \frac{d(m)}{M^n} \\
&= d(M) + \sum_{m=1}^{M-1} \left(\frac{d(m)}{M^n} - \frac{d(m)}{(M+1)^n} \right) - \frac{d(M)}{M+1} \\
&= d(M) + \sum_{m=1}^{M-1} d(m) \left(\frac{1}{M^n} - \frac{1}{(M+1)^n} \right) - \frac{d(M)}{M+1}. \qquad \text{(D.2)}
\end{aligned}$$

The only part that depend on n is $\frac{1}{M^n} - \frac{1}{(M+1)^n}$. For $M > 0$, it can be easily observed that $\frac{1}{M^n} - \frac{1}{(M+1)^n}$ is an increasing function in n. Since $d(m) \geqslant 0, \forall m$, (D.2) is an increasing function in $n > 0$. ■

Theorem D.1 *For an equally-likely channel, $E[\gamma_{sys}(M)]$ in D.1 is an incresing in M if $d(m) \geqslant 0, \forall m$.* □

PROOF: *Again, we need to prove that $\Delta(M)$ in (D.2) is greater than 0 for all values of M and n. From proposition D.1, the minimum of $\Delta(M)$ is at $n = 1$ (one mobile). Therefore, we only need to prove for $n = 1$ that $\Delta(M) \geqslant 0, \forall M \geqslant 1$. From (D.2) with $n = 1$,*

$$\begin{aligned}
\Delta(M) &= d(M) + \sum_{m=1}^{M-1} d(m) \left(\frac{1}{M} - \frac{1}{(M+1)} \right) - \frac{d(M)}{M+1} \\
&= d(M) \left(1 - \frac{1}{M+1} \right) + \sum_{m=1}^{M-1} d(m) \left(\frac{1}{M} - \frac{1}{M+1} \right). \qquad \text{(D.3)}
\end{aligned}$$

Again, since $1 - \frac{1}{M+1} \geqslant 0$ and $\frac{1}{M} - \frac{1}{M+1} \geqslant 0$, $\Delta(M) \geqslant 0$. Therefore, $E[\gamma_{sys}(M)]$ is an increasing function in M. ■

From Theorem D.1, we conclude that if

- $r(m)$ is an increasing function in m (therefore $d(m) \geqslant 0$), and
- the channel state is equally-likely,

increasing M will lead to increased system throughput.

Appendix E

Proof of Theorem 3.2

Lemma E.1 *Let $\mathbf{\Omega}$ be a nil-potent matrix, where $\Omega_{ij} = [\mathbf{\Omega}]_{ij}$, and*

$$\Omega_{i,j} = \begin{cases} a, & i = j, \\ b, & i = j-1, \\ 0, & otherwise. \end{cases} \tag{E.1}$$

Then, $\Omega_{ij}^{(t)} = [\mathbf{\Omega}^{(t)}]_{ij}$ and $\Gamma_{ij} = [\mathbf{\Gamma}]_{ij} = [(\mathbf{I}-\mathbf{\Omega})^{-1}]_{ij}$ can be calculated from (E.2) below

$$\begin{aligned} \Omega_{ij}^{(t)} &= \begin{cases} \binom{t}{j-i} a^{t-(j-i)} b^{j-i}, & t \geqslant j-i, \\ 0, & otherwise, \end{cases} \\ \Gamma_{ij} &= \begin{cases} \frac{1}{1-a}, & i = j, \\ \frac{1}{1-a}\left(\frac{b}{1-a}\right)^{(j-i)}, & i < j, \\ 0, & otherwise. \end{cases} \end{aligned} \tag{E.2}$$

PROOF: LEMMA *E.1 can be proven by induction. Since $\mathbf{\Omega}$ is nil-potent*

$$\Omega_{ij}^{(t)} = \begin{cases} a \cdot \Omega_{ij}^{(t-1)} + b \cdot \Omega_{i+1,j}^{(t-1)}, & i < j, \\ a \cdot \Omega_{ij}^{(t-1)}, & i = j, \\ 0, & otherwise. \end{cases} \tag{E.3}$$

STEP 1*: Obviously,* LEMMA E.1 *is true for $t = 2$.*

STEP 2*: Assume that* LEMMA E.1 *is true for $t > 0$, and prove that prove that* LEMMA E.1 *is true for $t+1$. From (E.3),*

$$\Omega_{ij}^{(t+1)} = \begin{cases} \binom{t+1}{j-i} a^{(k+1)-(j-i)} b^{j-i}, & i < j, \\ a^{k+1}, & i = j, \\ 0, & \textit{otherwise}. \end{cases} \tag{E.4}$$

We observe that (E.4) is equivalent to that obtained from (E.2), which proves the first part of Lemma E.1. Similarly, we can also prove the second part with relationship $\mathbf{\Gamma}(\mathbf{I} - \mathbf{\Omega}) = \mathbf{I}$. *The proof includes the cases for* $\Gamma_{i,j+1}$, $\Gamma_{i+1,j}$, *and* $\Gamma_{i+1,j+1}$ *for all values of* i *and* j, *which is omitted for brevity.* ■

Consider an absorbing DTMC with initial probability matrix $\boldsymbol{\alpha}$ and TPM $\mathbf{W}$ given in (3.7). From (2.16)-(2.18), the joint probability that the process will be absorbed to state s at time d, absorbing probability to state s, and the expected time to absorption to state s can be calculated from $[\boldsymbol{\alpha}\mathbf{\Omega}^{(d-1)}\boldsymbol{\omega}]_{1,s}$, $[\boldsymbol{\alpha}(\mathbf{I}-\mathbf{\Omega})^{-1}\boldsymbol{\omega}]_{1,s}$, and $[\boldsymbol{\alpha}(\mathbf{I}-\mathbf{\Omega})^{-2}\boldsymbol{\omega}]_{1,s}$, respectively. For the DTMC in THEOREM 3.2, $a = \frac{n-1}{n}$ and $b = q_0$. By applying LEMMA E.1 to $\mathbf{\Omega}^{(d-1)}$, $(\mathbf{I}-\mathbf{\Omega})^{-1}$, and $(\mathbf{I}-\mathbf{\Omega})^{-2}$, THEOREM 3.2 can be proven by replacing a and b with $\frac{n-1}{n}$ and q_0, respectively.

Appendix F

The Hungarian Method to Solve an Assignment Problem

We illustrate the Hungarian method by using the following cost matrix **C**:

$$\mathbf{C} = \begin{pmatrix} 5 & 2 & 3 & 4 \\ 7 & 8 & 4 & 5 \\ 6 & 3 & 5 & 6 \\ 2 & 2 & 3 & 5 \end{pmatrix}.$$

The solution steps are described below. The details can be found in [58].

- **Step1:** Find the minimum element min_i in each row and subtract each element in the row by this value (i.e., $c_{ij} = c_{ij} - min_i, \forall_j$).
- **Step2:** Find the minimum element min_j in each column and subtract each element in the column by this value (i.e., $c_{ij} = c_{ij} - min_j, \forall_i$).

After step 2, the matrix **C** will be as follows:

$$\mathbf{C} = \begin{pmatrix} 3 & 0 & 1 & 1 \\ 3 & 4 & 0 & 0 \\ 3 & 0 & 2 & 2 \\ 0 & 0 & 1 & 2 \end{pmatrix}.$$

- **Step 3:** Use minimum line drawn[1] through all zeros in the matrix **C**. If the number of lines is equal to the matrix dimension (i.e., 4 in this case), go to step

[1] A line drawn through a row is represented by a line under the row. A line drawn through a column is represented by a line on the right of the column. In this example, lines are drawn through row 2, column 1, and column 2.

5. In this example, three lines is needed, and therefore, we proceed to to step 4.

$$\mathbf{C} = \left(\begin{array}{c|c|cc} 3 & 0 & 1 & 1 \\ 3 & 4 & 0 & 0 \\ \hline 3 & 0 & 2 & 2 \\ 0 & 0 & 1 & 2 \end{array}\right).$$

- **Step 4:** Find the minimum min_{ij} among the elements that are not drawn through by any line (in this example, min_{ij} is 1). Subtract min_{ij} from each element which does not has any line drawn through and add min_{ij} to each element which has two lines drawn through. Go back to step 3.

$$\mathbf{C} = \begin{pmatrix} 3 & 0 & 0 & 0 \\ 4 & 5 & 0 & 0 \\ 3 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

Step 3:

$$\mathbf{C} = \left(\begin{array}{cc|c|c} 3 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 \\ 3 & 0 & 1 & 1 \\ \hline 0 & 0 & 0 & 1 \end{array}\right).$$

Now, the number of lines needed to cover all zeros is four, which is equal to the dimension of the matrix, and therefore, we proceed to step 5.

- **Step 5:** Now the solution can be identified on the lines as a set of zeros (marked by squares) which contains only one marked zero in each row and column.

$$\mathbf{C} = \begin{pmatrix} 3 & 0 & 0 & \boxed{0} \\ 4 & 5 & \boxed{0} & 0 \\ 3 & \boxed{0} & 1 & 1 \\ \boxed{0} & 0 & 0 & 1 \end{pmatrix}.$$

$$x_{ij} = \begin{cases} 1, & (i,j) = \{(4,1),(3,2),(2,3),(1,4)\}, \\ 0, & otherwise. \end{cases}$$

Therefore, the minimum cost will be achieved if we assign the 1^{st} job to 4^{th} individual, 2^{nd} job to 3^{rd} individual, 3^{rd} job to 2^{nd} individual, and 4^{th} job to 1^{st} individual.

Appendix G

Proof of Theorem 5.1

Let $\Delta(t|i)$ be average prediction error when the last know state is i, and the probabilities that predicted state ($\mathscr{M}'$) and actual channel state ($\mathscr{M}$) are in states m' and m in time slot t be represented by $f^{(t)}_{\mathscr{M}'}(m'|i) = p^{(t)}_{i,m'}$ and $f^{(t)}_{\mathscr{M}}(m|i) = p^{(t)}_{i,m}$, respectively. Since predicted and actual channel are identical and independent to each other,

$$\Delta(t) = \sum_{i=1}^{M} \pi_i \cdot \Delta(t|i), \tag{G.1}$$

$$\text{where} \quad \Delta(t|i) = \sum_{m'=1}^{M} \sum_{m=1}^{M} |m - m'| \cdot p^{(t)}_{i,m} \cdot p^{(t)}_{i,m'}. \tag{G.2}$$

G.1 Proof of eq. (5.24)

From (5.11),

$$\Delta_E(t) = \sum_{i=1}^{M} \pi_i \sum_{m=1}^{M} |m - E[\mathscr{M}_t|i]| \cdot p^{(t)}_{i,m} \cdot 1, \tag{G.3}$$

which proves (5.24).

G.2 Proof of eq. (5.25)

Let an autocorrelation function of $p^{(t)}_{i,m}$ ($R_\tau(t|i)$) be the probability that the states in time slot t of two identical FSMC processes starting at state i will be differed by τ and be calculated from (G.4) below. The expectation of $R_\tau(t|i)$ over all values of τ ($E_\tau[R_\tau(t|i)]$) can be calculated from (G.5) below,

$$
\begin{aligned}
R_\tau(t|i) &= \sum_{\forall m: m+\tau \leqslant M} p_{i,m}^{(t)} \cdot p_{i,m+\tau}^{(t)} + \sum_{\forall m: m-\tau \geqslant 1} p_{i,m}^{(t)} \cdot p_{i,m-\tau}^{(t)} \\
&= \sum_{m=1}^{M-\tau} 2 \cdot p_{i,m}^{(t)} \cdot p_{i,m+\tau}^{(t)}, \qquad \text{(G.4)} \\
E_\tau[R_\tau(t|i)] &= \sum_{\forall \tau} \tau \cdot R_\tau(t|i) \\
&= \sum_{\tau=1}^{M-1} \sum_{m=1}^{M-\tau} 2\tau \cdot p_{i,m}^{(t)} \cdot p_{i,m+\tau}^{(t)}. \qquad \text{(G.5)}
\end{aligned}
$$

Equivelent to $\Delta(t|i)$, $E_\tau[R_\tau(t|i)]$ is an average prediction error in time slot t when the initial state is i. To verify this, we replace $m' = m + \tau$ in (G.2) and obtain $\Delta(t|i) = E_\tau[R_\tau(t|i)]$. Therefore,

$$
\begin{aligned}
\Delta_S(t) &= \sum_{i=1}^{M} \pi_i \cdot E_\tau[R_\tau(t|i)] \\
&= \sum_{i=1}^{M} \pi_i \sum_{\tau=1}^{M-1} \sum_{m=1}^{M-\tau} 2\tau \cdot p_{i,m}^{(t)} \cdot p_{i,m+\tau}^{(t)}. \qquad \text{(G.6)}
\end{aligned}
$$

G.3 Proof of eqs. (5.26) and (5.27)

In an RSC, $p_{i,m}^{(t)} = \pi_m(\forall i, t)$ and the average channel state is $\sum_{m=1}^{M} m \cdot \pi_m$. By replacing the $p_{i,m}^{(t)}$ and the average channel state into (G.1) and (G.2), eq. (5.26) is proven. In common with the proof of (5.25), eq. (5.27) can be proven by replacing $p_{i,m}^{(t)}$ and $p_{i,m+\tau}^{(t)}$ with π_m and $\pi_{m+\tau}$, respectively.

Appendix H

Derivation of (6.25)

$$\begin{aligned}
\sum_{t=0}^{\infty} F_{\mathscr{T}}^{ARQ1}(t) - \sum_{t=0}^{\infty} F_{\mathscr{T}}^{ARQ2}(t) &= \sum_{t=1}^{\infty} \left\{1 - \boldsymbol{\alpha} \cdot (\mathbf{T}^{ARQ1})^t \cdot \boldsymbol{e}\right\} - \\
&\qquad \sum_{t=1}^{\infty} \left\{1 - \boldsymbol{\alpha} \cdot (\mathbf{T}^{ARQ2})^t \cdot \boldsymbol{e}\right\} \\
&= \boldsymbol{\alpha} \cdot \sum_{t=1}^{\infty} (\mathbf{T}^{ARQ2})^t \cdot \boldsymbol{e} - \boldsymbol{\alpha} \cdot \sum_{t=1}^{\infty} (\mathbf{T}^{ARQ1})^t \cdot \boldsymbol{e} \\
&= \boldsymbol{\alpha} \cdot ((\mathbf{I} - \mathbf{T}^{ARQ2})^{-1} - \mathbf{I}) \cdot \boldsymbol{e} - \\
&\qquad \boldsymbol{\alpha} \cdot ((\mathbf{I} - \mathbf{T}^{ARQ1})^{-1} - \mathbf{I}) \cdot \boldsymbol{e} \\
&= E^{ARQ2}[\mathscr{T}] - E^{ARQ1}[\mathscr{T}]. \qquad \text{(H.1)}
\end{aligned}$$

Appendix I

Illustrative Examples for Major Matrices in ARQ$^{\infty}$ and ARQ^{FP}

For ARQ$^{\infty}$, the TPM for the case with $N = 3$ in matrix and scalar forms are shown in (I.2) and (I.3), repectively. Also, with $K_1 = K_2 = 0$, the matrix and scalar forms for $\mathbf{\Omega}_3^{FP}$ and ω_3^{FP} are shown in (I.4) and (I.5), respectively. Representing each row/column, each three-letter label and two-letter label in the first row and column stand for $(X_1^{(t)}, X_2^{(t)}, X_3^{(t)})$ and $(i, X_1^{(t)})$, respectively, where i is the number of packets in the system $\mathcal{T}_i$. In a scalar form, each two-letter code (c_1, c_2) representing each entry of $\mathbf{P}_3$ and $\mathbf{\Omega}_3^{FP}$ corresponds to probability $P_1(c_1) \cdot P_2(c_2)$, where

$$P_n(c_n) = \begin{cases} 1, & c_n = X, \\ p_n(0) \cdot \xi_n, & c_n = F, \\ p_n(0) \cdot (1 - \xi_n), & c_n = 0, \\ p_n(c_n), & otherwise, \end{cases} \tag{I.1}$$

and $p_n(c_n)$ is calculated from (7.1).

$$\mathbf{P}_3 = \left(\begin{array}{c|c|ccc|cccc|ccc|c}
 & 003 & 012 & 021 & 030 & 102 & 111 & 120 & 201 & 210 & 300 \\
\hline
003 & 1 & & & & & & & & & \\
\hline
012 & & \multicolumn{3}{c|}{\multirow{3}{*}{$\mathbf{A}'_3$}} & & & & & & \\
021 & & & & & & & & & & \\
030 & & & & & & & & & & \\
\hline
102 & & \multicolumn{3}{c|}{\multirow{3}{*}{$p_1(1)\cdot\mathbf{A}_3$}} & \multicolumn{4}{c|}{\multirow{3}{*}{$p_1(0)\cdot\mathbf{A}_3$}} & & \\
111 & \mathbf{R}_3 & & & & & & & & & \\
120 & & & & & & & & & & \\
\hline
201 & & \mathbf{0} & \multicolumn{2}{c|}{\multirow{2}{*}{$p_1(2)\cdot\mathbf{A}_2$}} & \mathbf{0} & \multicolumn{2}{c|}{\multirow{2}{*}{$p_1(1)\cdot\mathbf{A}_2$}} & \multicolumn{2}{c|}{\multirow{2}{*}{$p_1(0)\cdot\mathbf{A}_2$}} & \\
210 & & & & & & & & & & \\
\hline
300 & & \multicolumn{2}{c|}{\mathbf{0}} & $p_1(3)\cdot\mathbf{A}_1$ & \multicolumn{2}{c|}{\mathbf{0}} & $p_1(2)\cdot\mathbf{A}_1$ & \mathbf{0} & $p_1(1)\cdot\mathbf{A}_1$ & $p_1(0)\cdot\mathbf{A}_1$
\end{array}\right). \tag{I.2}$$

$$\mathbf{P}_3 = \left(\begin{array}{c|c|ccc|ccc|cc|c}
 & 003 & 012 & 021 & 030 & 102 & 111 & 120 & 201 & 210 & 300 \\
\hline
003 & 1 & & & & & & & & & \\
\hline
012 & X,1 & X,0 & & & & & & & & \\
021 & X,2 & X,1 & X,0 & & & & & & & \\
030 & X,3 & X,2 & X,1 & X,0 & & & & & & \\
\hline
102 & & 1,X & & & 0,X & & & & & \\
111 & & 1,1 & 1,0 & & 0,1 & 0,0 & & & & \\
120 & & 1,2 & 1,1 & 1,0 & 0,2 & 0,1 & 0,0 & & & \\
\hline
201 & & & 2,X & & & 1,X & & 0,X & & \\
210 & & & 2,1 & 2,0 & & 1,1 & 1,0 & 0,1 & 0,0 & \\
\hline
300 & & & & 3,X & & & 2,X & & 1,X & 0,X
\end{array}\right). \tag{I.3}$$

$$\Omega_3^{FP} =$$

	10 11	20 21 22	30 31 32 33
10	$\mathbf{Q}_1$		
11			
20	$\mathbf{V}_{21}$	$\mathbf{Q}_2$	
21			
22			
30	$\mathbf{V}_{31}$	$\mathbf{V}_{32}$	$\mathbf{Q}_3$
31			
32			
33			

$$=$$

	010 100	011 020 101 110 200	012 021 030 102 111 120 201 210 300
010	$\mathbf{Q}_1$		
100			
011			
020			
101		$\mathbf{Q}_2$	
110	$F,0$		
200			
012			
021			
030			
102			
111		$F,0$	$\mathbf{Q}_3$
120		$F,1$ $F,0$	
201			
210	$F,0$		
300			

(I.4)

$$\omega_3^{FP} = \left(\begin{array}{c|cc|c|c} & 000 & 001 & 002 & 003 \\ \hline 10 & \mathbf{u}_1' & \multirow{2}{*}{$\mathbf{R}_1$} & \multicolumn{2}{c}{\multirow{2}{*}{$\mathbf{0}$}} \\ 11 & \mathbf{u}_1 & & & \\ \hline 20 & \multicolumn{2}{c|}{\mathbf{u}_2'} & \multirow{2}{*}{$\mathbf{R}_2$} & \multirow{2}{*}{$\mathbf{0}$} \\ 21 & \multicolumn{2}{c|}{\mathbf{u}_2} & & \\ \hline 22 & \mathbf{u}_1 & $\mathbf{0}$ & & \\ \hline 30 & \multicolumn{2}{c|}{\mathbf{u}_3'} & & \multirow{3}{*}{$\mathbf{R}_3$} \\ 31 & \multicolumn{2}{c|}{\mathbf{u}_3} & & \\ \hline 32 & \multicolumn{2}{c|}{\mathbf{u}_2} & $\mathbf{0}$ & \\ \hline 33 & \mathbf{u}_1 & \multicolumn{2}{c|}{$\mathbf{0}$} & \end{array}\right) = \left(\begin{array}{c|c|c|c|c} & 000 & 001 & 002 & 003 \\ \hline 010 & X,F & \multirow{2}{*}{$\mathbf{R}_1$} & \multicolumn{2}{c}{\multirow{2}{*}{$\mathbf{0}$}} \\ 100 & F,X & & & \\ \hline 011 & & X,F & \multirow{3}{*}{$\mathbf{R}_2$} & \multirow{3}{*}{$\mathbf{0}$} \\ 020 & X,F & & & \\ \hline 101 & & F,X & & \\ 110 & X,F & F,1 & & \\ \hline 200 & F,X & & & \\ \hline 012 & & & X,F & \multirow{6}{*}{$\mathbf{R}_3$} \\ 021 & & X,F & & \\ 030 & X,F & & & \\ \hline 102 & & & F,X & \\ 111 & & X,F & F,1 & \\ 120 & X,F & & F,2 & \\ \hline 201 & & F,X & & \\ 210 & X,F & F,1 & & \\ \hline 300 & F,X & & & \end{array}\right). \quad \text{(I.5)}$$

Appendix J

Size of Matrices in ARQ^{FP}

$\mathbf{Q}_k^{FP}$ consists of $k+1$ sub-blocks, each representing a fixed value of $X_1^{(t)}$. Only the first sub-block with size k corresponds to $X_1^{(t)} = 0$. In each subsequent sub-block $j = \{2, \cdots, k+1\}$ whose size is $k-j+2$, only the first column corresponds to $X_2^{(t)} = 0$. In order to track the transmission counter of node n with $X_n^{(t)} \neq 0$, we expand the matrix by K_n+1 times. Accordingly, the size of the first sub-block, where $X_1^{(t)} = 0$, is $|\mathbf{S}_1| \cdot (K_2+1)$, where $|\mathbf{S}_i|$ is the size of sub-block i in case of $K_1 = K_2 = 0$. Similarly, the size of sub-block $j = \{2, \cdots, k+1\}$ is $(K_1+1)+(K_1+1)(K_2+1)(|\mathbf{S}_j|-1)$. Therefore, the size of $\mathbf{Q}_k^{FP}$ is

$$
\begin{aligned}
L_k &= |\mathbf{S}_1| \cdot (K_2+1) + \sum_{j=2}^{k+1} [(K_1+1) + (K_1+1)(K_2+1)(|\mathbf{S}_j|-1)] \\
&= k \cdot (K_2+1) + \sum_{j=2}^{k+1} (K_1+1) + (K_1+1)(K_2+1) \sum_{j=2}^{k+1} (|\mathbf{S}_j|-1) \\
&= k \cdot (K_2+1) + k(K_1+1) + (K_1+1)(K_2+1) \sum_{j=2}^{k+1} (k-j+2-1) \\
&= k \cdot (K_1+K_2+2) + (K_1+1)(K_2+1) \left[\sum_{j=2}^{k+1} (k+1) - \sum_{j=2}^{k+1} j \right] \\
&= k \cdot (K_1+K_2+2) + (K_1+1)(K_2+1) \left[k(k+1) - \frac{k(k+1)}{2} - k \right] \\
&= k \cdot (K_1+K_2+2) + (K_1+1)(K_2+1) \frac{k(k+1)}{2}. \qquad \text{(J.1)}
\end{aligned}
$$

Since $\mathbf{\Omega}_N^{FP}$ consists of $\mathbf{Q}_k^{FP}$, the size of $\mathbf{\Omega}_N^{FP}$ can be calculated from (J.2), where

$A = K_1 + K_2 + 2$ and $B = (K_1 + 1)(K_2 + 1)$.

$$
\begin{aligned}
L_N^{FP} &= \sum_{k=1}^{N} L_k \\
&= \sum_{k=1}^{N} \left(Ak + B \cdot \frac{k(k-1)}{2} \right) \\
&= A \sum_{k=1}^{N} k + \frac{B}{2} \left(\sum_{k=1}^{N} k^2 - \sum_{k=1}^{N} k \right) \\
&= \left(A - \frac{B}{2} \right) \left(\sum_{k=1}^{N} k \right) + \frac{B}{2} \left(\sum_{k=1}^{N} k^2 \right) \\
&= \left(A - \frac{B}{2} \right) \frac{N(N+1)}{2} + \frac{B}{2} \cdot \frac{N(N+1)}{2} \cdot \frac{2N+1}{3} \\
&= \frac{N(N+1)}{2} \left(A - \frac{B}{2} + \frac{B}{2} \cdot \frac{2N+1}{3} \right) \\
&= \frac{N(N+1)}{12} \left(6A - 3B + B(2N+1) \right) \\
&= \frac{N(N+1)}{12} \left(6A - B(2N-2) \right) \\
&= \frac{N(N+1)}{6} \left(3A - B(N-1) \right) \\
&= \frac{N(N+1)}{6} \left(3(K_1 + K_2 + 2) - (K_1 + 1)(K_2 + 1)(N-1) \right). \qquad \text{(J.2)}
\end{aligned}
$$

Bibliography

[1] H. Holma and A. Toskala, *WCDMA for UMTS: Radio Access for Third Generation Mobile Communications*, 3rd ed. Wiley-VCH, 2005.

[2] V. K. Varma, *et al.*, "Guest editorial - integration of 3G wireless and wireless LANs," *IEEE Communications Magazine*, vol. 41, no. 11, pp. 72–73, Nov. 2003.

[3] M. M. Buddhikot, G. Chandranmenon, S. Han, Y. Lee, S. Miller, and L. Salgarelli, "Design and implementation of a WLAN/CDMA2000 interworking architecture," *IEEE Communications Magazine*, vol. 41, no. 11, pp. 90–95, Nov. 2003.

[4] S. Y. Hui and K. H. Yeung, "Challenges in the migration to 4G mobile systems," *IEEE Communications Magazine*, vol. 41, no. 12, pp. 54–59, Dec. 2003.

[5] P. Bender, *et al.*, "CDMA/HDR: A bandwidth efficient high speed wireless data service for nomadic users," *IEEE Communications Magazine*, vol. 38, no. 7, pp. 70–77, Jul. 2000.

[6] D. N. C. Tse, "Optimal power allocation over parallel Gaussian broadcast channels," in *Proceedings of IEEE International Symposium on Information Theory*, 1997.

[7] T. Issariyakul and E. Hossain, "Channel-quality-based opportunistic scheduling with ARQ in multi-rate wireless networks: Modeling and analysis," to appear in *IEEE Transactions on Wireless Communications*, 2005.

[8] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: The single-node case," *IEEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 344–357, Jun. 1993.

[9] S. Lu, V. Bharghavan, and R. Skikant, "Fair scheduling in wireless packet networks," *IEEE/ACM Transactions on Networking*, vol. 7, no. 4, pp. 473–489, Aug. 1999.

[10] T. S. E. Ng, I. Stoica, and H. Zhang, "Packet fair queuing for wireless networks with location-dependent errors," in *Proceedings of the* 17^{th} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'98)*, Mar. 1998.

[11] S. Lu, T. Nandagopal, and V. Bharghavan, "A wireless fair service algorithm for

packet cellular networks," in *Proceedings of the 4th ACM Annual International Conference on Mobile Computing and Networking (MobiCom'98)*, Oct. 1998.

[12] T. Nandagopal, S. Lu, and V. Bhargharvan, "A unified architecture for the design and evaluation of wireless fair queuing algrithms," in *Proceedings of the 5th ACM Annual International Conference on Mobile Computing and Networking (MobiCom'99)*, Aug. 1999, pp. 132–142.

[13] T. Issariyakul, E. Hossain, and D. I. Kim, "A survey of medium access control in wireless ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, Dec. 2003.

[14] R. Bruno, M. Conti, and E. Gregori, "Mesh networks: Commodity multihop ad hoc networks," *IEEE Communications Magazine*, vol. 43, no. 3, pp. 123–131, Mar. 2005.

[15] I. F. Akyildiz, *et al.*, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, Mar. 2002.

[16] H.-Y. Wei and R. D. Gitlin, "Two-hop-relay architecture for next-generation WWAN/WLAN integration," *IEEE Wireless Communications*, vol. 11, no. 2, pp. 24–30, Apr. 2004.

[17] R. Pabst, *et al.*, "Relay-based deployment concepts for wireless and mobile broadband radio," *IEEE Communications Magazine*, vol. 42, no. 9, pp. 80–89, Sept. 2004.

[18] R. Nelson, *Probability, Stochastic Processes, and Queuing Theory: The Mathematical of Computer Performance Modeling.* New York: Springer-Verlag, 2000.

[19] K. S. Trivedi, *Probability and Statistics with Reliability, Queuing and Computer Science Applications.* New York: Wiley & Sons, Inc., 2002.

[20] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models.* The John Hopkins University Press, 1981.

[21] Q. Liu, S. Zhou, and G. B. Giannakis, "Cross-layer combining of adaptive modulation and coding with truncated ARQ over wireless links," *IEEE Transactions on Wireless Communications*, vol. 3, no. 5, pp. 1746–1755, Sept. 2004.

[22] M. Hassan, M. M. Krunz, and I. Matta, "Markov-based channel characterization for tractable performance analysis in wireless packet networks," *IEEE Transactions on Wireless Communications*, vol. 3, no. 3, pp. 821–831, May 2004.

[23] Q. Liu, S. Zhou, and G. B. Giannakis, "Queuing with adaptive modulation and coding over wireless links: Cross-layer analysis and design," *IEEE Transactions on Wireless Communications*, vol. 4, no. 3, pp. 1142–1153, May 2005.

[24] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Sys Tech. Journal*, vol. 39, no. 9, pp. 1253–1265, Sept. 1960.

[25] E. O. Elliott, "Estimates of error rates for codes on burst-noise," *Bell Sys Tech. Journal*, vol. 42, no. 9, pp. 1977–1997, Sept. 1963.

[26] A. Chockalingam, *et al.*, "Performance of a wireless access protocol on correlated Rayleigh-fading channel with capture," *IEEE Transactions on Communications*, vol. 46, no. 5, pp. 644–655, May 1998.

[27] *Physical layer aspects of UTRA high speed downlink packet access (release 4)*, 3GPP TR 25.848 V4.0.0 Std., 2001.

[28] *Physical layer standard for CDMA2000 spread spectrum systems*, 3GPP2 C.S0002-0 V1.0 Std., Jul. 1999.

[29] *IEEE standard for local and metropolitan area networks part 16: Air interface for fixed broadband wireless access systems*, IEEE standard 802.16 Working Group Std., 2002.

[30] *IEEE 802.11 - Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, IEEE inc. Std., 1999.

[31] A. Doufexi, *et al.*, "A comparison of the HIPERLAN/2 and IEEE 802.11a wireless LAN standards," *IEEE Communications Magazine*, vol. 40, no. 5, pp. 172–180, May 2002.

[32] Q. Zhang and S. A. Kassam, "Finite-state Markov model for Rayleigh fading channels," *IEEE Transactions on Communications*, vol. 47, no. 11, pp. 1688–1692, Nov. 1999.

[33] X. Qiu and K. Chawla, "On the performance of adaptive modulation in cellular systems," *IEEE Transactions on Communications*, vol. 47, no. 6, pp. 884–895, Jun. 1999.

[34] Y. L. Guan and L. F. Turner, "Generalized FSMC model for radio channel with correlated fading," *IEE Proceedings Communications*, vol. 146, pp. 133–137, Apr. 1999.

[35] R. Knopp and P. A. Humbelt, "Information capacity and power control in single-cell multiuser communications," in *Proceedings of IEEE International Conference on Communications (ICC'95)*, 1995, pp. 331–335.

[36] "White paper: Technical overview of 1xEV-DV version G1.4," Motorola Inc., 2002.

[37] X. Liu, E. K. P. Chong, and N. B. Shroff, "Transmission scheduling for efficient wireless utilization," in *Proceedings of the* 20^{th} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'01)*, Apr. 2001, pp. 776–785.

[38] A. Jalali, R. Padovani, and R. Pankaj, "Data throughput of CDMA-HDR a high

efficiency-high data rate personal communication wireless system," in *Proceedings of IEEE Vehicular Technology Conference (VTC'00) Spring*, May 2000.

[39] S. Shakkottai and A. L. Stolyar, "Scheduling for multiple flows sharing a time-varying channel: The exponential rule," *Analytic Methods in Applied Probability*, vol. 207, no. 2, pp. 185–202, 2002.

[40] M. Andrews, *et al.*, "Providing quality of service over a shared wireless link," *IEEE Communications Magazine*, vol. 39, no. 2, pp. 150–154, Feb. 2001.

[41] T. Issariyakul and E. Hossain, "ORCA-MRT: An optimization-based approach for fair scheduling in multi-rate TDMA wireless networks," to appear in *IEEE Transactions on Wireless Communications*, 2005.

[42] Y. Liu and E. Knightly, "Opportunistic fair scheduling over multiple wireless channels," in *Proceedings of the* 22^{nd} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'03)*, Mar. 2003.

[43] J.-W. Lee, R. R. Mazumdar, and N. B. Shroff, "Opportunistic power scheduling for multi-server wireless systems with minimum performance constraints," in *Proceedings of the* 23^{rd} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, Mar. 2004.

[44] M. Andrews and L. Zhang, "Scheduling over non-stationary wireless channels with finite rate sets," in *Proceedings of the* 23^{rd} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, Mar. 2004.

[45] M. Zorzi, "Data-link packet dropping models for wireless local communications," *IEEE Transactions on Vehicular Technology*, vol. 51, no. 4, pp. 710–719, July 2002.

[46] M. S. Alouini and A. J. Goldsmith, "Adaptive modulation over Nakagami fading channels," *Wireless Personal Communications*, vol. 13, no. 1-2, pp. 119–143, May 2000.

[47] H. Fattah and C. Leung, "An overview of scheduling algorithms in wireless multimedia networks," *IEEE Wireless Communications*, vol. 9, no. 5, pp. 76–83, Oct. 2002.

[48] P. Bhagwat, P. Bhattacharya, A. Krishna, and S. Tripathi, "Enhancing throughput over wireless LANs using channel state dependent packet scheduling," in *Proceedings of the* 15^{rd} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'96)*, Mar. 1996.

[49] S. Lu, V. Bharghavan, and R. Srikant, "Fair scheduling in wireless packet networks," in *Proceedings of the ACM Annual Conference of the Special Interest Group on Data Communication (SIGCOMM'97)*, Aug. 1997.

[50] P. Ramanathan and P. Agrawal, "Adapting packet fair queuing algorithms to

wireless networks," in *Proceedings of the 4th ACM Annual International Conference on Mobile Computing and Networking (MobiCom'98)*, Oct. 1998.

[51] C. Fragouli, V. Srivaraman, and M. B. Srivastava, "Controlled multimedia wireless link sharing via enhanced class-based queuing with channel-state-dependent packet scheduling," in *Proceedings of the 17th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'98)*, Mar. 1998.

[52] A. Demers, S. Keshev, and S. Shenker, "Analysis and simulation fo a fair queuing algorithm," in *Proceedings of ACM the ACM Annual Conference of the Special Interest Group on Data Communication (SIGCOMM'89)*, Aug. 1989.

[53] P. Goyal, H. M. Vin, and H. Chen, "Start-time fair queuing: A scheduling algorithm for integrated service access," in *Proceedings of the ACM Annual Conference of the Special Interest Group on Data Communication (SIGCOMM'96)*, Aug. 1996.

[54] H. W. Khun, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, no. 2, pp. 83–97, 1955.

[55] C. H. Papadimitriou and K.Steiglitz, *Combinatorial Optimization: Algorithms and Complexity.* New Jersey: Prentice-Hall, 1982.

[56] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, 1993.

[57] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly, "Opportunistic media access for multirate ad hoc networks," in *Proceedings of the 8th ACM Annual International Conference on Mobile Computing and Networking (MobiCom'02)*, Sept. 2002.

[58] L. Cooper and D. Steinberg, *Methods and Applications of Linear Programming.* Philadelphia: Saunders, 1974.

[59] H. Kushner and D. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems.* Springer-Verlag, 1978.

[60] H.-Y. Hsieh and R. Sivakumar, "On using peer-to-peer communication in cellular wireless data networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 1, pp. 57–72, Jan.-Mar. 2004.

[61] V. Gambiroza, B. Sadeghi, and E. W. Knightly, "End-to-end performance and fairness in multihop wireless backhaul networks," in *Proceedings of the 10th ACM Annual International Conference on Mobile Computing and Networking (MobiCom'04)*, Sept. 2004.

[62] S. Bansal, *et al.*, "Energy efficiency and throughput for TCP traffic in multi-hop wireless networks," in *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'02)*, June 2002, pp. 210–219.

[63] S. Xu and T. Saadawi, "Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?" *IEEE Communications Magazine*, vol. 39, no. 6, pp. 130–137, Jun. 2001.

[64] Z. Fu, *et al.*, "The impact of multihop wireless channel on TCP throughput and loss," in *Proceedings of the* 22^{nd} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'03)*, Mar. 2003.

[65] A. A. Kherani and R. Shorey, "Throughput analysis of TCP in multi-hop wireless networks with IEEE 802.11 MAC," in *Proceedings of IEEE Wireless Communications & Networking Conference (WCNC'04)*, Mar. 2004.

[66] S. Gobriel, R. Melhem, and D. Mosse, "A unified interference/collision analysis for power-aware adhoc networks," in *Proceedings of the* 23^{rd} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, Mar. 2004.

[67] S. Banerjee and A. Misra, "Minimum energy paths for reliable communication in multi-hop wireless networks," in *Proceedings of the* 3^{rd} *ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02)*, June 2002, pp. 146–156.

[68] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[69] L.-C. Wang, S.-Y. Huang, and A. Chen, "On the throughput performance of CSMA-based wireless local area network with directional antennas and capture effect: A cross-layer analytical approach," in *Proceedings of IEEE Wireless Communications & Networking Conference (WCNC'04)*, Mar. 2004.

[70] The Network Simulator - ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns/

[71] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network," in *Proceedings of the* 22^{nd} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'03)*, Mar. 2003.

[72] C. Ware, T. Wysocki, and J. Chicharo, "Hidden terminal jamming problems in IEEE 802.11 mobile ad hoc networks," in *Proceedings of IEEE International Conference on Communications (ICC'01)*, June 2001, pp. 261–265.

[73] A. K. Salkintzis, C. Fors, and R. Pazhyannur, "WLAN-GPRS integration for next-generation mobile data networks," *IEEE Wireless Communications*, vol. 9, no. 5, pp. 112–124, Oct. 2002.

[74] G. N. Aggelou and R. Tafazolli, "On the relaying capability of next-generation

GSM cellular networks," *IEEE Personal Communications*, vol. 8, no. 1, pp. 40–47, Feb. 2001.

[75] H. Luo, *et al.*, "UCAN: A unified cellular and ad-hoc network architecture," in *Proceedings of the* 9^{th} *ACM Annual International Conference on Mobile Computing and Networking (MobiCom'03)*, Sept. 2003, pp. 353–367.

[76] Y.-D. Lin and Y.-C. Hsu, "Multihop cellular: A new architecture for wireless communications," in *Proceedings of the* 19^{th} *Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'00)*, Mar. 2001, pp. 1273–1282.

[77] H. Wu, *et al.*, "Integrated cellular and ad hoc relaying systems: iCAR," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 10, pp. 2105–2115, Oct. 2001.

[78] P. Lin, W.-R. Lai, and C.-H. Gan, "Modeling opportunity driven multiple access in UMTS," *IEEE Transactions on Wireless Communications*, vol. 3, no. 5, pp. 1669–1677, Sept. 2004.

[79] J. Cho and Z. J. Haas, "On the throughput enhancement of the downstream channel in cellular radio networks through multihop relaying," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 7, pp. 1206–1219, Sept. 2004.

[80] Z. Fu, *et al.*, "The impact of multihop wireless channel on TCP performance," *IEEE Transactions on Mobile Computing*, vol. 4, no. 2, pp. 209–221, Mar.–Apr. 2005.

[81] T. Issariyakul and E. Hossain, "Analysis of end-to-end performance in a multi-rate wireless network for different hop-level ARQ policies," in *Proceedings of IEEE Global Communications (Globecom'04)*, Nov. 2004.

[82] C.-Y. Wan, A. T. Campbell, and L. Krishnamurthy, "PSFQ: A reliable transport protocol for wireless sensor networks," in *Proceedings of ACM International Workshop on Wireless Sensor Networks and Applications (WSNA'04)*, Sept. 2002, pp. 1–11.

[83] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proceedings of the* 10^{th} *ACM Annual International Conference on Mobile Computing and Networking (MobiCom'04)*, Sept. 2004, pp. 114–128.

[84] W. H. Press, *et al.*, *Numerical Recipes in C++ : The Art of Scientific Computing*, 2nd ed. New York: Cambridge University Press, 2002.

[85] W. R. Stevens, *TCP/IP Illustration vol. 1: The protocol.* Massachusetts: Addison Wesley Publishing, 1994.

[86] T. Issariyakul and E. Hossain, "Optimal radio channel allocation for fair queuing

in wireless data networks," in *Proceedings of IEEE International Conference on Communications (ICC'03)*, May 2003.

[87] T. Issariyakul and E. Hossain, "Performance modeling and analysis of a class of ARQ protocols in multi-hop wireless networks," to appear in *IEEE Transactions on Wireless Communications*, 2005.

[88] T. Issariyakul, E. Hossain, and A. S. Alfa, "End-to-end batch transmission in a multi-hop and multi-rate wireless network: Latency, reliability, and throughput analysis," to appear in *IEEE Transactions on Mobile Computing*, 2005.

VITA

Surname:	Issariyakul	***Given Names:***	Teerawat
Place of Birth:	Thailand	***Date of Birth:***	Dec. 07, 1976

Educational Institutions Attended

Thammasat University	1993 to 1997
Asian Institute of Technology (AIT)	1997 to 1999

Degrees Awarded

B.Eng. in Electrical Engineering, Thammasat University	1997
M.Eng. in Telecommunications, Asian Institute of Technology (AIT)	1999

Honors and Awards

Full scholarship sponsored by Telephone Organization of Thailand	1997-1999
Telecommunications Research LABS (TRLabs) Research Graduate Scholarship	2001-2005
University of Manitoba Graduate Fellowship (UMGF)	2003-2005

Journal Publications

1. Teerawat Issariyakul, Ekram Hossain, and Attahiru Sule Alfa, "End-to-end batch transmission in a multi-hop and multirate wireless network: Latency, reliability, and throughput analysis," to appear in *IEEE Transactions on Mobile Computing* (30-page manuscript).
2. Teerawat Issariyakul and Ekram Hossain, "Channel-quality-based opportunistic scheduling with ARQ in multi-rate wireless networks: Modeling and analysis," to appear in *IEEE Transactions on Wireless Communications* (11 pages).
3. Teerawat Issariyakul and Ekram Hossain, "Performance modeling and analysis of a class of ARQ protocols in multi-hop wireless networks," to appear in *IEEE Transactions on Wireless Communications* (9 pages).
4. Teerawat Issariyakul and Ekram Hossain, "ORCA-MRT: An optimization-based approach for fair scheduling in multirate TDMA wireless networks," to appear in *IEEE Transactions on Wireless Communications* (12 pages).
5. Teerawat Issariyakul and Ekram Hossain, "Performance bound of dynamic for-

ward link adaptation in cellular WCDMA networks using high-order modulation and multicode formats," *IEE Electronics Letters*, vol. 40, no. 2, pp. 132-133, Jan. 2004.

6. Teerawat Issariyakul, Ekram Hossain, and Dong In Kim, "Medium access control protocols for wireless mobile ad hoc networks: Issues and approaches," *Wiley Interscience Wireless Communications and Mobile Computing (WCMC)*, vol. 3, no. 8, Dec. 2003.

Conference Publications

1. Teerawat Issariyakul, Dusit Niyato, Ekram Hossain, and Attahiru Sule Alfa, "Exact Distribution of Access Delay in IEEE 802.11 DCF MAC," *in Proceedings of IEEE Global Telecommunications Conference (Globecom'05)*, on Nov., 2005, St. Louis, USA.
2. Teerawat Issariyakul, Ekram Hossain, and Attahiru Sule Alfa, "Markov-based analysis of end-to-end batch transmission in a multi-hop wireless networks," *in Proceedings of IEEE International Conference on Communications 2005 (ICC'05)*, May, 2005, Seoul, Korea.
3. Teerawat Issariyakul, Ekram Hossain, and Attahiru Sule Alfa, "Analysis of latency for reliable end-to-end batch transmission in multi-rate multi-hop wireless networks," *in Proceedings of IEEE International Conference on Communications 2005 (ICC'05)*, May, 2005, Seoul, Korea.
4. Teerawat Issariyakul and Ekram Hossain, "Analysis of end-to-end performance in a multi-hop wireless network for different hop-level ARQ," *in Proceedings of IEEE Global Telecommunications Conference 2004 (Globecom'04)*, Nov., 2004, Dallas, Texas.
5. Teerawat Issariyakul and Ekram Hossain, "Throughput and temporal fairness optimization in a multirate TDMA wireless network," *in Proceedings of IEEE International Conference on Communications 2004 (ICC'04)*, Jun., 2004, Paris, France.
6. Teerawat Issariyakul and Ekram Hossain, "Optimal radio channel allocation for fair queuing in wireless data networks," *in Proceedings of IEEE International Conference on Communications 2003 (ICC'03)*, May, 2003, Anchorage, Alaska.
7. Teerawat Issariyakul and Ekram Hossain, "Designing wireless fair queuing MAC protocols using optimization techniques," *in Proceedings of IEEE International Symposium on Advances in Wireless Communications 2002 (ISWC'02)*, Sept., 2002, Victoria, British Columbia, Canada.
8. Teerawat Issariyakul and Tapio Erke, "Individual location area management for cellular mobile networks," *in Proceedings of International Symposium on*

Wireless Personal Multimedia Communications 2000 (WPMC'00), pp.115-119, Nov., 2000, Bangkok, Thailand.