Queueing analysis for cross-layer design with adaptive modulation and coding

Xingyu Han

Submitted for the degree of Doctor of Philosophy

School of
Electronic Engineering and Computer Science
Queen Mary, University of London

August 2015

Statement of Originality

I, Xingyu Han, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I accept that the College has the right to use plagiarism detection software to check the electronic version of the thesis.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

Signature: XINGYU HAN

Date: 23/08/2015

Abstract

With the development of wireless networks, Quality of Service (QoS) has become one of the most important mechanisms to improve the system performance such as loss, delay and throughput. Cross-layer design is seen as one of the main approaches to achieve QoS provisioned services in contrast to the well-adopted TCP/IP network model. This thesis focuses on the cross-layer design incorporating queueing effects and adaptive modulation and coding (AMC), which operates at both the data-link layer and the physical layer, to obtain the performance analyses on loss, delay and throughput using the matrix geometric method. More specifically, this thesis explores the potential to extend the cross-layer analysis, at the data-link and the physical layer respectively.

At the data-link layer, since the traffic types such as voice, video and data are proven to be bursty, and the well-adopted Poisson arrivals fail to capture the burstiness of such traffic types, the bursty traffic models including ON-OFF and aggregated ON-OFF arrivals are introduced in the cross-layer analysis. This thesis investigates the impact of traffic models on performance analysis, identifying the importance of choosing the proper traffic model for cross-layer analysis.

At the physical layer, IEEE 802.11ac standard is adopted for the cross-layer analysis. In order to meet the specifications of 802.11ac with higher-order Modulation and Coding Schemes (MCS), wider channel bandwidth and more spatial streams, the Signal-to-Noise Ratio (SNR) thresholds are re-determined for the AMC; in addition, a single user (SU) multiple in multiple out (MIMO) spatial multiplexing system with zero-forcing (ZF) detector is adopted for the cross-layer analysis. Furthermore, this thesis explores the impact of antenna correlations on the system performance.

All of the work done in this thesis aims at obtaining more practical performance analysis on the cross-layer design incorporating queueing effects and AMC. The proposed cross-layer analysis is quite general, so that it's ready to be applied to any QoS provisioned networks.

Acknowledgement

Firstly, I would like to convey my gratitude to my supervisor, Dr. John Schormans, for your guidance and patience throughout these four years. This thesis cannot be accomplished without your caring and encouragement. You not only teach me the way of thinking for the research, but also show me how to cherish the family, friends and other peoples around. I'll never forget the first time we met at Eng209 when you patiently answered questions from students after class; I'll never forget the moment when you talked about your wedding anniversary; and I'll never forget a picture taken from your back for a Network Group barbecue gathering. Actually, I just can't express how grateful I am to you. You're more like a father than a supervisor to me. I'm so lucky to accept your guidance, and I'll benefit from it all the time.

Secondly, I want to thank my family for the love you give to me. I'm always being selfish to pursue my unrealistic freedom, and I often make wrong decisions in life. You're always ready to forgive me and help me out. I know I'm not able to pay back temporarily, but I'm learning to be mature enough to take responsibilities. I love you all.

Thirdly, I just need to express my appreciation to my ever beloved ex-girlfriend. I'm so happy to share my life with you for last 7 years. Thanks for all the efforts and promises we made together. All those happiness and sorrow between us will be recorded in my brain. I've heard you got married recently. I wish you all the best.

Fourthly, I must give my thanks to my friends and colleagues. Thanks for all the happy time we spent together. I know we may split, we may contact little for years and we may settle down around the world, but I'll keep everything in mind with best wishes to you all.

Finally, I should thank myself for being able to develop normal personalities, which give me the courage to face the difficulties and accept the life in the future.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

4G	4^{th}-Generation
5G	5^{th}-Generation
i.i.d.	independent and identical distributed
p.m.f.	probability mass function
AM	Adaptive Modulation
AMC	Adaptive Modulation and Coding
AP	Access Point
ARQ	Automatic Repeat-reQuest
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase-Shift Keying
CDF	Cumulative Distribution Function
CR	Cognitive Radio
CSI	Channel State Information
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
CTMC	Continuous Time Markov Chain
DBMAP	Discrete-time Batch Markovian Arrival Process
DCF	Distributed Coordination Function
DF	Decoded-and-Forward
DQCA	Distributed Queueing Collision Avoidance
DSRC	Dedicated Short Range Communications
DTMC	Discrete Time Markov Chain
ECC	Error Correction Code
FFT	Fast Fourier Transform
FIFO	First In First Out

FSMC	Finite State Markov Chain
GI	Guard Interval
HSDPA	High Speed Downlink Packet Access
IDFT	Inverse Discrete Fourier Transform
IFFT	Inverse Fast Fourier Transform
ISI	Inter-Symbol Interference
LCR	Level Crossing Rate
LTE	Long Term Evolution
MAC	Medium Access Control
MC	Markov Chain
MCS	Modulation and Coding Scheme
MIMO	Multiple In Multiple Out
ML	Maximum Likelihood
MMPP	Markov Modulated Poisson Process
MMSE	Minimum Mean Squared Error
MRC	Maximal-Ratio Combining
MS	Modulation Scheme
MU	Multi-User
OFDM	Orthogonal Frequency-Division Multiplexing
PDP	Packet Drop Probability
PER	Packet Error Rate
PLP	Packet Loss Probability
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase-Shift Keying
RF	Radio Frequency

RLC	Radio Link Control
SE	Spectral Efficiency
SER	Symbol Error Rate
SNR	Signal-to-Noise Ratio
SU	Single User
ZF	Zero-Forcing

Chapter 1 Introduction

1.1 Background & Motivation

Wireless networks have been under detailed investigation during recent decades, not only because of the mobility and convenience they've provided for mobile subscribers and any other devices which have wireless-enabled functionality, but also because the extra complexity in design and implementation they've created due to the restrictions of wireless channel variations and spectrum scarcity. This is very different from the static channel conditions and static resource allocations in wired networks. The whole history of the development of wireless networks is one of combating channel variations with limited spectral resources in order to make sure that more and more users are supplied with Quality of Service (QoS) provisioned services.

With the development of multimedia packet-based wireless networks, users are no longer only satisfied with basic data or audio services. Therefore, more and more services like video streaming, real-time video chatting and faster reliable data transfer have emerged to meet the users' needs. As a result, multimedia packet-based wireless networks require high data rates with the provision of higher spectral efficiency (SE) and controlled packet loss probability (PLP) and packet latencies (delays).

Channel variations are mainly caused by the mobility of wireless-enabled devices and the impact of fading. Most of the existing wireless networks use adaptive modulation and coding (AMC) [6] at the physical layer as a means of combating channel quality variations, i.e. as the channel gets worse a modulation and coding scheme (MCS) with fewer bits per symbol is chosen for transmission by AMC, while as the channel gets better an MCS with more bits per symbol is chosen. This is to maintain the same target bit error rate (BER) by reacting to the changing channel conditions. As a result, AMC has become one of the key technologies used to combat channel variations and maximize system throughput.

However, traditional analyses of AMC such as [6] [8] are based on an assumption that AMC operates at the physical layer alone, and these papers don't consider the impact of queueing and buffering at the data-link layer. More specifically, traditional analyses

on AMC assume there are always data packets for the system to transmit and yet these never cause buffer overflow (i.e. the saturation model), which is unrealistic in real world packet-based wireless network scenarios. As indicated, current analyses cannot predict the overall system behavior at the packet level. Therefore, queueing analysis which takes into account the conditions of the data-link layer becomes necessary. During the last decade, cross-layer performance evaluations combining queueing analysis at the data-link layer and AMC at the physical layer have received a lot of attention in various scenarios such as [28] [30] [33] [34]. These include single-user (SU) and multi-user (MU) with single channel, MIMO and multi-hop relay studies. This cross-layer approach to performance evaluation makes it possible to analyze the probability of queue length in the buffer, the packet drop probability (PDP) due to buffer overflow, the average packet delay, and further obtain the PLP and average throughput while accounting for the impact of fading at the physical layer.

At the transmitter side, this kind of cross-layer design can be well modelled with queueing theory. The kind of queue model used by most of the cross-layer designs found in published works such as [22] [28] [30] [31] [34] [42] are working-vacation queue models. Typically, queue models with working vacations [2] assume a lower service rate during vacations compared to the busy period, instead of the server completely stopping as in a conventional queue vacation model (in fact, working vacation queue model can be generalized as the special case of vacation queue model), as shown in Figure 1.1.

The durations for both busy periods and working vacation periods ($0 - t_1, t_2 - t_3$ for non-vacation periods and $t_1 - t_2, t_3 - t_4$ for working vacation periods) are usually assumed to follow the exponential distribution, which is suitable to build up a Markov Chain (MC) to identify the transition probabilities between busy and working vacation periods. However, the design for server working-vacation model shown in Figure 1.1 only applies two independent service rates. Fortunately, the published works such as [28] have extended this original server-vacation model into a range of working vacations with different service rates, which is suitable to combine physical layer techniques such as AMC for cross-layer analysis.

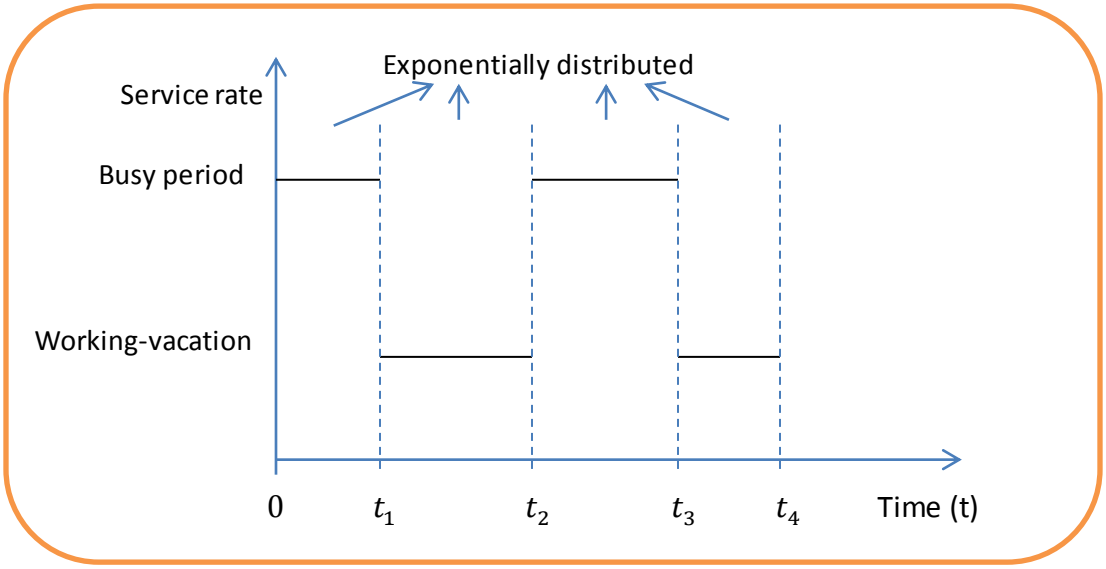

Figure 1.1 Service rate for busy period and working vacations

Queueing analyses on cross-layer designs adopt a range of working vacations with different service rates, to fit with the MCSs used, at the physical layer. More specifically, most of the cross-layer designs found in the literature such as [28] [31] [34] utilize an M/D/s(t)/K queue model, in which the packet arrivals are homogenous (i.e. single class) and collectively form a Poisson arrival process, the buffer service rate (which is directly controlled by the MCSs) varies over time, and the buffer size K is finite.

As indicated above, the cross-layer design, which considers both queueing effects at the data-link layer and AMC at the physical layer with application to various wireless scenarios, has been well studied in the literature. These published works also provide the basic framework of performance analysis on cross-layer design. However, there are still some interesting problems left to be solved on the cross-layer queueing analysis, which motivate the work completed in this thesis.

Motivation 1: The applications of bursty traffic models for cross-layer analysis

Most of the current literature such as [22] [28] [30] [31] [34] [42] apply Poisson arrivals as the only packet arrival process, for the memorylessness of Poisson arrivals provide convenience in queueing analysis; however, in practice Poisson arrivals only approximate well when packet arrivals are non-bursty or there's no correlation between successive arrivals. With the development of the packet-based wireless networks, more and more published work [44]-[47] [49] reveal that the behaviors of

arrival traffic patterns, such as voice, video and data center, turn out to be bursty. As a result, adopting a traffic model, which can capture the burstiness of the arrival packets, becomes necessary for practical performance analysis.

In addition, choosing the proper arrival process based on arrival stream type before queueing analysis should be investigated as the complement of the state-of-the-art. As a generalization, the queueing behavior at the buffer suits a G/D/s(t)/K vacation queue model, where G indicates that the arrival process follows a general distribution including not only Poisson, but also more generally distributed arrivals.

Motivation 2: The extension to IEEE 802.11ac standard

As the packet-based wireless networks develop, it is necessary for industry to fulfil wireless standards with higher data rates to meet the QoS provisions for users. As a result, wireless standards have begun to consider higher-order MCSs, wider channel bandwidth and more spatial streams. As specified in [52] [53], the IEEE 802.11ac standard has become the most promising WLAN standard in the next three to five years as the substitute for IEEE 802.11n. IEEE 802.11ac, often seen as 5th-generation (5G) Wi-Fi standard, has already been supported by WLAN routers and handsets since 2013. As predicted in [52], there will be one billion 802.11ac enabled devices by 2015, and the number of commercialised 802.11ac enabled devices is highly likely to reach this at the current rate of growth. With the rapid spread of 802.11ac, the cross-layer analysis with modified physical layer specifications, including higher-order MCSs, wider channel bandwidth and more spatial streams, become interesting to study.

More specifically, the extension to 802.11ac for cross-layer analysis can be divided into two tasks:

Task 1: The re-determination of SNR thresholds for AMC

As mentioned above, the AMC scheme found in the literature such as [6] [22] adopts a range of MCSs with different service rates at the physical layer. And one of the most important steps for any AMC algorithm is to determine the SNR thresholds for MCSs. In the field of interest, SNR thresholds are mainly dependent on the physical layer specifications. Therefore, maintaining updated functional blocks for data processing

becomes quite important. Within the last 5 years, OFDM has been adopted as one of the key technologies in most popular packet-based wireless networks and standards such as 4th-generation cellular mobile network (4G) and 802.11a, g, n, ac. In general, OFDM technology splits the whole bandwidth into several parallel orthogonal subcarriers with frequency-flat fading for transmission. Compared to single carrier modulation and the original FDM, OFDM achieves higher spectral efficiency with the same bandwidth, which is important as bandwidth is always relatively scarce in wireless networks. With the development of the Fast Fourier Transform (FFT), it was possible for OFDM to be applied into packet-based wireless networks at the physical layer with low complexity (this is because OFDM modulation is equivalent to taking an IFFT operation to bit streams). This ensures OFDM is a useful and convenient tool to help combat channel variations for frequency-selective channels.

However, queueing analyses on cross-layer design found in the literature such as [22] [28] [30] [31] [34] [42] do not take OFDM into consideration; although some of these claim to use OFDM, they actually adopt the same SNR thresholds originated in [22] in which no evidence has been shown to include OFDM. In addition, other physical layer specifications for 802.11ac such as higher-order MCS (up to 256 QAM with a coding rate of 5/6) and more channel bandwidth (up to 80+80 or 160 MHz) also demand the re-determination of SNR thresholds for MCSs adopted.

Task 2: The extension to MIMO at the air interface

As specified in [52] [53], the Access Points (APs) supporting 802.11ac standard can be equipped with up to 8 antennas, providing the ability to simultaneously transmit up to 4 spatial streams for each user, which leads to the use of MIMO both at the transmitter and the receiver side. In general, the usage of MIMO can be classified into two fields: diversity and spatial multiplexing. The diversity technique transmits the same piece of data on all antennas to achieve the diversity order, which is, in other words, to make the channel more reliable; while the spatial multiplexing technique exploits the spatial order to transmit different data on different antennas, which dramatically increase the channel capacity. As indicated by the specifications of 802.11ac, the introduction of MIMO is mainly used for obtaining spatial multiplexing.

In addition, 802.11ac is the first Wi-Fi standard to propose MU MIMO; while maintaining SU MIMO proposed in 802.11n. In general, SU MIMO allows AP to exchange data with only one user with multiple antennas at a time; while MU MIMO allows AP to exchange data with multiple users simultaneously. Compared to SU MIMO, MU MIMO requires further operations, usually beamforming, for AP to distinguish between the multiple users in different directions. As a result, although SU & MU MIMO bring out extra complexity for the system, they dramatically increase the spectral efficiency of the channel.

For the state-of-the-art, most of the published works such as [28] [34] [42] only consider single antenna scenarios for cross-layer analysis; for those which consider MIMO such as [30] [31], they either concentrate on the diversity technique or fail to provide a reasonable system model for SU or MU MIMO as specified by 802.11ac. Thus, the extensions at the air interface become necessary for the cross-layer analysis on IEEE 802.11ac standard.

In summary, the motivations based on the queueing analysis on cross-layer design both at the physical layer and the data-link layer are detailed in order to meet the specifications of IEEE 802.11ac standard. The functional block of the proposed cross-design, which indicates main modifications and extensions, is shown in Figure 1.2.

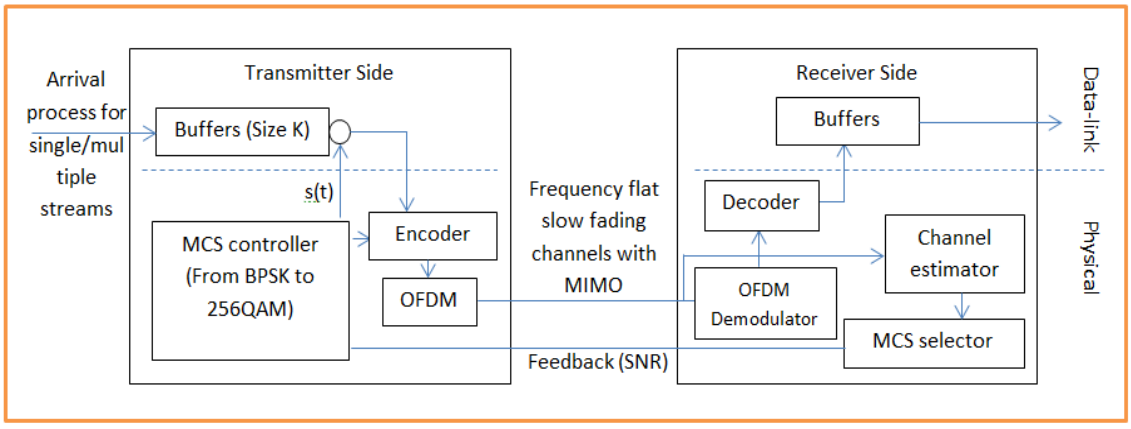

Figure 1.2 Functional block diagram for proposed cross-layer design

1.2 Organisations of the thesis

The rest of the thesis is organised as follows. Chapter 2 is the literature review on cross-layer design; the main contributions of this thesis are also given at the end of Chapter 2. Chapter 3 gives formulas, implementations and validations of the basic M/D/1 queue model with and without buffer and vacation, laying the foundation of the application of queueing analysis using the chosen matrix geometric method. Chapter 4 discusses AMC and operates it with a Poisson arrival process to achieve a traditional single user cross-layer design; performance analyses on queue length in the buffer, PDP due to buffer overflow, average packet delay, PLP and throughput, are the main focus as we look into the proposed cross-layer design. Chapter 5 modifies the arrival process to incorporate ON-OFF arrivals and aggregated ON-OFF arrivals at the data-link layer, and so facilitate evaluation of performance in cross-layer design and implementation. Chapter 5 also discusses the importance of using a properly representative arrival processes. Chapter 6 extends the cross-layer design at the physical layer to meet the specifications of IEEE 802.11ac by re-determining SNR thresholds for AMC and extending the scenario to single-user (SU) MIMO at the air interface. Chapter 7 concludes the thesis and provides possible future work.

Chapter 2 Literature Review & Basic Concepts

2.1 Literature review

2.1.1 Queue models with working vacations

Working vacation queue models originate from vacation queue models. There has been extensive literature on vacation queue models, most of which dates back to 30 years ago, and the most up to date one is [1]. In [1], the authors focus on the M/D/1 vacation queue model with general distributed busy and vacation periods.

After [1], the scholars turned their interest onto queue models with working vacations. As discussed in Chapter 1, queue models with working vacations can be good tools for queueing analysis on any cross-layer design when the model adopts a range of working vacations with different service rates to incorporate MCSs used by AMC. The emergence of this type of queue model partly originated from wireless network scenarios, and was first proposed by [2].

In [2], the authors analysed one of the most common queue models: the M/M/1 model with working vacations. The authors applied this model into a scenario in which there are multiple data streams waiting to be transmitted. A token moves among these streams: when one data stream gets the token, it will transmit with a higher data rate – it is as if its corresponding server went into a busy period. The queue model is well-fitted for a system with two transmission modes. In addition, [2] is the first attempt to relate vacation queue models to wireless networking scenarios.

After [2], more related papers appeared which have targeted more queue models [3]-[5]. [3] analysed the M/G/1 working vacation model with general or exponential distributed busy and vacation periods. [4] and [5] concentrates on the G/Geo/1 and Geo/Geo/1 working vacation models respectively with exponential distributed busy and vacation periods.

Moreover, it is worth mentioning that [4] and [5] firstly use a different analytical method compared to former published works. The method, which is known as the matrix geometric method, is aiming at obtaining analytical results by building up MCs

for queue state transitions taking arrival and service processes into account. Normally, the steady-state distribution for the queue state can be obtained by analysing the eigenvectors of the queue state transition matrix.

In general, most of the literature on queue models, no matter if queue models experience vacations or not, try to achieve closed-form expressions for average queue length and average waiting time for each customer in different scenarios. As introduced above, there are two main kinds of methods used for the queueing analysis: algebraic method and matrix geometric method.

Compared to the matrix geometric method, the algebraic method often requires more mathematical manipulations including integration, differentiation and transforms in order to get closed-form expressions. And closed-form expressions that can be obtained by the algebraic method are often quite complicated even for a simple vacation queue model. Alternatively, the matrix geometric method aims at building a repetitive state transition matrix block and then tries to obtain the steady-state queue length distribution, and is comparatively straightforward. Although the matrix geometric method has its advantages in queueing analysis, the method still has its limitations. The main challenges of using matrix geometric method are focussed on:

1. Whether there exists a repetitive state transition matrix block, which limits the method to the scope of quasi-birth-death process, continuous-time Markov chain (CTMC) or discrete-time Markov chain (DTMC).

2. Whether there exists a steady-state distribution for the state transition matrix block, which requires the method to analyse reducibility, homogeneity and positive recurrence of the state transition matrix.

The challenges of using the matrix geometric method are discussed in this thesis, in particular the feasibility of applying this method into our proposed cross-layer design analysis, as this is essential in order to evaluate the target metrics.

In summary, the literature related to working vacation queue model only considers the case with two service rates, i.e. busy and working vacation service rates. In addition, these papers mainly concentrate on a theoretical analysis and generally lack

applications in practical scenarios. Working vacation models with only two states cannot be used to address the wireless system models considered in this thesis. All of the reasons above require us to extend the model with a range of working vacations with different service rates to fit into cross-layer design incorporating AMC, in a more straightforward way, which is matrix geometric method, for queueing analysis.

2.1.2 Adaptive modulation and coding

AMC was first adopted by [7] in High Speed Downlink Packet Access (HSDPA, also known as 3.5G) to substitute for the traditional fast power control used in CDMA for channel adaptation. AMC adapts to the channel variations such as to maximize the system throughput. [7] summarizes the advantages and challenges of using AMC as a channel adaptation technology. As suggested in [7], AMC is susceptible to feedback channel error and end-to-end packet transmission delay. To solve these two problems, we make the following assumptions in this thesis:

1. For feedback channel errors, it is reasonable to assume an error-free feedback channel because the feedback payload is much smaller compared to the overall throughput. The system can ensure an error-free feedback at the cost of minor payload increase which is negligible.

2. For packet transmission delay, it is reasonable to assume that the round-trip time for transmission is negligible compared to the duration for each timeslot; the timeslot is regarded as the basic time unit for each MCS adjustment, especially for WLAN standards such as 802.11a, g, n, ac.

3. For channel fading conditions, it is reasonable to assume slow fading in WLAN scenarios because the mobility of devices and the rate at which the environment change are both quite slow.

[7] also adopted a range of MCSs in HSDPA, but research has shown that 64QAM was rarely used. With the development of adaptive antenna technology and more efficient coding and decoding techniques, it is now feasible to use higher-order modulation schemes. In 802.11a standard, 64QAM has been much more commonly used. It is also suggested in [53] that 256QAM could be achieved in the IEEE 802.11ac standard.

Both [7] and [53] gave an overview of the development of AMC in different standards. However, they did not give any guidelines for analysing channel adaptation theoretically, which leaves a gap in the use of theory (and simulation), making AMC hard to study by performance analysis.

Before the investigations of AMC, the studies to apply finite-state Markov chain (FSMC) for channel variation analysis had become a field under heated discussion [11]-[13]. [11] and [12] firstly introduced FSMC into slow Rayleigh fading channel modelling. Compared to the traditional Gilbert-Elliot channel, which classifies the channel condition only as good and bad, FSMC extends the two-state channel with more feasible states, which can better describe the channel variations. By analysing the quasi-birth-and-death process suggested by slow fading, [11] and [12] divided the whole SNR scope into several thresholds to indicate the channel state transitions, which resulted in the analysis on the second-order statistics on Rayleigh fading channel such as level crossing rate (LCR) which is detailed later for MIMO channel modelling. However, the SNR threshold settings in [11] and [12] didn't consider the combination of MCSs. Although [13] tried to divide SNR in terms of delay for each MS, the threshold settings still followed the old way as in [11] and [12].

Inspired by the studies on FSMC, there have been several published papers investigating rate adaptation strategies like adaptive modulation (AM) before AMC. Actually, AMC originated from AM, and AMC just combines AM and coding strategies, including encoding and puncturing operations, to make the bit streams more reliable under transmission. [6] was the first paper to analyze AM theoretically over a Nakagami-m fading channel. [6] concluded that a channel adaptation strategy like AM is the key factor in maximizing the throughput and spectral efficiency, compared to an old-fashioned power adaptation strategy in CDMA. The results in [6] secured a promising future for a channel adaptation strategy such as AMC as one of the key physical layer techniques. In addition, [6] provided a method for setting thresholds for each modulation scheme (MS) to achieve the rate adaptation strategy, which lays the foundation for a slot-based queueing analysis at the physical layer. After [6], [8] gave more detailed work on the AM strategy, concluding that restricting the transmit power such as to make it always constant also achieves a nearly optimal spectral efficiency for

the rate adaptation; this simplifies the SNR threshold settings for AM as well as AMC strategies.

The development of FSMC and AMC resulted in the combination of them for channel variation analysis, especially after the introduction of OFDM. Compared to [11] and [12], [6] and [8] set SNR thresholds for MSs in terms of bit error rate (BER), which provided a more practical approach for OFDM-based networks. However, the focus of [6] [8] were only on the physical layer. Although [9] managed to detail AMC strategy in an OFDM system, and [10] took a step further by using packet error rate (PER) to set SNR thresholds with AMC strategy, all of the literature mentioned above didn't consider the impact of the queueing behavior at the data-link layer on AMC. Because this impact does exist in practical scenarios, it is necessary to investigate the cross-layer analysis on how the physical layer and the data-link layer affect each other.

2.1.3 Cross-layer design

As discussed in sections 2.1 and 2.2, the existing queue models with working vacations are impractical for applications with AMC, and the literature related to AMC mentioned above lacked the consideration of the effects at the data-link layer. The cross-layer design, which can be abstracted as a queue model with a range of working vacations, is obtained by combining the packet-level effects as well as mechanisms at the data-link layer and AMC at the physical layer. An extensive amount of work has been done to study the cross-layer design and analysis.

2.1.3.1 Overview

The classical five-layer TCP/IP model standardizes the communication networks, and it is beneficial for network implementations. However, as the development of communication systems, the nature of the TCP/IP model, which indicates that the information cannot be shared throughout layers, becomes an obstacle to meet the requirement of each user, as well as to the optimisation of the overall system performance. As a result, the emergence of cross-layer design and analysis becomes necessary. As generalised by [21], the goals of cross-layer designs, which is shown by Figure 2.1, are security, QoS and mobility, and any cross-layer design should satisfy at least one of these goals.

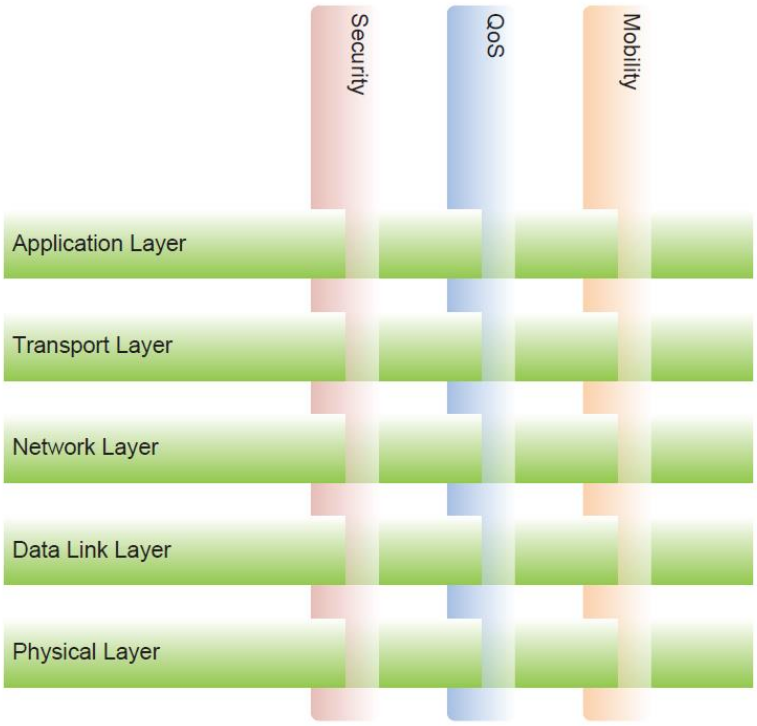

Figure 2.1 The goals of cross-layer designs [BF,YX,HD,HZ]

Among these three goals, the most investigated one is QoS. Cross-layer designs which aim at achieving QoS provisions often focus on bitrate, delay, loss probability and throughput under various constraints such as power, delay and error rate. Therefore, the QoS provisioned cross-layer designs including scheduling and resource allocations have become one of the most important aspects of modern packet-based wireless networks. A large amount of work has been done to overview the cross-layer designs form different aspects during last decade, such as [17]-[21].

Among these published works, [17] was one of the earliest papers to overview cross-layer designs. In [17], the authors investigated cross-layer designs on TCP, MU scenario in cellular network and the deployment of APs, and [17] also talked about the status of cross-layer design standardization and the interoperability of WLAN and cellular networks. After [17], [18] and [19] studied cross-layer designs for 3G & B3G CDMA-based networks, and problems including cross-layer resource allocation and scheduling schemes were discussed in these two papers. After that, the investigations of cross-layer designs on multi-hop networks [20] were given, and these led to more cross-layer problems such as joint routing and channel assignment. Until recently, [21] classified

cross-layer designs with two dimensions: for the link level, cross-layer designs are classified as manager and non-manager methods; while for the network level, cross-layer designs are classified as distributed and centralised methods. In addition, a discussion of the challenges including coexistence, signalling, overhead, lack of standardization and destruction of the layered model were also given in [21].

Following the timeline of the development of cross-layer designs, it can be concluded that the investigations on cross-layer designs have become more and more systematic, and it is nearly impossible to standardise cross-layer designs because they've covered nearly all fields of modern packet-based wireless networks by maintaining the classical layered model. Therefore, finding a unified solution to cross-layer designs becomes impossible, and instead we have to concentrate on specific layers for particular cross-layer analysis. Based on sections 2.1 and 2.2, the cross-layer design which operates at the physical and data-link layers is our research focus. In addition, we concentrate on the link level cross-layer design because we investigate cross-layer effects and mechanisms at the lowest two layers. As a summary, Table 2.1 has given the mechanisms and parameters analysed in lower-layer cross-layer design.

Table 2.1 Parameters analysed in lower-layer cross-layer design

	Mechanism	Parameters
Data-link layer	Error correction	Link layer PER
	Retransmissions	No. of retransmissions N_r^{max} (ARQ)
	Queueing effects	Queue length, queueing delay
Physical layer	Adaptive modulation and coding	Mode n, SNR γ thresholds (channel fading)

Based on Table 2.1, we review the literature on two aspects, most of which concentrate on data-link layer mechanisms and parameters with AMC:

1. Cross-layer design incorporating Automatic Repeat-reQuest (ARQ)

2. Cross-layer design incorporating queueing effects

2.1.3.2 Cross-layer design incorporating ARQ

Cross-layer design incorporating ARQ tries to combine ARQ protocol operated at the data-link layer with AMC at the physical layer. The utilization of ARQ at the data-link layer normally relieves the stringent error control requirement at the physical layer by setting up the maximum number of retransmissions for error packets. Lots of the published work has investigated this kind of cross-layer design incorporating ARQ such as [22]-[27].

As far as we know, [22] was the first attempt to complete a cross-layer design with AMC. This paper was the first one that used packet-level error rate constraints instead of physical-level BER utilized in [6] for MCS threshold setting. In addition, [22] adopted a range of MCSs, together with their asymptotic statistics and expressions for PER behaviours which have been used by many other related published work, specified in IEEE 802.11a with reasonable assumptions, which shows a way to apply a cross-layer design into WLAN standards. In summary, [22] provided a framework and procedure to analyse AMC at the physical layer in slot-based systems.

After [22], the other researchers tried to apply this kind of cross-layer design incorporating ARQ to many other practical system models or networks. [24] managed to combine ARQ and AMC in a diversity MIMO system with STBC codes by using SISO equivalence for MIMO channel model. In addition, [24] tried to re-determine SNR thresholds, which agree to the values we obtained in Appendix A, for higher-order MCSs such as 256QAM. [25] applied the cross-layer design into a cognitive radio (CR) network, and obtained the SNR distributions for secondary users in order to optimise the spectral efficiency for secondary users. [26] investigated the cross-layer design incorporating ARQ in IEEE 802.11p standard, namely, the Dedicated Short Range Communications (DSRC) system, to maximize system spectral efficiency. And [27] investigated the cross-layer design incorporating ARQ for Long Term Evolution (LTE) system. In LTE, ARQ is used by two sublayers of the data-link layer, namely Radio Link Control (RLC) sublayer and Medium Access Control (MAC) sublayer. Therefore, two independent ARQ mechanisms were used and discussed in [27].

In summary, the literature mentioned above provided a similar classical method for cross-layer analysis incorporating ARQ. All of them tried to maximize the system throughput and spectral efficiency under prescribed delay and PER requirements, which were suitable for various practical scenarios. However, these published works also have limitations, which are listed below:

1. They usually assume a saturated queue, which means there are always packets waiting to be transmitted (while ignoring buffer overflow), which is totally unrealistic in practical scenarios such as IEEE 802.11 standards.

2. They assume the buffer size is infinite at the data-link layer, thus they didn't consider the possible packet drops due to buffer overflow. Although delay constraint is a reasonable factor to consider, we need a more general approach to include both buffer overflow and delay in order to better understand the system behaviour.

Both of the reasons listed above motivate continuing work on queueing analysis with finite buffer size on the cross-layer design, i.e. the cross-layer design incorporating queueing effects.

2.1.3.3 Cross-layer design incorporating queueing effects

Different from the cross-layer design incorporating ARQ, we can obtain more metrics, including the stationary probability distribution for queue length in the buffer, PDP due to buffer overflow, average packet delay, PLP and system throughput, through cross-layer analysis incorporating queueing effects. An extensive amount of work on the cross-layer design incorporating queueing effects has been done to cover various scenarios and mechanisms such as [28]-[36].

Among them, [29] and [30] adopted delay constraint analysis by effective capacity approach to obtain the PLP and throughput for the cross-layer design incorporating queueing effects for MIMO diversity systems. They used asymptotic expressions for queue length in the buffer larger than B assuming this tail probability decays exponentially. However, the behaviour of queue state in the buffer is impossible to obtain if the assumption on tail probability doesn't stand for the effective capacity approach. In addition, both [29] and [30] assumed an infinite buffer size, which is

impractical. Therefore, queueing analysis, which investigates the system model by building up FSMC for state transitions and obtaining the steady-state queue state distribution with the matrix geometric method, has become a more general approach than delay constraint analysis.

It is worth mentioning that queueing analysis on the cross-layer design lays the foundation of the whole scope of this thesis. In other words, any extensions specified in this thesis are based on the queueing analysis approach using the matrix geometric method.

As far as we know, [28] firstly tried to achieve the cross-layer design using queueing analysis. The authors of [28] provided a basic analysis procedure, i.e. by building up the state transition matrix for queue state at the data-link layer and channel state at the physical layer based on an embedded FSMC. In addition, [28] generalised the metrics obtained by the matrix geometric method, including also queue length in the buffer, PDP due to buffer overflow, average packet delay, PLP and throughput, which covers nearly all system metrics for link level performance analyses. Furthermore, the authors of [28] managed to maximize system throughput and minimise the PLP caused by both transmission error and buffer overflow for Poisson arrivals.

Based on the work done by [22] and [28], [23] managed to incorporate both ARQ and queueing effects for cross-layer analysis by building up the FSMC for a tuple of queue state, channel state and retransmission state. However, to our surprise, there is no continuing work after [23] extending the system model to MIMO or relay. As a comparison, the work on extensions to [28] is massive (detailed later). The reason might be because of computational complexity. Since [23] took more mechanisms into consideration, its analysis will require a much larger probability transition matrix. Although this brings extra complexity, the work in [23] is still worth further investigation and extensions.

After [28], continuing work done by other researchers [31]-[33] have proposed several extensions on MIMO. [31] firstly extended to an MIMO scenario for the cross-layer design by analysing both diversity and multiplexing MIMO systems, and an algorithm was proposed which selected to use diversity or multiplexing for transmission to

optimise the overall system throughput. For a spatial multiplexing MIMO system, the authors of [31] considered zero-forcing (ZF) detector in presence of channel correlations. However, the method they provided to obtain the stationary probability of each MCS was too general, and could not be applied to every correlation condition. [32] studied the spatial multiplexing MIMO system with ZF detector, and obtained the SNR distributions for users with imperfect channel state information (CSI); however, they assumed no channel correlations in order to obtain the closed-form expressions for the SNR distributions. [33] investigated the multiplexing MIMO system in the presence of transmit correlations, and the authors assumed Batch Bernoulli arrivals to try to capture the burstiness of packet arrivals. In addition, [33] assumed the number of transmit antennas was smaller than at the receiver, resulting in an approach for user selection. However, the system model for [33] did not consider AMC at the physical layer.

After these papers on MIMO, researchers investigated more on relay networks [34]-[36]. [34] studied a two-hop scenario with single decoded-and-forward (DF) relay, which aimed at addressing the problem of relay deployment. [35] managed to insert a sleep mode, i.e. the server incorporating AMC will stop working for a period of time if the buffer goes empty, which improves the energy efficiency especially when the system load is low. In the meantime, [36] investigated the two-hop scenario with multiple DF relays, each of which was supported by the LAZY protocol, which was equivalent to inserting a sleep mode. [36] also assumed a Markov Modulated Poisson Process (MMPP) for bursty traffic approximation, and analysed the system by truncating an infinite-state transition matrix assuming infinite buffer size.

Note that [33] and [36] had used two different traffic models to capture burstiness of arrival packets, but both of them managed to approximate the bursty arrivals to some extent.

In summary, all of the literature mentioned above follows a similar queueing analysis procedure by building up embedded FSMC and utilising the matrix geometric method to obtain system metrics, which is inspiring and efficient for analysing cross-layer design. However, all of them failed to address the following problems:

1. How to analyse practical system behaviours with various (and more practically representative) packet arrival processes. Most of the literature only assumes a Poisson arrival process, which cannot cover all types of practical traffic patterns. None of the literature stated above introduces ON-OFF or aggregated ON-OFF arrival processes, which can appropriately approximate bursty behaviour.

2. How to analyse a frequency-selective fading channel. All of the literature assumes the fading channel to be frequency-flat, which is not always the case as in practical scenarios. In order to help combat frequency-selective fading, we have to introduce OFDM technology into the physical layer. However, none of the literature stated above ever mentioned to achieve OFDM functionality.

3. How to adjust the queueing analysis to meet the requirements for new standards such as IEEE 802.11ac with higher-order modulation schemes, more channel bandwidth and more spatial streams. Most of the literature adopts the same parameters and set of MCSs in 802.11a, and none of them ever tries to make these extensions.

All of the problems stated above, together with motivations discussed in Chapter 1, require continuing work to make extensions to existing cross-layer designs, both at the physical layer and the data-link layer.

2.2 Basic concepts

2.2.1 Coding scheme

In general, the source bits coming down from the data-link layer are first processed to be encoded by a convolutional or a Turbo code, a process also known as error correction coding (ECC), at the physical layer, in order to give the ability to detect and correct errors when decoding at the receiver side after going through the wireless channel. Although ECC adds a large percentage of bits to the original source bits, it makes the encoded bits more reliable and avoids most of the retransmissions caused by transmission errors, achieving system performance improvement.

IEEE 802.11 standards define various coding rates ranging from 1/2, 2/3, 3/4 to 5/6. In general, the lower the coding rate is, the more reliable, but the more payload need to

be processed for the same number of source bits. How to determine the best coding rate for each MCS is affected by different factors, such as signal-to-noise ratio (SNR) of the wireless channel and the type of ECC we use, and this is out of the scope of our research.

In real world scenarios, the source bit encoding is often achieved by a convolutional encoder with a coding rate of 1/2, which means that the number of source bits is doubled after convolutional encoding. In order to get various coding rates other than 1/2, puncturing technology is used, together with 1/2 convolutional encoding.

2.2.1.1 Convolutional encoder

Convolutional encoder uses convolutional code with a coding rate of 1/2 to encode source bits, producing two encoded parity bits for each source bit [56]. Convolutional codes are often specified by a vector [m, n, K], where m is the number of input bits, n is the number of output bits and K is the constraint length. Therefore, the coding rate for convolutional code is obtained by m/n, and the output for each port j is relevant to previous (K-1) inputs. As indicated by its name, convolutional code performs convolution operation for the input stream with the output at each port j at timeslot i given by (2.1).

$$y_i^j = \sum_{k=0}^{K-1} h_k^j \times x_{i-k}, \text{ (2.1)}$$

x_i is the input source bit stream; y_i^j is output encoded bit stream at port j; and h_k^j is the impulse response for port j.

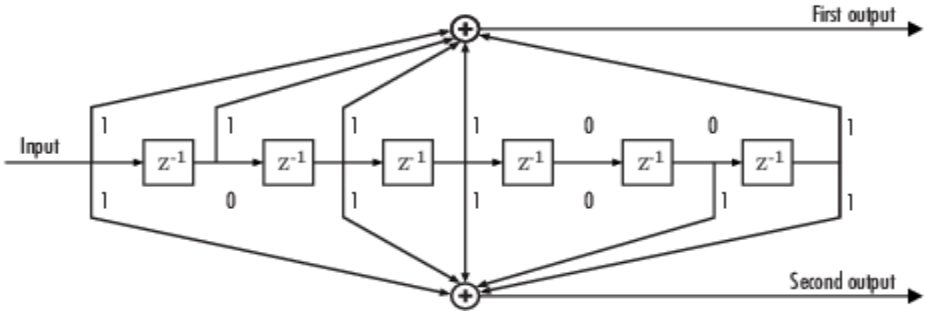

Figure 2.2 Classical convolutional encoder with the generator polynomial g = [171, 133]

The classical convolutional encoder used in industry performs 1/2 convolutional encoding with the constraint length K=7, as shown in Figure 2.2. From Figure 2.2, we get the first generator polynomial is represented by binary vector [1 1 1 1 0 0 1], which is equivalent to 171 in octal; while the second generator polynomial is [1 0 1 1 0 1 1] in binary and 133 in octal.

2.2.1.2 Puncturing patterns

As detailed in [58] and IEEE 802.11 standards, the basic operation of puncturing is to remove some bits of the encoded data at the transmitter side; while at the receiver side, the received data is extended by adding 0 to corresponding positions where bits are removed in order to prepare for further decoding.

From [58], we obtain the following observations for encoding and puncturing operations:

1. The number of source bits is better to be a multiple of 5, 6 or 9 to avoid redundant tail bits for achieving the coding rate of 5/6, 2/3 or 3/4. However, the puncturing operation also gives the puncture pattern for tail bits if tail bits exist with little impact on the overall coding rate.

2. The number of encoded bits after puncturing should be a multiple of the number of bits per symbol used by the modulation scheme; otherwise this MCS cannot be used for transmission because of redundant bits.

2.2.1.3 Viterbi decoder

The Viterbi decoder, which was proposed by Viterbi [54], applies maximum likelihood (ML) method for encoded sequence decoding [57]. The Viterbi method avoids the enumeration for all possible 2^n combinations for encoded stream with n bits, which efficiently decodes the convolutional encoded bits with error corrections. The Viterbi decoder is often used as the corresponding decoding block at receiver side for convolutional encoder with 1/2 coding rate at transmitter side. We use a Viterbi decoder at the receiver side for error rate checking; however, we won't explain the Viterbi method in detail since it is out of our research scope.

2.2.2 Modulation scheme

IEEE 802.11 standards use various modulation schemes ranging from BPSK, QPSK to 16QAM, 64QAM and 256QAM.

High rate modulation schemes are highly susceptible to noise, and SNR is the main metric for the quality of the channel condition. Transmission with higher order modulation schemes of higher data rates is dependent on a good channel condition; otherwise, the system has to use a modulation scheme with a lower data rate to achieve the same prescribed BER with the poorer channel condition. The modulated stream is measured in symbols, each of which contains a varying number of bits for different modulation schemes. In general, higher order modulation schemes contain more bits per symbol, which is also a measurement for data rate assuming a fixed value of symbol rate. The numerical indicator before each modulation scheme is the number of all possible symbols, which is also the number of all possible combinations *for* n binary data bits per symbol (2^n possible combinations), for this modulation scheme.

Both PSK and QAM can be represented in constellation diagrams, which display all possible symbols that can be achieved by the given modulation scheme with points indicating modulated amplitude and phase in the complex plane. As an example, the constellation diagrams for QPSK (2 bits/symbol) and rectangular 16QAM (4 bits/symbol) are shown in Figure 2.3.

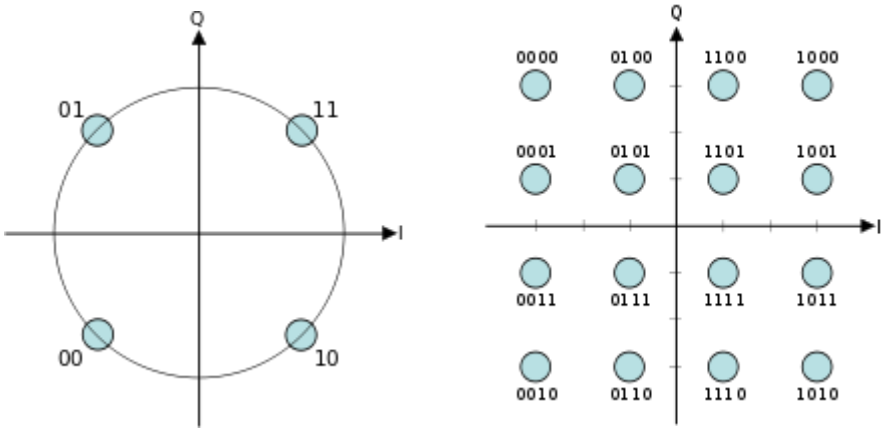

Figure 2.3 Constellation diagrams for QPSK and rectangular 16QAM

As shown by constellation diagrams in Figure 2.3, we observe that the distance between each symbol for 16QAM is shorter than the one for QPSK, which explains why

modulation schemes with lower data rates are more robust to noise. The bit stream can also be correctly decoded with a low SNR for a low order modulation scheme such as QPSK, compared to possible decoding errors caused by channel noise when using a high order modulation scheme such as 16QAM.

In addition, rectangular QAM schemes are applied in our research as shown by Figure 2.3. Also note that rectangular QAM is a sub-optimal solution for symbol point spacing in constellation diagrams for a given energy.

2.2.3 PHY layer IEEE 802.11ac standard

In this section, we focus on the physical layer specifications of IEEE 802.11ac standard. 802.11ac expects the maximum single-link throughput to be at least 500 Mbits/s and the minimum multi-antenna throughput using spatial multiplexing to be at least 1 Gbits/s, which requires the extension of 802.11n to air interfaces with higher-order MCS, wider radio frequency (RF) bandwidth and more MIMO spatial streams. As shown in the specifications of 802.11ac [52] [53], the standard can provide the maximum single-channel throughput of 866.7 Mbits/s with up to 4 spatial streams for each user (therefore, achieving a maximum of 3.47 Gbits/s for multi-antenna throughput for each user), which successfully meets the specifications of the next generation WLAN standard.

2.2.3.1 Higher-order MCSs

Firstly, higher-order MCSs (up to 256QAM with a coding rate of 5/6) are adopted in 802.11ac. 256QAM encodes 8 bits per symbol, which improves the highest data rate by nearly 1/3 with the same symbol rate compared to 64QAM (6 bits per symbol) which was adopted as the highest order modulation scheme in 802.11n. Compared to the MCSs adopted in 802.11a-g, we add two new MCSs with MCS index 8 and 9 which are detailed in next section.

Note that MCS 9 cannot be directly used sometimes since the number of convolutional encoded bits in a packet should be a multiple of 3. As a consequence, MCS 9 is not always available for all system settings. Theoretically, we can still achieve MCS 9 by

padding redundant bits to original data bits before convolutional coding. The details of obtaining the punctured convolutional encoded bits were introduced in section 2.2.1.2.

2.2.3.2 More RF bandwidth and OFDM

Next, more RF bandwidth is specified in 802.11ac as an option, and we introduce OFDM operation at the physical layer. OFDM is one of the most widely used multiplexing techniques in 4th and 5th generation wireless networks such as the LTE and 802.11 family. OFDM divides the bandwidth with a series of orthogonal subcarriers, within which the fading can be treated as frequency flat, to help combat the frequency selective fading. Compared to classical FDM with non-overlapping subcarriers, OFDM improves the spectral efficiency with more divided subcarriers.

For the 802.11 family, the minimum bandwidth of each channel is 20 MHz. 802.11ac also supports an option of 40 MHz, 80 MHz or 160 (or 80+80) MHz to meet the requirement of wider bandwidth to achieve the targeted throughput (Figure 2.4 gives the RF bandwidth extensions for 802.11ac compared to 802.11n). The whole operating bandwidth is divided into 13 overlapping channels at 2.4 GHz spectrum; roughly there are 185 overlapping channels at 5GHz spectrum. There are only 3 non-overlapping channels operating at 2.4 GHz spectrum, which is also one of the main reasons why 802.11ac operates only at 5 GHz spectrum since the bandwidth is not enough for even one 80 MHz or 160 MHz channel used by 802.11ac, while the 5 GHz spectrum provides more available bandwidth to meet the specifications of 802.11ac.

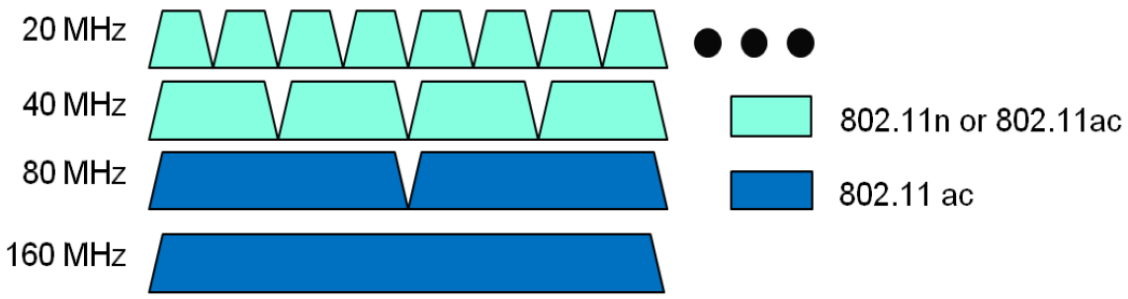

Figure 2.4 RF bandwidth supported by 802.11n and 802.11ac

In addition, the access point (AP) using 802.11ac supports up to 8 spatial streams for all accessed users and up to 4 spatial streams for each user. Table 2.2 [58] gives the OFDM parameter settings for all MCSs adopted by 802.11ac with 3 spatial streams for each user.

Table 2.2 OFDM parameter settings for 20 MHz bandwidth and 3 spatial streams

MCS Index	Modulation	Coding rate	N_{BPSC}	N_{SD}	N_{SP}	N_{CBPS}	N_{DBPS}	Data rate (Mbits/s)	
								800 ns GI	400 ns GI
0	BPSK	1/2	1	52	4	156	78	19.5	21.7
1	QPSK	1/2	2	52	4	312	156	39.0	43.3
2	QPSK	3/4	2	52	4	312	234	58.5	65.0
3	16QAM	1/2	4	52	4	624	312	78.0	86.7
4	16QAM	3/4	4	52	4	624	468	117.0	130.0
5	64QAM	2/3	6	52	4	936	624	156.0	173.3
6	64QAM	3/4	6	52	4	936	702	175.5	195.0
7	64QAM	5/6	6	52	4	936	780	195.0	216.7
8	256QAM	3/4	8	52	4	1248	936	234.0	260.0
9	256QAM	5/6	8	52	4	1248	1040	260.0	288.9

Here N_{BPSC} is the number of bits per subcarrier; N_{SD} is the number of data subcarriers; N_{SP} is the number of pilot subcarriers; N_{CBPS} is the number of convolutional coded bits per OFDM symbol; N_{DBPS} is the number of data bits per OFDM symbol; GI is the width of the guard interval for each OFDM symbol.

Note that the GI is inserted between successive OFDM symbols to combat the inter-symbol interference (ISI) caused by the multi-path effect. We can choose from 800 ns and 400 ns for different scenarios. Usually, 800 ns is chosen when the multi-path effect is severe; on the contrary, 400 ns is chosen to further increase data rates.

Also note that the 20 MHz channel was originally divided into a total of 52 subcarriers with 48 data subcarriers and 4 pilot subcarriers; however, the channel is divided into 56 subcarriers with 4 more data subcarriers after optimization. Since 802.11ac provides an

option of 20 MHz, 40 MHz, 80 MHz and 160 MHz channels, the number of data subcarriers for channels with different bandwidth are listed in Table 2.3 with 400 ns GI.

Table 2.3 Data subcarriers for different channel bandwidth with 400 ns GI

Channel bandwidth	20 MHz	40 MHz	80 MHz	160 MHz
No. of subcarriers	52	108	234	468

Next, we introduce the basic operations to obtain the OFDM symbols. For pass band notation, the transmitted OFDM symbol is shown in (2.2) for 52 data subcarriers.

$$s(t) = \sum_{k=-26,k\neq 0}^{26} m(k) \times e^{i2\pi f_k t}, \text{ (2.2)}$$

Where $m(k)$ is the modulated symbol; f_k is the transmitted frequency for subcarrier k with $f_k = f_c + k \times \Delta f$; f_c is the central frequency for the channel; Δf is the bandwidth spacing for every adjacent two subcarriers.

Therefore, for baseband notation, the transmitted OFDM symbol is shown by (2.3).

$$s(t) = \sum_{k=-26,k\neq 0}^{26} m(k) \times e^{i2\pi k \Delta f t}, \text{ (2.3)}$$

Assume $t = n \times T_s$ (where T_s is the sample rate for each OFDM symbol), then $\Delta f = \frac{B_c}{N} = 1/(N \times T_s)$ (where B_c is the channel bandwidth which equals 20 MHz). Then the baseband transmitted OFDM symbol can be converted as shown in (2.4).

$$s(n) = \sum_{k=-26,k\neq 0}^{26} m(k) \times e^{i2\pi kn/N}, \text{ (2.4)}$$

Equation (2.4) is actually an Inverse Discrete Fourier Transform (IDFT) for the modulated symbol $m(k)$. In order to use inverse Fast Fourier Transform (IFFT), we define *N*=64 to ensure that *N* is a power of 2 and N is larger than the number of data subcarriers. Thus, the OFDM operation is actually equivalent to taking a 64-point IFFT of the modulated symbols with bandwidth spacing for every two adjacent subcarriers $= \Delta f = \frac{B_c}{N} = 312.5\ kHz$.

Note that there are a total of *N*=64 sub-channels numbered from -32 to 31 for a 20 MHz channel, but only sub-channels of [-26, -1] and [1, 26] are used as data subcarriers. We can set the values of the other sub-channels to 0 to do 64-point IFFT operation. Thus, the baseband transmitted OFDM symbol can also be written as shown in (2.5).

$$s(n) = \sum_{k=-N/2}^{\frac{N}{2}-1} m(k) \times e^{i2\pi kn/N}, \text{ (2.5)}$$

2.2.3.3 More spatial streams

The specifications of 802.11ac extend 802.11n with more spatial streams. 802.11ac AP can serve up to 4 users with a total of 8 spatial streams, and each 802.11ac user supports up to 4 spatial streams. In addition, 802.11ac is the first Wi-Fi standard to use multi-user MIMO (MU-MIMO), which means an AP can simultaneously transmit data to multiple users with multiple antennas.

There are mainly two MIMO techniques: diversity and spatial multiplexing. The diversity technique transmits the same piece of data through multiple antennas to increase the reliability of data transmission, which is measured by the diversity order; the spatial multiplexing technique transmits multiple spatial streams with different data simultaneously to increase the overall system throughput. We focus on the spatial multiplexing technique because of the specifications of 802.11ac.

The first thing to consider with respect to spatial multiplexing technique is the receiver detecting technology. Without loss of generality, a zero-forcing detector, which is widely used in literature [63] [61] [65] [64] because of its low complexity. Zero-forcing detecting is a technique for spatial signal processing that separates multiple spatial streams at the receiver after the channel. More specifically, the received signal is processed by the pseudo-inverse operation to obtain the non-interfered symbol vector. Zero-forcing detecting is widely adopted by the 802.11 family such as 802.11n and 802.11ac.

Next, spatial correlation is another factor that affects the performance of MIMO. Ideally, the channel gain of each pair of transmit and receive antennas are assumed to be independent and identical distributed (i.i.d.); however, the antennas are often

correlated with each other in practice because of limitations on antenna spacing. The existence of correlation in real networks motivates the emergence of the proper spatial correlation model, such as the Kronecker model which is well-accepted and has been widely used in the literature [63] [KM,MV,YJ] [62] [60], to make the performance analysis more accurate and practical.

Note that both zero-forcing detector and spatial correlation model are detailed in Chapter 6 where cross-layer analysis on 802.11ac is extended.

2.2.4 Confidence interval

Confidence interval is a type of interval estimate for parameters in statistics. In this thesis, confidence interval is used for validating simulation results of performance analysis. The estimate is achieved by getting replications of simulation results, each of which is called a sample; the confidence interval is obtained by analyzing statistical behavior of these samples under the assumption that the variations of samples follow the normal distribution.

More specifically, let $\boldsymbol{S}$ denote the sample set of the measured results with a total of n samples. Then we can get the sample expectation μ and the standard deviation σ. Next, we obtain that the sample mean $\overline{\boldsymbol{S}}$ also follows the normal distribution with expectation μ and the standard deviation $\frac{\sigma}{\sqrt{n}}$. By transforming to the standard normal distribution, the standardized random variable $\boldsymbol{Z}$ is obtained by (2.6),

$$\boldsymbol{Z} = \frac{\overline{\boldsymbol{S}} - \mu}{\frac{\sigma}{\sqrt{n}}}, (2.6)$$

As a convention, we often choose 95% confidence interval for the interval estimate. Therefore, we have (2.7),

$$P(-z_{\alpha/2} \leq \boldsymbol{Z} \leq z_{\alpha/2}) = 1 - \alpha = 0.95, (2.7)$$

Where $z_{\alpha/2}$ follow the cumulative standard normal distribution function $\Phi(\cdot)$, thus, we obtain (2.8),

$$z_{\alpha/2} = \Phi^{-1}\left(\Phi\left(z_{\alpha/2}\right)\right) = \Phi^{-1}\left(1 - \frac{\alpha}{2}\right) \approx 1.96, (2.8)$$

Finally, we determine the 95% confidence interval with $[\overline{\boldsymbol{S}} - 1.96\frac{\sigma}{\sqrt{n}}, \overline{\boldsymbol{S}} + 1.96\frac{\sigma}{\sqrt{n}}]$.

Note that all the validations in this thesis use the confidence interval estimate method as a standard. In other words, we validate all simulation results with 95% confidence interval except specific explanations.

2.3 Contributions of this thesis

Based on the motivations discussed in Chapter 1 as well as the unfinished work on cross-layer design incorporating queueing effects stated in section 2.3, we generalise the contributions of this thesis accordingly, including the literature review for newly adopted techniques and models.

Contribution 1: The applications of bursty arrivals for cross-layer analysis (Chapter 5)

We apply ON-OFF and aggregated ON-OFF arrival processes as bursty traffic models at the data-link layer, in cross-layer design incorporating queueing effects. The performance analyses on queue state in the buffer, PDP due to buffer overflow, average queueing delay, PLP and average throughput are obtained by the matrix geometric method. The state transition matrix based on an embedded FSMC with a tuple of queue state, channel state and bursty state, is constructed for queueing analysis, which is the first time to provide bursty arrival cross-layer analysis. Finally, the G/D/s(t)/k queue model incorporating AMC is generalised for both Poisson and bursty arrivals. Numerical results show the importance of choosing proper arrival traffic models for practical performance analysis.

The ON-OFF arrival process is considered as one of the most suitable traffic processes to model bursty arrivals. The ON-OFF traffic model discussed in this thesis assumes constant arrival rate during ON periods and zero arrivals during OFF periods, and similar assumption can also be found in the literature such as [46] [47] [48] [49].

Furthermore, for users requiring multiple streams, which means multiple bursty packet streams arriving and getting served simultaneously, the classical single ON-OFF arrival process should be extended to aggregated ON-OFF arrival process ([44] [45]), in which each stream has an independent ON-OFF arrival process.

In fact, there is always debate on the proper model for bursty traffic. [36] and [48] adopted MMPP as bursty traffic model; while [46] [47] [49] supported ON-OFF process. As a matter of fact, the arrival rate for ON period can either be a constant or follow a Poisson distribution, which is then equivalent to an MMPP. This thesis is not intended to address this debate, but to choose the ON-OFF process with constant arrival rates during ON periods to ensure widely acceptance and limited complexity.

Contribution 2: The extension to IEEE 802.11ac standard (Chapter 6)

We apply the generalized G/D/s(t)/K queue model incorporating AMC, which is detailed in Chapter 5, into an analysis of the IEEE 802.11ac standard. In order to meet the specifications of 802.11ac with higher-order MCSs, more channel bandwidth and more spatial streams, we re-determine the SNR thresholds for AMC at the physical layer, applying with OFDM and interleaving blocks for data processing. In addition, we introduce spatial multiplexing MIMO system with a SU-MIMO model in which both AP and the user device are equipped with 4 antennas, and the proposed cross-layer analysis is ready to apply to any other antenna settings. Moreover, regarding antenna correlations, we set up two scenarios: one with transmit correlation only; the other with both transmit and receive correlations. We construct a probability transition matrix, which is based on a FSMC with slow fading for the service process, by analyzing LCRs for envelops of the instantaneous SNR. Finally, the cross-layer analysis is achieved with both Poisson and bursty arrivals, and the impact of antenna correlations is also investigated by numerical results.

For the part of re-determining SNR thresholds, we obtain the specifications of 802.11ac from [53] and [58], and we also get basic procedure for data processing from [55] [56] [57]. The PER analysis for each MCS adopted is detailed in Appendix A. It is the first time to obtain the statistics to determine SNR thresholds for 802.11ac.

For the part of multiplexing MIMO in presence of antenna correlations, [61] [62] [63] managed to get the closed form expression for SNR distribution of each spatial stream with ZF detector in presence of transmit correlation only. [63] ever tried to obtain the closed-form expression for symbol error rate (SER), which is restricted to a 2×2 MIMO, with both transmit and receive correlations. Therefore, it is quite hard, if not impossible,

to get the closed-form expression for SNR distribution in presence of both transmit and receive correlations. As a result, we use Monte-Carlo simulation instead to get the asymptotic expression for SNR distribution for each spatial stream.

For the part of constructing service rate transition matrix, we look for solution in terms of LCR for correlated channels. However, all of the related works found in the literature [66] [68] [69] [70] are for diversity MIMO especially for maximal-ratio combining (MRC) system. There's no related work for spatial multiplexing MIMO with ZF detector. Through simulation, we find that the operation taken by ZF detector has changed the LCRs for correlated channels. Thus, it is too hard to get the closed-form expression for LCR, even with transmit correlation only. Therefore, we use Monte-Carlo simulation instead to obtain LCRs for constructions of service rate transition matrices.

Chapter 3 Queue Model with Vacations

In this chapter, we validate and then analyse the basic queue models with and without vacations, in terms of average queue length, average waiting time and PDP. As specified in Chapter 1, the cross-layer design can be abstracted as an M/D/s(t)/K queue model with a range of working vacations with different service rates. Since we adopt a limited number of MCSs for the cross-layer design, the service rate change is actually a change in the number of servers. This chapter lays the foundation for Chapter 4 which achieves analysis of the cross-layer design. Based on the goal of achieving an M/D/s(t)/K queue model with a range of working vacations for Chapter 4, we build up and validate the queue model step by step:

1. Construct and validate an M/D/1 queue model without vacations and with infinite buffer

2. Construct and validate an M/D/1/K queue model without vacations

3. Construct and validate an M/D/1/K queue model with vacations (normally exponentially distributed)

3.1 Construct M/D/1 and M/D/1/K queue model without vacations

3.1.1 System model and queueing analysis

The task of this section is to build up basic M/D/1 and M/D/1/K queue models without vacations and obtain the stationary distribution of queue length, average packet delay and PDP due to buffer overflow. The queueing analysis is packet-based so that the queue model is actually focused on the data-link layer. The system diagram of M/D/1 and M/D/1/K queue models without vacations is shown in Figure 3.1. Note that the system is slot-based, which means the time is divided into a series of equal-length timeslot $\delta\tau$, and we make analysis and validations on a slot-by-slot basis.

The service rate is deterministic, i.e. a fixed value between inter-service times in any busy period, and with only one server, so we set the service rate=1 packet/slot without loss of generality.

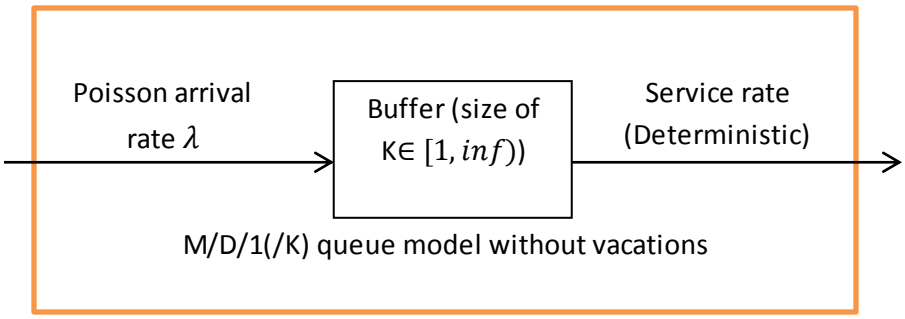

Figure 3.1 M/D/1(/K) queue model without vacations

The basic M/D/1 queue model without limitation as to the buffer length is equivalent to M/D/1/K queue model with infinite buffer, i.e. K=inf. Therefore, both M/D/1 and M/D/1/K can be generalised as one type of queue model.

The arrival process is a Poisson distributed number of arrivals per slot and the probability mass function (p.m.f.) for Poisson arrival process A_t which denotes the number of arrival packets within *t*-th timeslot is obtained by (3.1).

$$P(A_t = a) = \begin{cases} \frac{(\lambda)^a \times e^{(-\lambda)}}{a!}, a \in N, \\ 0, otherwise, \end{cases} \quad (3.1)$$

λ is the mean arrival rate for each timeslot of length δτ.

Let B_t denote the queue state at the end of t-th timeslot, S_t denotes the number of departing packets at the beginning of t-th timeslot and K denotes the buffer size. We use the matrix geometric method, which will always be used in this thesis for theoretical analysis. Recall the general procedure for matrix geometric method specified in Chapter 1, the probability transition matrix for queue state should be obtained at first.

The queue state B_t at the end of t-th timeslot after packet arrivals and departures is obtained by (3.2).

$$B_t = max\big(0, \min(K, (B_{t-1} + A_t - S_t))\big), \quad (3.2)$$

Then the probability transition matrix $\boldsymbol{P}^{M/D/1/K}$ for the process of queue state $\{B_t\}$ can be obtained in terms of the p.m.f. of A_t, S_t and buffer size K. By changing the value of

service rate, we can get different transition matrices. In general, the matrix $\boldsymbol{P}^{M/D/1/K}$ is shown by (3.3) with each element $p_{B_{t-1},B_t}^{M/D/1/K}$ (3.10).

$$\boldsymbol{P}^{M/D/1/K} = \begin{pmatrix} p_{0,0}^{M/D/1/K} & \cdots & p_{0,K}^{M/D/1/K} \\ \vdots & \ddots & \vdots \\ p_{K,0}^{M/D/1/K} & \cdots & p_{K,K}^{M/D/1/K} \end{pmatrix}, \quad (3.3)$$

And

$$p_{B_{t-1},B_t}^{M/D/1/K} = \begin{cases} P(A_t = B_t - B_{t-1} + S_t), if\ B_t \in [0,K) \\ 1 - \sum_{B_t \in [0,K)} P(A_t = B_t - B_{t-1} + S_t), if\ B_t = K \end{cases}, \quad (3.10)$$

Based on the probability transition matrix $\boldsymbol{P}^{M/D/1/K}$, the stationary distribution $\boldsymbol{\pi}^{M/D/1/K}$ for queue state in the buffer is obtained by solving (3.4).

$$\boldsymbol{\pi}^{M/D/1/K} = \boldsymbol{\pi}^{M/D/1/K} \times \boldsymbol{P}^{M/D/1/K}, \sum_{b \in [0,K]} \pi_b^{M/D/1/K} = 1, \quad (3.4)$$

Based on (3.2), we obtain the process for packet drop $\{D_t\}$ at the end of each timeslot δτ by (3.5).

$$D_t = \max(0, (A_t + B_{t-1} - S_t - K)), \quad (3.5)$$

Thus, we can get the average number of packets dropped $E(D)$ within each timeslot δτ by (3.6).

$$E(D) = \sum_{a=0}^{\infty} \sum_{B_{t-1}=0}^{K} D_t \times P(A_t = a) \times \pi_{B_{t-1}}^{M/D/1/K}, \quad (3.6)$$

$P(A_t = a)$ is the stationary distribution of packet arrivals specified in (3.1) and $\pi_{B_{t-1}}^{M/D/1/K}$ is the element of stationary distribution $\boldsymbol{\pi}^{M/D/1/K}$ for queue state in the buffer (3.4).

Then the PDP for M/D/1/K queue model is obtained by (3.7) based on (3.6).

$$PDP = \frac{Average\ number\ of\ packets\ dropped}{Average\ number\ of\ packets\ arrived} = \frac{E(D)}{\lambda}, \quad (3.7)$$

The average queue length $E(B)$ for M/D/1/K queue model is obtained by (3.8) from (3.4).

$$E(B) = \sum_{B=0}^{K} B \times \pi_B^{M/D/1/K}, \quad (3.8)$$

By Little's Law, we can obtain the average packet delay $E(W)$, which is the packet waiting time before entering service, by (3.9) from (3.8).

$$E(W) = E(B)/[\lambda \times (1 - PDP)], \quad (3.9)$$

The flow chart of getting queue state B_t and packet drop D_t for each timeslot δτ is shown in Figure 3.2.

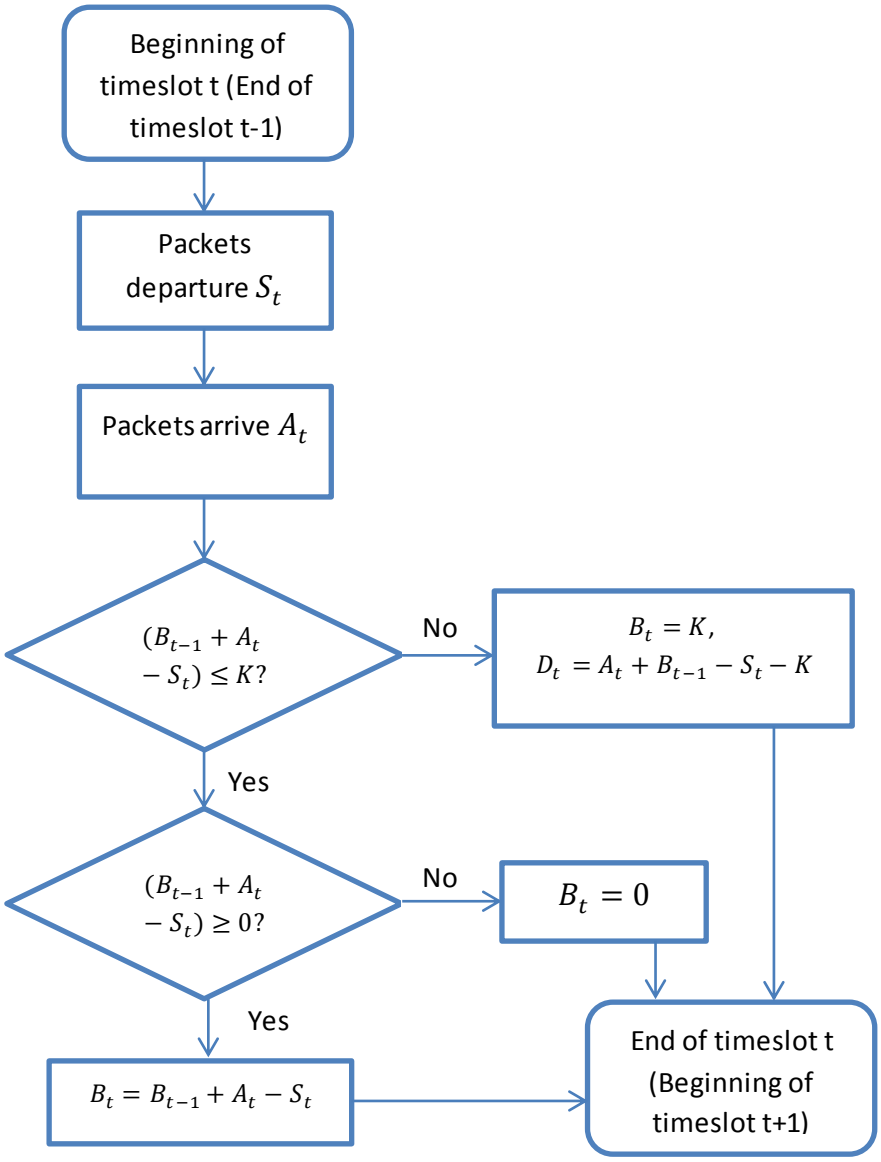

Figure 3.2 Flow chart for obtaining B_t and D_t for M/D/1/K queue model

In summary, equations (3.7)-(3.9) provide performance analysis for an M/D/1/K queue model without vacations in terms of PDP due to buffer overflow, average queue length in the buffer and average packet delay. We will validate all of them in section 3.1.2.

3.1.2 Numerical results

As detailed in 3.1.1, we use the matrix geometric method for queueing analysis. In this section, we validate the theoretical results obtained from 3.1.1 by making comparisons with Monte-Carlo simulations. The metrics we check are the stationary distribution of queue length in the buffer, PDP due to buffer overflow and average packet delay.

We specify a slot-based system at the data-link layer using an M/D/1(/K) queue model. The parameters setting are listed in Table 3.1.

Table 3.1 Parameters setting for M/D/1(/K) queue model at the data-link layer

Parameter	Value
Slot duration T_s	2 ms ([28])
Poisson arrival rate λ	[0.1, 0.3, 0.5, 0.7, 0.9] packet per slot
Service rate μ	1 per slot (500 packets/s)
System load ρ	[0.1, 0.3, 0.5, 0.7, 0.9]

Note that the system load $\rho = \lambda/\mu$ is always less than 1; otherwise the queue length will grow to infinity. The slot duration is set to 2 ms for unification with parameters adopted in cross-layer design, which is specified in Chapter 4.

In addition, we need to make a comparison between an M/D/1 queue model with infinite buffer and an M/D/1/K queue model with a finite buffer length K in order to show the impact of buffer size K on performance at the data-link layer.

3.1.2.1 Comparisons of queue length in the buffer

Firstly, we validate and check the probability mass function for the queue length in the buffer for M/D/1(/K) queue models. We set buffer size K=50 ([28]) for the M/D/1/K queue model so that we only need to check the corresponding queue length up to 50

for the M/D/1 queue model. We make a comparison with a range of system loads (0.1, 0.3, 0.5, 0.7 and 0.9) to check the impact of system loads on queue length in the buffer. Figure 3.3 and 3.4 are stationary probability mass functions for the queue length in the buffer for M/D/1 and M/D/1/K queue model respectively.

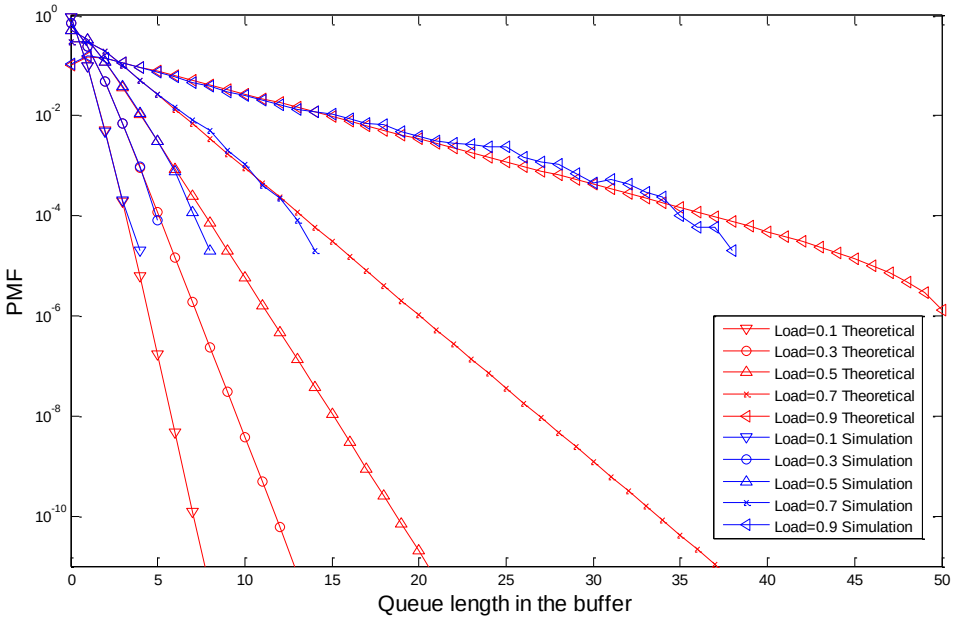

Figure 3.3 Stationary probability mass function of queue length for M/D/1 queue

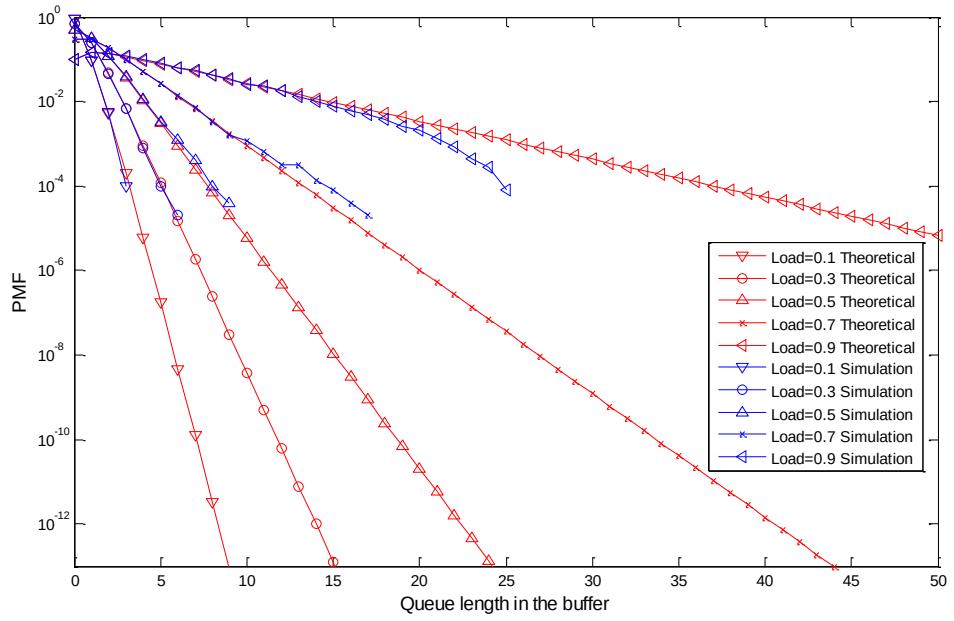

Figure 3.4 Stationary probability mass function of queue length for M/D/1/K queue

In general, we obtain from Figure 3.3 and 3.4 that the lower the load, the lower the probability for the buffer reaching larger queue lengths. The gaps between each line with different system loads become more obvious (often a gap of 2 orders of magnitude when the probability is around 10^{-12}) as the queue length grows.

We can also get the differences between M/D/1 and M/D/1/K queue models from Figure 3.3 and 3.4. Generally, the M/D/1/K queue model achieves comparatively higher (e.g. a gap of nearly 1 order of magnitude when the system load is 0.9) probabilities for queue length at the tail than M/D/1 queue model under all ranges of system loads, which is caused by the finite buffer size K in M/D/1/K queue model.

3.1.2.2 PDP due to buffer overflow

Next, we validate and check PDP due to buffer overflow for the M/D/1/K queue model. Based on the validation of M/D/1/K queue state and equations (3.5)-(3.7), we obtain PDP due to buffer overflow with different loads and different buffer sizes. The matrix analysis result is shown in Figure 3.5 in comparison with simulation results.

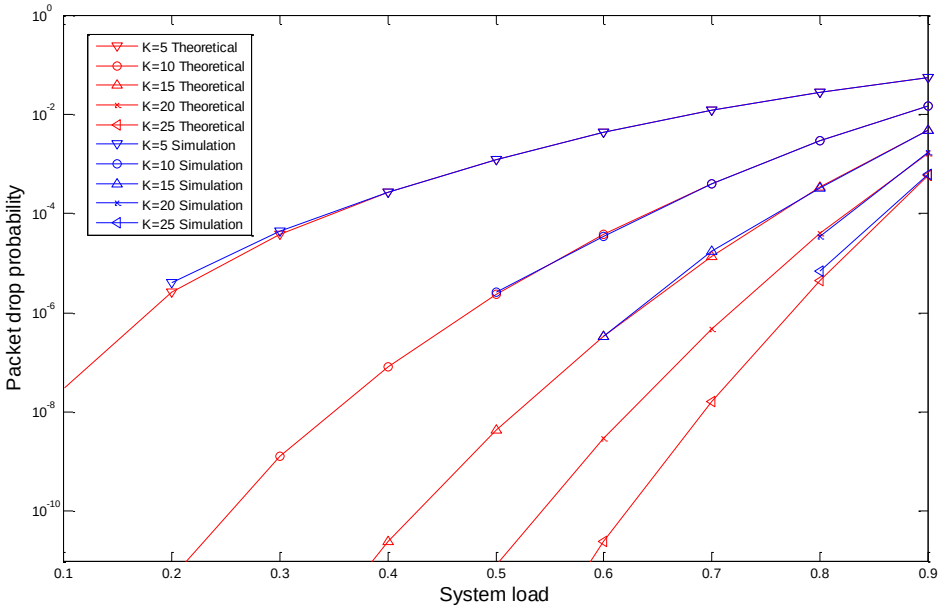

Figure 3.5 PDP with different loads and buffer sizes for M/D/1/K

Note that due to well-understood time constraints we have not been able to simulate probability values lower than about 10^{-6}. From Figure 3.5, we obtain that the PDP for the M/D/1/K model is only influenced by two parameters: system load ρ and buffer size

K. For the same system load ρ, PDP decreases as the buffer size K increases; while maintaining the same buffer size K, PDP increases as the system load ρ increases.

3.1.2.3 Average packet delay

In this section, we validate and check average queueing delay for the M/D/1/K queue model. Based on equations (3.8)-(3.9), we obtain average queueing delay with different loads and different buffer sizes. The analysis results, together with simulation results, are shown in Figure 3.6.

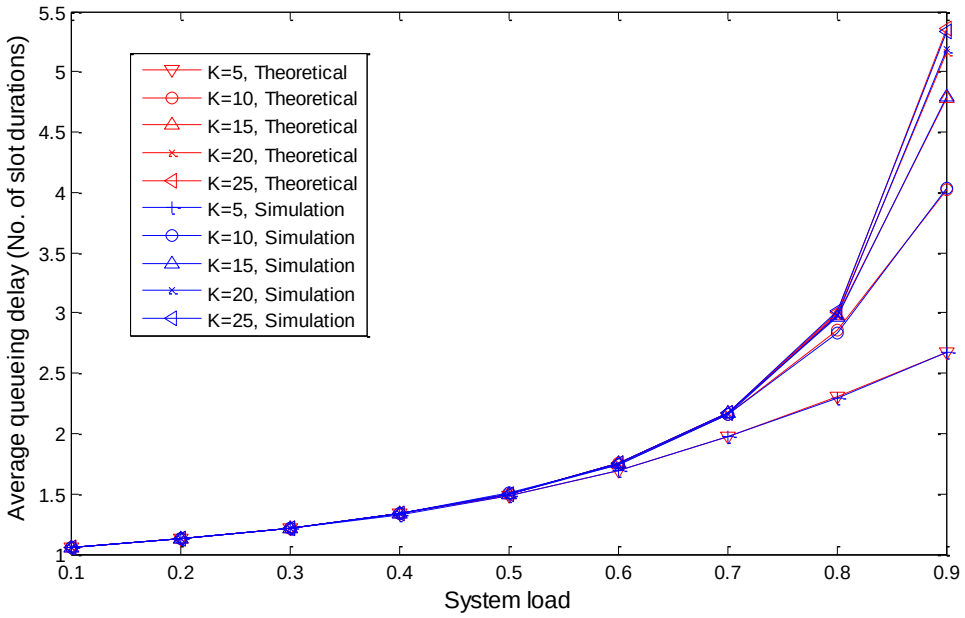

Figure 3.6 Average queueing delays with different loads and buffer sizes for M/D/1/K

From Figure 3.6, we get that the average queueing delay is only influenced by system load ρ and buffer size K. For the same system load ρ, average queueing delay decreases as the buffer size K increases; while maintaining the same buffer size K, average queueing delay increases as the system load ρ increases.

3.1.3 Summary

The queueing analysis is for M/D/1 and M/D/1/K queue models without vacations. We use the matrix geometric method for theoretical analysis and validate the results by comparison with Monte-Carlo simulations in terms of queue length in the buffer, PDP due to buffer overflow and average packet delay. This section also provided the general

procedures of obtaining metrics using the matrix geometric method. The results reveal the impact of buffer size K on basic M/D/1 model and lay the foundation for analysing the M/D/1/K queue model with vacations.

3.2 Constructing the M/D/1/K queue model with vacations

3.2.1 System model and queueing analysis

The task of this section is to build up an M/D/1/K working vacations model. As suggested in Chapter 1, working vacation queue model is the special case of vacation model, thus, we only check vacation queue model for simplicity. Similar to the method used in section 3.1, we obtain the queue length in the buffer by the matrix geometric method; then we get PDP and average packet delay based on the queue length in the buffer. Simulations and validations are done for M/D/1/K vacation model, and further comparisons with the M/D/1/K model without vacations are observed and analysed.

The M/D/1/K vacation model is a little more complicated to analyse because we must also consider possible service rate variations between adjacent timeslots. For a normal M/D/1/K vacation model, we often assume there are two states for the server: non-vacation state and vacation state. For non-vacation periods, the server works as in classical M/D/1/K queue model. While for vacation periods, the service rate is zero. Therefore, a two-state Markov chain for service rates is constructed to indicate service rate transition probabilities between adjacent timeslots shown in Figure 3.7. The probability transition matrix for service rate change $\boldsymbol{P}^{sr\_change}$ is obtained by (3.11).

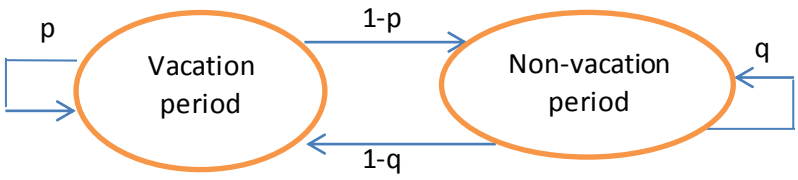

Figure 3.7 General Markov chain for service rate change in a simple vacation model

$$\boldsymbol{P}^{sr\_change} = \begin{pmatrix} p & 1-p \\ 1-q & q \end{pmatrix}, \quad (3.11)$$

Based on the property of Markov chains, both vacation and non-vacation periods are exponentially distributed [78]. Note that the vacation model we adopt here is different from certain server vacation models proposed in the literature, such as [1]. Traditional server vacation models transfer into a vacation period when the buffer goes empty; it changes back to a non-vacation period after (usually) an exponentially distributed period of time when the queue length in the buffer hits a predefined threshold. However, our vacation model is driven by a probability transition matrix which is independent of the queue state. Our adopted vacation model is suitable for extensions to cross-layer design with AMC which is detailed in Chapter 4.

We first obtain the probability transition matrix for the M/D/1/K vacation model. The system operates on a slot-by-slot basis. Similar to the flow chart for analysing M/D/1/k model without vacations specified in Figure 3.2, we can obtain the queue state process $\{B_t\}$ slot by slot. Note that the values for service process $\{S_t\}$ are no longer fixed to 1 packet/slot but have to choose from [0, 1] based on the probability transition matrix for service rate change $\boldsymbol{P}^{sr\_change}$ obtained by (3.11). Therefore, the probability transition matrix for state change for M/D/1/K vacation model should consider not only queue state B_t but also service rate state S_t. Thus, the transitions for both B_t and S_t form an embedded FSMC with each element (B_{t-1}, S_t) where all elements are accessible from each other.

Based on the embedded Markov chain (B_{t-1}, S_t) , we can obtain the probability transition matrix $\boldsymbol{P}^{MD1K\_vac}$ (3.12) for (B_{t-1}, S_t) with each element $\boldsymbol{A}_{B_{t-1},B_t}^{MD1K\_vac}$ (3.13).

$$\boldsymbol{P}^{MD1K\_vac} = \begin{pmatrix} \boldsymbol{A}_{0,0}^{MD1K\_vac} & \cdots & \boldsymbol{A}_{0,K}^{MD1K\_vac} \\ \vdots & \ddots & \vdots \\ \boldsymbol{A}_{K,0}^{MD1K\_vac} & \cdots & \boldsymbol{A}_{K,K}^{MD1K\_vac} \end{pmatrix}, \quad (3.12)$$

$$\boldsymbol{A}_{B_{t-1},B_t}^{MD1K\_vac} = \begin{pmatrix} p_{(B_{t-1},0),(B_t,0)}^{MD1K\_vac} & p_{(B_{t-1},0),(B_t,1)}^{MD1K\_vac} \\ p_{(B_{t-1},1),(B_t,0)}^{MD1K\_vac} & p_{(B_{t-1},1),(B_t,1)}^{MD1K\_vac} \end{pmatrix}, \quad (3.13)$$

The elements $p^{MD1K\_vac}_{(B_{t-1},S_t),(B_t,S_{t+1})}$ of matrix $\boldsymbol{P}^{MD1K\_vac}$ satisfy (3.14).

$$p^{MD1K\_vac}_{(B_{t-1},S_t),(B_t,S_{t+1})} =$$
$$\begin{cases} p^{sr\_change}_{S_t,S_{t+1}} \times P(A_t = B_t - B_{t-1} + S_t), if\ B_t \in [0,K) \\ p^{sr\_change}_{S_t,S_{t+1}} - \sum_{B_t \in [0,K)} p^{sr\_change}_{S_t,S_{t+1}} \times P(A_t = B_t - B_{t-1} + S_t), if\ B_t = K \end{cases}, \quad (3.14)$$

$p^{sr\_change}_{S_t,S_{t+1}}$ is the element of service rate transition matrix $\boldsymbol{P}^{sr\_change}$ obtained by (3.11); $P(A_t = a)$ is the Poisson distribution obtained by (3.1).

Then we can get the stationary distribution $\boldsymbol{\pi}^{MD1K\_vac}$ for queue state and service state for the M/D/1/K vacation model (3.15).

$$\boldsymbol{\pi}^{MD1K\_vac} = \boldsymbol{\pi}^{MD1K\_vac} \times \boldsymbol{P}^{MD1K\_vac}, \quad \sum_{\substack{B\epsilon[0,K] \\ S\epsilon[s_0,s_1]}} \boldsymbol{\pi}^{MD1K\_vac}_{(B,S)} = 1, \quad (3.15)$$

Thus, we can obtain the average number of packets dropped $E(D)$ within each timeslot δτ by (3.16).

$$E(D) = \sum_{a=0}^{\infty} \sum_{B_{t-1}=0}^{K} \sum_{S_t=0}^{1} D_t \times P(A_t = a) \times \pi^{sr\_change}_{S_t} \times \pi^{MD1K\_vac}_{(B_{t-1},S_t)}, \quad (3.16)$$

D_t is obtained by (3.5); $P(A_t = a)$ is obtained by (3.1); $\pi^{MD1K\_vac}_{(B_{t-1},S_t)}$ is obtained by (3.15); $\pi^{sr\_change}_{S_t}$, which is the stationary probability distribution for service rates, is obtained by (3.17).

$$\boldsymbol{\pi}^{sr\_change} = \boldsymbol{\pi}^{sr\_change} \times \boldsymbol{P}^{sr\_change}, \quad \sum_{S\epsilon[s_0,s_1]} \boldsymbol{\pi}^{sr\_change}_{S} = 1, \quad (3.17)$$

Similar to the method specified in 3.1.1, we can obtain the PDP due to buffer overflow, and the average packet delay by (3.7)-(3.9) for the M/D/1/K vacation model.

3.2.2 Numerical results

We validate the theoretical results specified in 3.2.1 with Monte-Carlo simulations. In addition, we check the similarities and differences between M/D/1/K queue models with and without vacations in terms of queue length in the buffer, PDP due to buffer overflow and average queue length, in order crucially to find the impact of vacations on performance (this aspect is vital, as in the future the 'vacations' represent the behaviour of the fading channel).

3.2.2.1 Comparisons of queue length in the buffer

Firstly, we validate queue length in the buffer for M/D/1/K queue models with and without vacations. We set buffer size K=50 (the same setting as in [28]). To check the similarities and differences between M/D/1/K queue models with and without vacations, we need to adjust the probability transition matrix for service rate change $\boldsymbol{P}^{sr\_change}$ to different values as well as keeping system loads the same.

In order to keep the same system loads for M/D/1/K model with and without vacations, we need to adjust the Poisson arrival rates for the two models. Since the service rate for M/D/1/K queue model without vacations is 1 packet/slot and service rate for M/D/1/K vacation model is 0 or 1 packet/slot with probability distribution of $\boldsymbol{\pi}^{sr\_change}$, we can obtain the relationship of Poisson arrival rates for two models by (3.18).

$$\lambda^{MD1K} = \lambda^{MD1K\_vac} / \pi_1^{sr\_change}, \text{ (3.18)}$$

Next, we need to adjust service rate transition matrix $\boldsymbol{P}^{sr\_change}$ to different values for comparisons. We've defined different settings of values for $\boldsymbol{P}^{sr\_change}$, λ^{MD1K_vac} and λ^{MD1K}:

1. When non-vacation periods dominate: state 0 for vacation period and state 1 for non-vacation period. The probability transition matrix for service change is obtained by (3.19) as an example.

$$\boldsymbol{P}^{sr\_change} = \begin{pmatrix} 0.1 & 0.9 \\ 0.1 & 0.9 \end{pmatrix}, \text{ (3.19)}$$

That is $\pi_0^{sr\_change} = 0.1$ and $\pi_1^{sr\_change} = 0.9$. The corresponding Markov chain is shown in Figure 3.8.

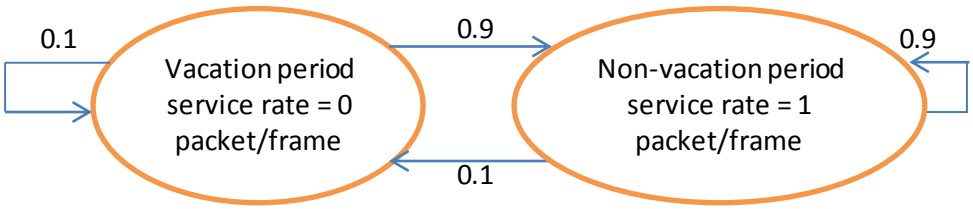

Figure 3.8 Service rate transition Markov chain for M/D/1/K vacation model by setting 1

We set λ^{MD1K_vac} =0.81, the equivalent classical M/D/1/K queue model without vacations $\lambda^{MD1K} = \lambda^{MD1K\_vac}/\pi_1^{sr\_change}$=0.9.

2. When equivalent λ^{MD1K} for M/D/1/K model without vacations is not very high maintaining (3.19). For example, $\lambda^{MD1K} = 0.7$. Then $\lambda^{MD1K\_vac} = \lambda^{MD1K} \times \pi_1^{sr\_change}$=0.63.

3. (a) When non-vacation periods are not much longer than vacation periods in M/D/1/K vacation model. The probability transition matrix for service change is obtained by (3.20) as an example.

$$\boldsymbol{P}^{sr\_change} = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}, \text{ (3.20)}$$

That is $\pi_0^{sr\_change} = 0.5 \; and \; \pi_1^{sr\_change} = 0.5$. The corresponding Markov chain is shown in Figure 3.9.

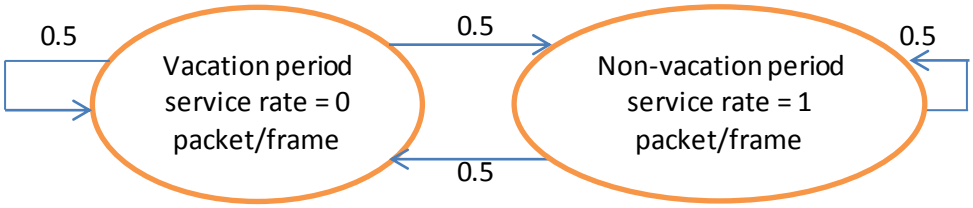

Figure 3.9 Service rate transition Markov chain for M/D/1/K vacation model by setting 3(a)

We set λ^{MD1K_vac} =0.45, the equivalent classical M/D/1/K queue model without vacations $\lambda^{MD1K} = \lambda^{MD1K\_vac}/\pi_1^{sr\_change}$ =0.9 by (3.20).

(b) Then we change (3.20) to (3.21).

$$\boldsymbol{P}^{sr\_change} = \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix}, \text{ (3.21)}$$

That is $\pi_0^{sr\_change} = 0.5 \; and \; \pi_1^{sr\_change} = 0.5$. The corresponding Markov chain is shown in Figure 3.10.

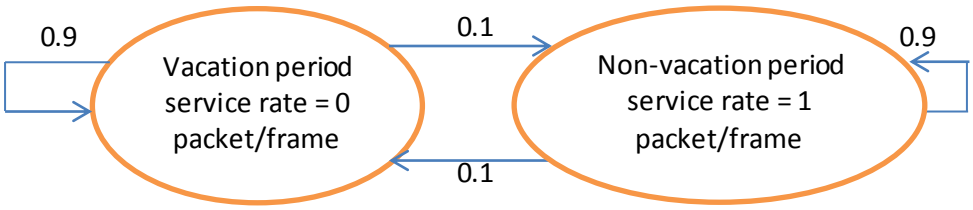

Figure 3.10 Service rate transition Markov chain for M/D/1/K vacation model by setting 3(b)

And we set λ^{MD1K_vac} =0.45, the equivalent classical M/D/1/K queue model without vacations $\lambda^{MD1K} = \lambda^{MD1K\_vac} / \pi_1^{sr\_change}$ =0.9 by (3.21).

Based on different settings for $\boldsymbol{P}^{sr\_change}$, λ^{MD1K_vac} and λ^{MD1K}, we specify a slot-based system for M/D/1/K queue model with and without vacations. The parameter settings are listed in Table 3.2.

Table 3.2 Parameter settings for M/D/1/K queue model with and without vacations

	M/D/1/K vacation model	M/D/1/K model no vacations
Slot duration T_s	2ms	2ms
Poisson arrival rate λ	[0.45, 0.63, 0.81] packet/slot	[0.7, 0.9] packet/slot
Service rate μ	[0, 1] packet/slot	1 packet/slot
System load ρ	[0.7, 0.9]	[0.7, 0.9]
Buffer size K	50	50

Figure 3.11-3.14 are queue state comparisons between M/D/1/K model with and without vacations obtained by different settings for $\boldsymbol{P}^{sr\_change}$, λ^{MD1K_vac} and λ^{MD1K}. All theoretical results for M/D/1/K vacation models are achieved by matrix geometric analysis and are validated with Monte-Carlo simulations. For comparison, queue state analysis for M/D/1/K model without vacations under same system load is also obtained.

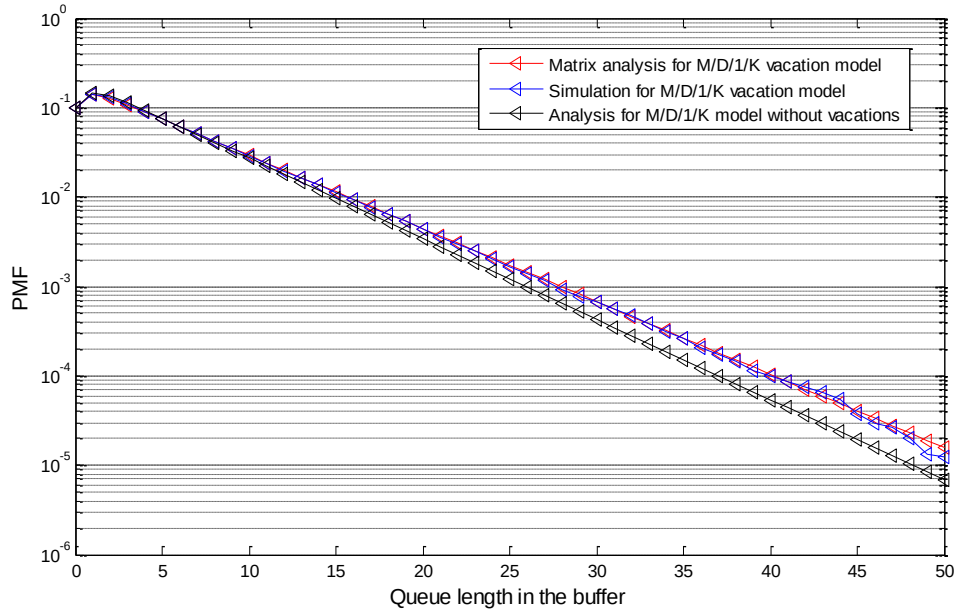

Figure 3.11 Queue state comparisons for M/D/1/K model given by setting 1

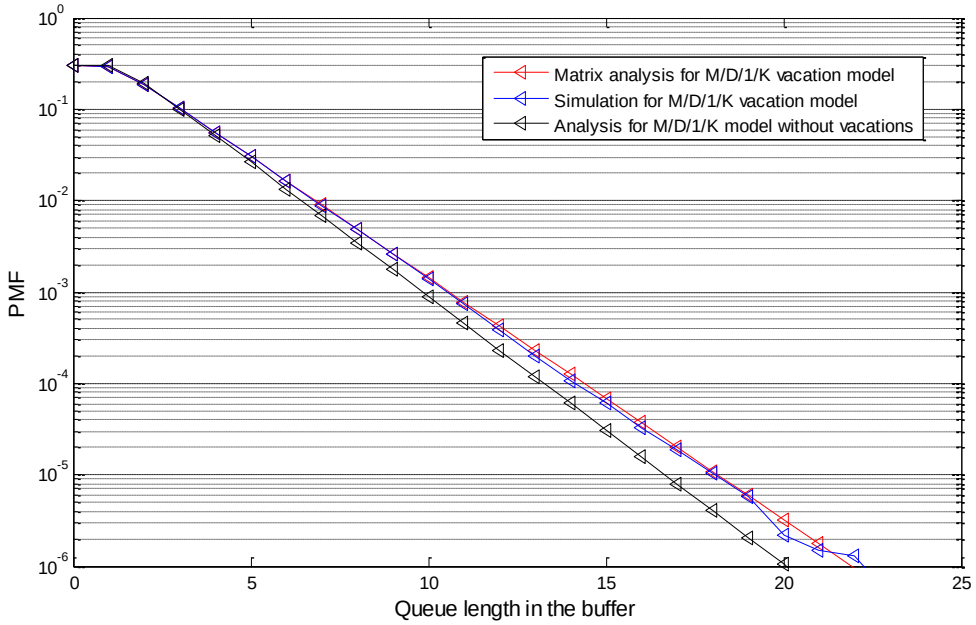

Figure 3.12 Queue state comparisons for M/D/1/K model given by setting 2

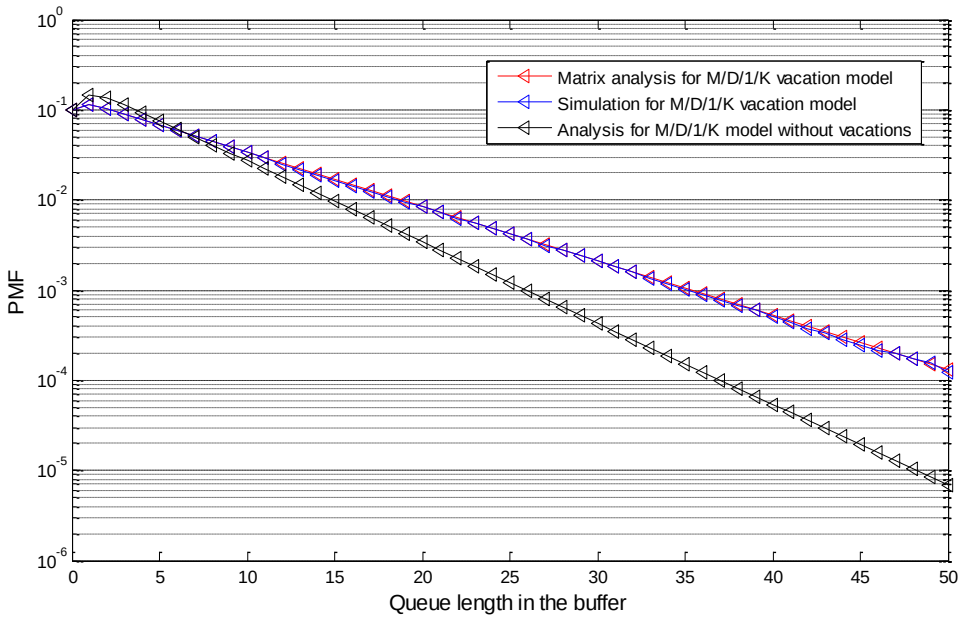

Figure 3.13 Queue state comparisons for M/D/1/K model given by setting 3 (a)

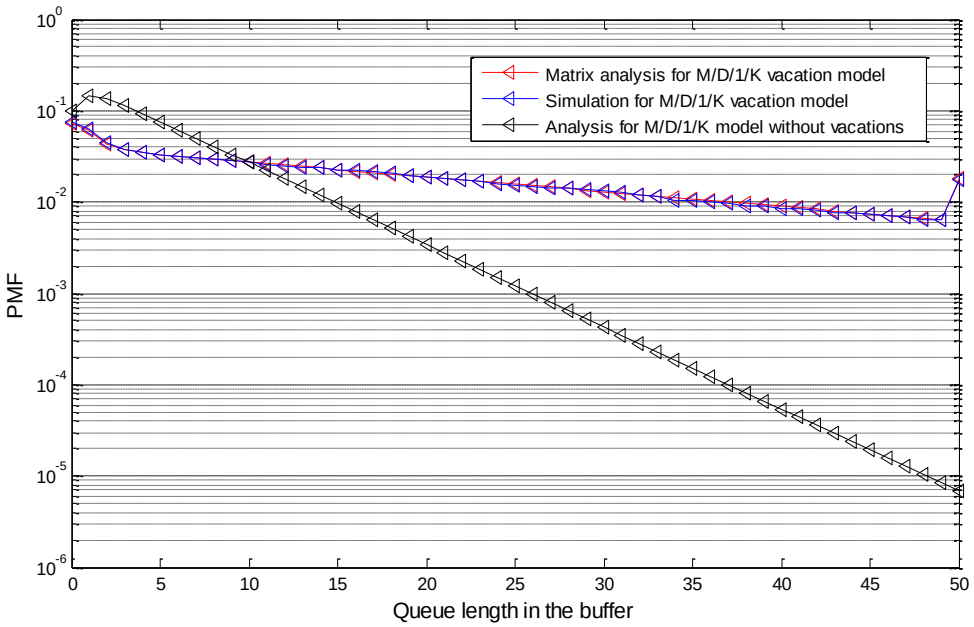

Figure 3.14 Queue state comparisons for M/D/1/K model given by setting 3 (b)

From Figure 3.11-3.14, we have following observations:

1. For each figure, there is a gap between M/D/1/K queue model with and without vacations under the same system load. M/D/1/K vacation models always achieve higher probabilities for queue length at the tail, which are influenced by vacation periods. Intuitively, during vacation periods, arriving packets fill the buffer more easily, causing higher probabilities for the buffer to stay at longer queue lengths.

2. Comparison between Figure 3.11 and Figure 3.12 illustrates an example of the impact of system load on queue models. Recall that we use λ^{MD1K}=0.9 in Figure 3.11 and λ^{MD1K}=0.7 in Figure 3.12. When system load gets lower, the probabilities for the buffer to stay at shorter queue lengths get higher, and the probabilities for the buffer to stay at longer queue lengths get lower.

3. Comparison between Figure 3.11 and Figure 3.14 illustrates an example of the impact of the stationary probability distribution for service rates $\boldsymbol{\pi}_S^{sr\_change}$ on queue models under the same system load. When the stationary probability of the server to stay at non-vacation period $\pi_1^{sr\_change}$ gets lower (from 0.9 to 0.5 in our case), arriving packets fill the buffer more easily with comparatively longer vacation periods, causing higher probabilities for buffer to stay at longer queue lengths.

4. Comparison between Figure 3.13 and Figure 3.14 illustrates an example of the impact of service rate transition matrix $\boldsymbol{P}^{sr\_change}$ on queue models under the same system load and the same stationary probability distribution for service rates $\boldsymbol{\pi}_S^{sr\_change}$. Recall equation (3.11). When p and q specified in (3.11) get higher (from 0.5 to 0.9 in our case), which means the transition between vacation and non-vacation periods become less frequent, arriving packets fill the buffer more easily during each comparatively longer individual vacation period, causing higher probabilities for buffer to stay at longer queue lengths.

In summary, we observe and analyse the comparisons for M/D/1/K queue model with and without vacations under different parameter settings in terms of queue length in the buffer, concluding that the queue model is highly influenced by system load ρ,

stationary probability distribution for service rates $\boldsymbol{\pi}_S^{sr\_change}$ and service rate transition matrix $\boldsymbol{P}^{sr\_change}$.

3.2.2.2 Comparisons of PDP due to buffer overflow

Based on the validations of queue length in the buffer for M/D/1/K vacation queue model with various service rate transition matrices $\boldsymbol{P}^{sr\_change}$, we check PDP due to buffer overflow with the same service rate transition matrices $\boldsymbol{P}^{sr\_change}$ specified by equations (3.19)-(3.21) under different system loads ρ ranging from 0.1 to 0.9 and different buffer size K ranging from 5 to 25. Note that there would be too many curves if we show all results within one diagram; in addition, there exists some common properties for M/D/1/K vacation models compared to M/D/1/K queue models without vacations. Therefore, we only analyse two representative figures including: 1. Comparisons between M/D/1/K queue models with and without vacations with one definite service rate transition matrix $\boldsymbol{P}^{sr\_change}$ specified by equation (3.19) to show the common properties for the M/D/1/K vacation models (Figure 3.15), and 2. Comparisons between M/D/1/K vacation models with one definite buffer size K=25 to show the differences in PDP due to buffer overflow with different service rate transition matrix $\boldsymbol{P}^{sr\_change}$ specified by equations (3.19)-(3.21) (Figure 3.16).

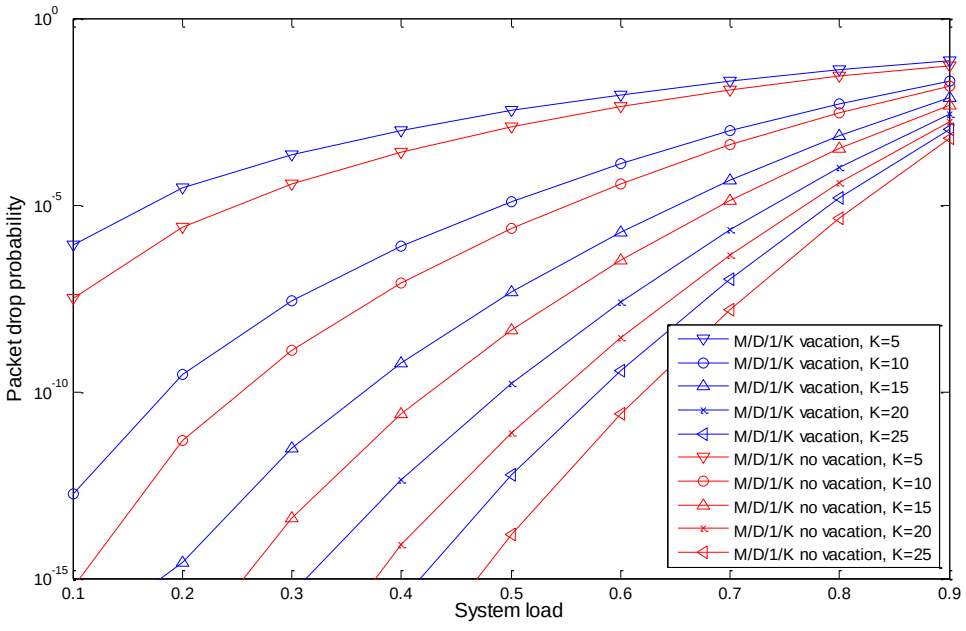

Figure 3.15 PDP comparisons for M/D/1/K model given by setting 1

From Figure 3.15, we have following observations:

1. Same as PDP behaviour for M/D/1/K queue model without vacations, the gaps between PDPs for M/D/1/K vacation model with different buffer sizes become more obvious (over 1 order of magnitude when the system load is 0.1) when the system load decreases.

2. Compared to M/D/1/K queue model without vacations, M/D/1/K vacation models always achieve higher PDPs with the same system loads and buffer sizes, which can be understood because of the impact of vacations although they only take up approximately 10% of the total time. The queue is much easier filled causing buffer overflow.

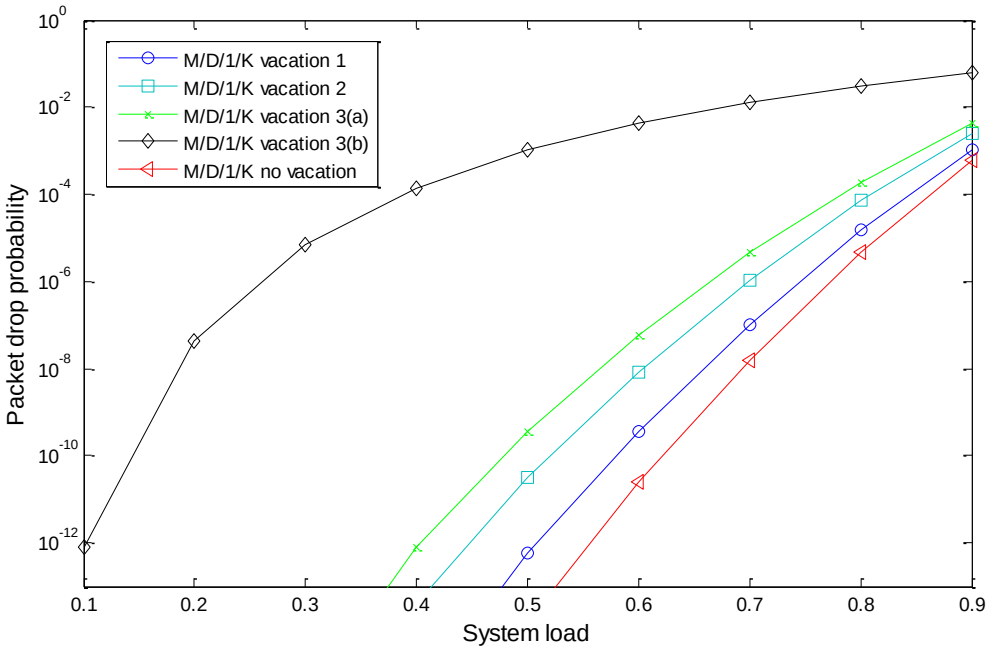

Figure 3.16 PDP comparisons between M/D/1/K vacation models when K=25

From Figure 3.16, we have following observations:

1. For the memory-less busy and vacation periods, which means whether the arrival traffic is in busy or vacation period for each slot is independent of former states (including settings 1, 2, 3 (a)), PDPs are comparatively higher for vacation models with lower probabilities to stay at busy periods, namely, lower values of q in (3.11). The situation can be understood because the lower the values of q, the longer the time the

vacations take. The impact of longer vacations often results in higher PDP, which has already been shown by Figure 3.15.

2. For the comparison between the curves for setting 3 (a) and 3 (b) with the same stationary probability distribution for service rate $\boldsymbol{\pi}_S^{sr\_change}$, the differences on PDPs are caused by the values of service rate transition matrix $\boldsymbol{P}^{sr\_change}$. For setting 3 (b), busy and vacation periods often last longer than ones for setting 3 (a) (namely, less transitions). As a result, higher PDPs are achieved for longer busy periods. In addition, compared to the memory-less property for settings 1, 2 and 3 (a), setting 3 (b) fundamentally approximates the burstiness for the arrival traffic, which lays the foundation of Chapter 5.

3.2.2.3 Comparisons of average queue delay

Same as the analysis process for PDP, we check average queueing delay by equations (3.8)-(3.9) from two aspects: 1. Comparisons between M/D/1/K queue models with and without vacations to show the common properties for the M/D/1/K vacation models (Figure 3.17), and 2. Comparisons between M/D/1/K vacation models to show the differences in average queueing delays with different service rate transition matrix $\boldsymbol{P}^{sr\_change}$ (Figure 3.18). Note that average queueing delay is measured by number of slot durations.

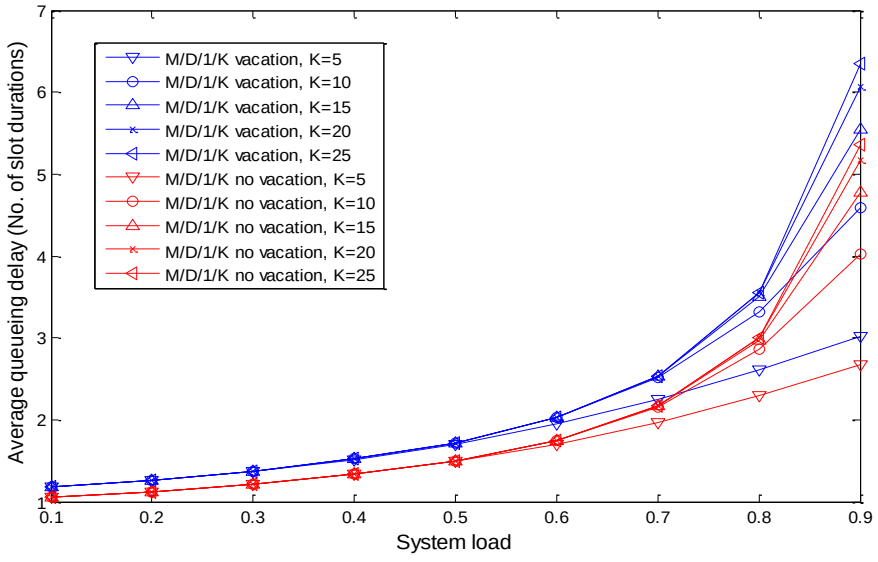

Figure 3.17 Delay comparisons for M/D/1/K model given by setting 1

From Figure 3.17, we have following observations:

1. For comparisons of M/D/1/K vacation models with different buffer sizes, the average queueing delay increases as the system load increases; and the differences become more obvious (nearly 0.4 in number of slot for K=5; while nearly 1 for K=25 when the system load=0.9) when the system load get larger. This property follows the same pattern as for M/D/1/K queue models without vacations.

2. For the comparisons of M/D/1/K queue models with and without vacations, M/D/1/K vacation models achieve higher average queueing delays compared to M/D/1/K queue model without vacations, which is caused by longer vacation times. During busy periods, the queue length in the buffer is often larger than in the equivalent queue model without vacations. As a result, the packets arrived during busy periods often take longer time to get served.

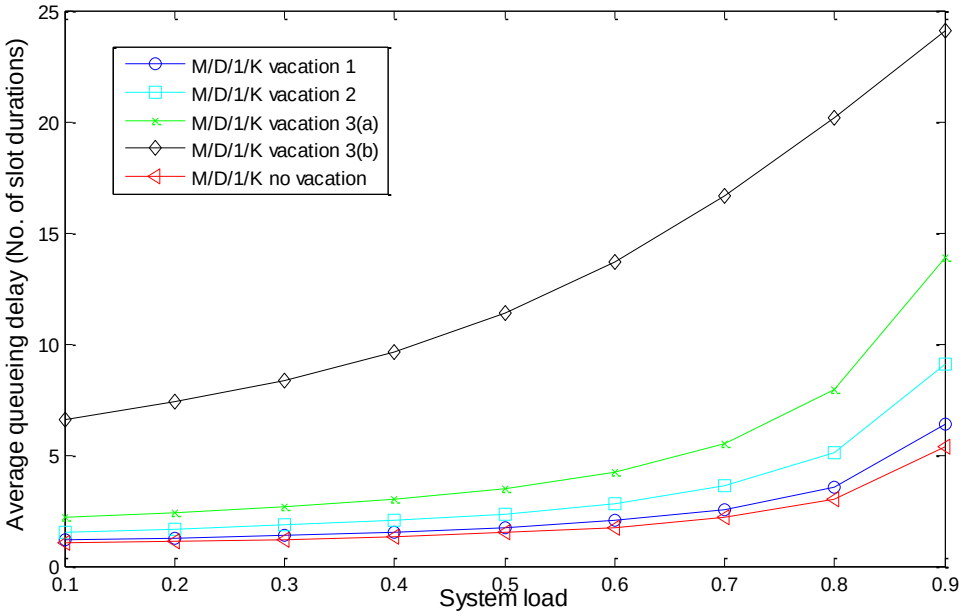

Figure 3.18 Delay comparisons between M/D/1/K vacation models when K=25

From Figure 3.18, we have following observations:

1. For comparisons between M/D/1/K vacation models with settings 1, 2 and 3 (a), which is summarized as memory-less arrival rate settings, average queueing delay gets

larger when the probability to stay at busy periods q in (3.11) gets smaller. The situation is caused by longer vacation times.

2. For comparisons between curves using setting 3 (a) and 3 (b), the differences on average queueing delays are caused by the values of service rate transition matrix $\boldsymbol{P}^{sr\_change}$. Longer average queueing delays are caused by on average longer busy and vacation periods.

3.2.3 Summary

We present a queueing analysis for M/D/1/K queue models with and without vacations. We use matrix geometric method for theoretical analysis and validate the results with Monte-Carlo simulations in terms of queue length in the buffer, PDP due to buffer overflow and average packet delay. This section also lays the foundation for queueing analysis of cross-layer design incorporating with AMC.

The simulation results are obtained for comparisons and reveal the impact of system load ρ, stationary probability distribution for service rates $\boldsymbol{\pi}_S^{sr\_change}$ and service rate transition matrix $\boldsymbol{P}^{sr\_change}$ on queue models.

Chapter 4 Cross-layer Design with AMC

In this chapter, we validate and analyse the cross-layer design incorporating AMC. We investigate how to combine packet-based queueing behaviour at the data-link layer and AMC at the physical layer to achieve a cross-layer design. As a result, an M/D/s(t)/K queue model with a full range of working vacations, representative of wireless networking, is obtained for queueing analysis on cross-layer design. The framework of queueing analysis on cross-layer design, including analysis on queue length in the buffer, PDP due to buffer overflow, average packet delay, PLP and throughput, is built up for further investigations. We evaluate the impact of fading in wireless channels on performance analysis.

4.1 MCS adopted

As introduced in Chapter 1, AMC adopts a range of MCSs with different service rates. In wireless channel scenarios, AMC adaptively adjusts to instantaneous channel conditions by assigning the proper MCS for transmission to help combat wireless fading. Before understanding how AMC works at the physical layer, we need to specify a range of MCSs for AMC to choose from. The MCS can be divided into two parts: modulation scheme and coding scheme, which have been specified in section 2.2. Note that we only consider a single channel scenario in this chapter, which fits with the specification of IEEE 802.11a standard. Therefore, the MCSs adopted by AMC in this chapter are those with only one spatial stream in a wireless channel.

As mentioned in Chapter 1, AMC is widely used in packet-based wireless networks, and MCSs adopted by AMC have been standardised with an MCS index. In general, we take MCSs with index number from 0 to 7, which are for cases with one spatial stream, into consideration for cross-layer design in this chapter. The MCSs for one spatial stream considered in this chapter (IEEE 802.11a standard) are listed in Table 4.1.

Table 4.1 MCSs for one spatial stream in 802.11a

MCS Index	Modulation	Coding rate	Coded bits per symbol	Data bits per symbol
0	BPSK	1/2	1	1/2
1	QPSK	1/2	2	1
2	QPSK	3/4	2	3/2
3	16QAM	1/2	4	2
4	16QAM	3/4	4	3
5	64QAM	2/3	6	4
6	64QAM	3/4	6	9/2
7	64QAM	5/6	6	5

As suggested in [7] [50], we don't need to use all of the MCSs for AMC; instead, we can choose some of them for AMC because there is little reduction in average throughput compared to the case when using all of them for AMC. The selection of MCSs has been done by [22], and the MCSs chosen are listed in Table 4.2.

Table 4.2 MCSs chosen for AMC with one spatial stream

MCS Index	Modulation	Coding rate	Coded bits per symbol	Data bits per symbol
0	BPSK	1/2	1	1/2
1	QPSK	1/2	2	1
2	QPSK	3/4	2	3/2
4	16QAM	3/4	4	3
6	64QAM	3/4	6	9/2

We use the chosen MCSs listed in Table 4.2 for queueing analysis on cross-layer design incorporating with AMC, which is detailed in section 4.2.

4.2 Cross-layer design incorporating with AMC

4.2.1 System model

The task of this scenario is to explain queueing analysis on cross-layer design incorporating AMC. To begin with, a system model with single transmitter, single wireless channel and single receiver is constructed. An M/D/s(t)/K queue model with a range of working vacations is built up for queueing analysis. The block diagram for the cross-layer design is shown in Figure 4.1. The basic idea for queueing analysis on this cross-layer design has been discussed by [28] and we generalise the design as a queue model (M/D/s(t)/K), and validate the analytical results with system-level simulations both at the physical and the data-link layer. We discuss how we can make the cross-layer analyses extendable for further implementations in the following chapters.

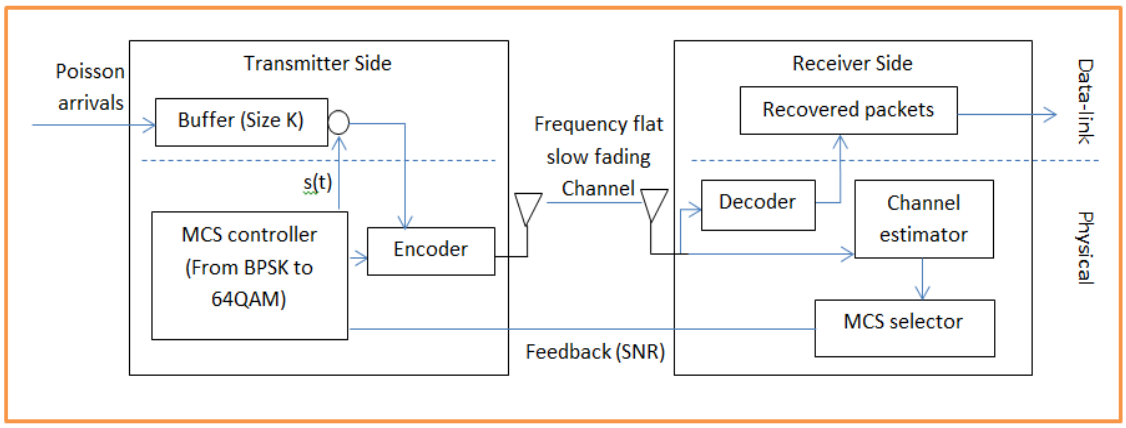

Figure 4.1 Functional block diagram for cross-layer design

The whole system shown in Figure 4.1 is time slot-based. As assumed by [28], incoming packets arrive at the data-link layer as a Poisson process, and enter a buffer of size K. The newly incoming packets will be dropped if the buffer is full. The MCS controller controls the number of servers, and this determines the service rate for the packet queue at the transmitter side, based on the information conveyed on an error-free feedback channel sent by the MCS selector at the receiver side. The MCS selector chooses the right MCS for transmission by using channel state information provided by the channel estimator. At the transmitter side, the served packets coming out of the buffer are encoded using convolutional codes with 1/2 coding rate and puncturing. The encoded bit streams are encapsulated into frames with some pilot and controlling

information, and transmitted through a frequency flat, slow Nakagami-m fading channel. At the receiver side, the received frames are broken down into packets after removing redundant information and decoding by puncturing recovery and a Viterbi decoder. Then the recovered packets are compared to original packets in order to get PER and system throughput.

AMC adopts a range of MCSs which are listed in Table 4.2. The task is to determine when to use which MCS. As mentioned above, the selection of MCS is influenced by instantaneous channel information determined by channel estimator at the receiver side. As inspired by [22], we make use of instantaneous SNR measured by the channel estimator for MCS selection. As indicated by [6], we can set a range of SNR threshold γ_n for adopted MCSs, and thus divide the whole range of SNR into several non-overlapping intervals $[\gamma_n, \gamma_{n+1})$. If instantaneous SNR $\gamma \in [\gamma_n, \gamma_{n+1})$, the MCS with mode label n will be selected for transmission.

For determining the SNR threshold γ_n, we need to specify the PER behaviour over a range of SNRs in an additive white Gaussian noise (AWGN) channel. A closed-form expression for PER in terms of SNR can be obtained based on [76], but the expression is too complex for analysis. For simplicity, [22] provides an asymptotic expression for PER which has been validated by Monte-Carlo simulation. We use this asymptotic approach for PER calculation which is obtained by (4.1).

$$PER_n(\gamma) = \begin{cases} a_n \times e^{-g_n \times \gamma}, & \text{for } \gamma \geq \gamma_{pn} \\ 1, & \text{otherwise} \end{cases} \quad (4.1)$$

$a_n, g_n \; and \; \gamma_{pn}$ are fitting parameters obtained by nonlinear regression based on Monte-Carlo simulation results, and the method for getting these parameters are detailed in Appendix A. In general, $\gamma_n \geq \gamma_{pn}$; otherwise the channel is too poor in SNR to use MCS with mode label n and PER will be 1. For a prescribed PER P_0 caused by channel noise, γ_n can be obtained by (4.2).

$$\gamma_n = \frac{-1}{g_n} \times \ln\left(\frac{P_0}{a_n}\right), \quad (4.2)$$

Table 4.2 has provided the MCSs adopted by AMC in this single user single channel scenario. As noted earlier, the system is slot-based and each timeslot lasts a period of a

frame duration. We assume that packet length $N_b = 1080\ bits$ ([28]) and data symbols per frame $N_s = 2160\ symbols/frame$, so that we can get the service rate measured in transmitted packets during each timeslot. Table 4.3 gives the parameter settings for each adopted MCS with mode label n, together with equivalent packet service rate for each timeslot.

Table 4.3 Parameter settings for MCSs with one spatial stream

Mode n	n=0	n=1	n=2	n=3	n=4	n=5
MCS Index	-	0	1	2	4	6
Modulation	-	BPSK	QPSK	QPSK	16QAM	64QAM
Coding rate	-	1/2	1/2	3/4	3/4	3/4
Coded bits per symbol	0	1	2	2	4	6
Data bits per symbol	0	0.5	1	1.5	3	4.5
Service rate (packets/slot)	0	1	2	3	6	9
a_n	0	274.7229	90.2514	67.6181	53.3987	35.3508
g_n	0	7.9932	3.4998	1.6883	0.3756	0.0900
$\gamma_{pn}(dB)$	-	-1.5331	1.0942	3.9722	10.2488	15.9784

We use mode n as a representation for each MCS, instead of the MCS index, from now on mainly for convenience. Note that there is also a mode 0 with a service rate of 0 packet/slot, which indicates that the instantaneous channel condition is too poor to use any of the MCSs adopted. Therefore, the server stops working to avoid transmission errors caused by noise in mode 0.

Once the MCSs adopted for AMC are determined, we need to investigate the transition behaviours among these MCSs in the slot-based system. We assume a frequency flat slow fading channel, which means the channel condition remains invariant during each timeslot, and the MCSs can only possibly be changed between adjacent modes or stay unchanged on a slot-by-slot basis. In other words, the transition probability $p_{m,n}^{MCS}$ from mode m to mode n satisfies (4.3).

$$p_{m,n}^{MCS} = 0, if\ |m-n| \geq 2, \quad (4.3)$$

We assume a Nakagami-m fading channel with probability density function (p.d.f) $p_\gamma(\gamma)$ in terms of SNR obtained by (4.4).

$$p_\gamma(\gamma) = \frac{m^m \times \gamma^{m-1}}{\overline{\gamma}^m \times \Gamma(m)} \times e^{-\frac{m\times\gamma}{\overline{\gamma}}}, \quad (4.4)$$

Note that $p_\gamma(\gamma)$ is Rayleigh distributed when m=1. $\Gamma(m)$ is the Gamma function obtained by (4.5).

$$\Gamma(m) = \int_0^\infty t^{m-1} \times e^{-t} dt, \quad (4.5)$$

Recall that mode n is selected if instantaneous SNR $\gamma \in [\gamma_n, \gamma_{n+1})$, thus the stationary probability π_n^{MCS} for the system to stay at mode n in Nakagami-m fading channel is obtained by (4.6).

$$\pi_n^{MCS} = \int_{\gamma_n}^{\gamma_{n+1}} p_\gamma(\gamma)\, d\gamma = \frac{\Gamma\left(m, \frac{m\times\gamma_n}{\overline{\gamma}}\right) - \Gamma\left(m, \frac{m\times\gamma_{n+1}}{\overline{\gamma}}\right)}{\Gamma(m)}, \quad (4.6)$$

$\Gamma\left(m, \frac{m\times\gamma_n}{\overline{\gamma}}\right)$ and $\Gamma\left(m, \frac{m\times\gamma_{n+1}}{\overline{\gamma}}\right)$ are the complementary incomplete Gamma function obtained by (4.7).

$$\Gamma(m, a) = \int_a^\infty t^{m-1} \times e^{-t} dt, \quad (4.7)$$

And the transition probability $p_{m,n}^{MCS}$ between adjacent modes has been given by (4.8) for N=5 in [11].

$$\begin{cases} p_{n,n+1}^{MCS} = \frac{N_{n+1}\times T_s}{\pi_n^{MCS}}, if\ n \in [0, N-1], n \in Z \\ p_{n,n-1}^{MCS} = \frac{N_n \times T_s}{\pi_n^{MCS}}, if\ n \in [1, N], n \in Z \end{cases}, \quad (4.8)$$

T_s is slot duration. π_n^{MCS} is the stationary probability for mode n obtained by (4.6). N_n is determined by (4.9).

$$N_n = \sqrt{2\pi \times \frac{m\times\gamma_n}{\overline{\gamma}}} \times \frac{f_d}{\Gamma(m)} \times \left(\frac{m\times\gamma_n}{\overline{\gamma}}\right)^{m-1} \times e^{-\frac{m\times\gamma_n}{\overline{\gamma}}}, \quad (4.9)$$

m is the coefficient of Nakagami-m channel (m=1 for Rayleigh fading channel). f_d is the Doppler spread which is measured in Hz. $\overline{\gamma}$ is the average SNR for a channel.

Furthermore, the transition probability $p_{n,n}^{MCS}$ for staying at the same mode between adjacent timeslots is obtained by (4.10).

$$\begin{cases} p_{0,0}^{MCS} = 1 - p_{0,1}^{MCS}, if\ n = 0 \\ p_{N,N}^{MCS} = 1 - p_{N,N-1}^{MCS}, if\ n = N \\ p_{n,n}^{MCS} = 1 - p_{n,n+1}^{MCS} - p_{n,n-1}^{MCS}, otherwise \end{cases}, (4.10)$$

Based on equations (4.3), (4.8) and (4.10), we construct the transition matrix for service rate $\boldsymbol{P}^{MCS}$ obtained by (4.11) which can be validated by (4.12).

$$\boldsymbol{P}^{MCS} = \begin{pmatrix} p_{0,0}^{MCS} & p_{0,1}^{MCS} & 0 & & 0 & 0 & 0 \\ p_{1,0}^{MCS} & p_{1,1}^{MCS} & p_{1,2}^{MCS} & \cdots & 0 & 0 & 0 \\ 0 & p_{2,1}^{MCS} & p_{2,2}^{MCS} & & 0 & 0 & 0 \\ & \vdots & & \ddots & & \vdots & \\ 0 & 0 & 0 & & p_{N-2,N-2}^{MCS} & p_{N-2,N-1}^{MCS} & 0 \\ 0 & 0 & 0 & \cdots & p_{N-1,N-2}^{MCS} & p_{N-1,N-1}^{MCS} & p_{N-1,N}^{MCS} \\ 0 & 0 & 0 & & 0 & p_{N,N-1}^{MCS} & p_{N,N}^{MCS} \end{pmatrix}, (4.11)$$

$$\boldsymbol{\pi}^{MCS} = \boldsymbol{\pi}^{MCS} \times \boldsymbol{P}^{MCS}, \sum_{n\in[0,N]} \pi_n^{MCS} = 1, (4.12)$$

Therefore, the mode transition process forms an FSMC which is shown in Figure 4.2. In particular, it is a quasi-birth-death process, which can be analysed by matrix geometric method as specified in Chapter 1. The values obtained in Figure 4.2 are based on equations (4.3), (4.8), (4.10) and Table 4.3.

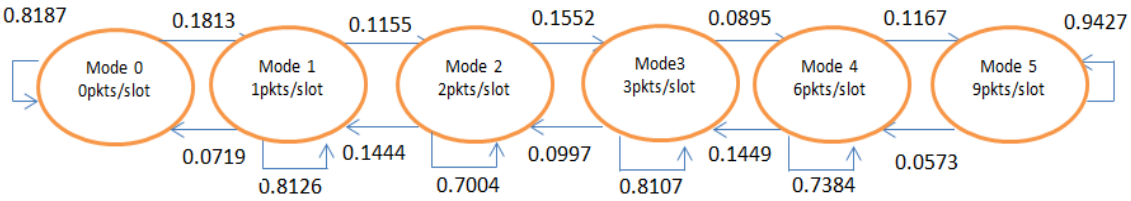

Figure 4.2 Transition process for service rate

4.2.2 Queueing analysis

In this section, we present the queueing analysis and obtain the metrics we want to evaluate, such as queue length in the buffer, PDP due to buffer overflow, average packet delay, PLP (caused by PDP and transmission error) and throughput through performance analysis using the matrix geometric method. This section provides a basic procedure for queueing analysis on cross-layer design, which lays the foundation for further extensions.

4.2.2.1 Arrival process

The arrival process A_t is Poisson distributed and the p.m.f for Poisson arrival process A_t which denotes the number of arrival packets within t-th timeslot is obtained from (4.13).

$$P(A_t = a) = \begin{cases} \frac{(\lambda)^a \times e^{(-\lambda)}}{a!}, a \in N, \\ 0, otherwise, \end{cases} \quad (4.13)$$

λ is mean arrival rate for each timeslot δτ.

4.2.2.2 Service process

Service process S_t is determined by the AMC operating at the physical layer in order to adjust to the instantaneous channel conditions. The number of packets that gets transmitted per slot is totally determined by the mode selected, which is equivalent to a deterministic service process with varied number of servers. The number of servers can only be chosen from a service rate set $\boldsymbol{S}$, which is shown by (4.14).

$$S_t \epsilon \boldsymbol{S}, \boldsymbol{S} = \{s_n, n\epsilon[0, N]\}, (4.14)$$

Based on Table 4.3, the service rate set $\boldsymbol{S} = \{0,1,2,3,6,9\}$ is measured by packets per slot.

The queueing analysis relies on the service rate probability transition matrix $\boldsymbol{P}^{MCS}$ and the stationary probability distribution $\boldsymbol{\pi}^{MCS}$ for each mode adopted, and the matrix geometric method in section 4.2.1.

In order to keep the system stable, we should make sure that average arrival rate λ is less than average service rate, that is, the system load ρ should satisfy the condition

specified in (4.15).

$$\rho = \frac{Average\ overall\ arrival\ rate}{Average\ service\ rate} = \frac{\lambda}{\sum_{n=1}^{N} \pi_n^{MCS} \times s_n} \leq 1, \quad (4.15)$$

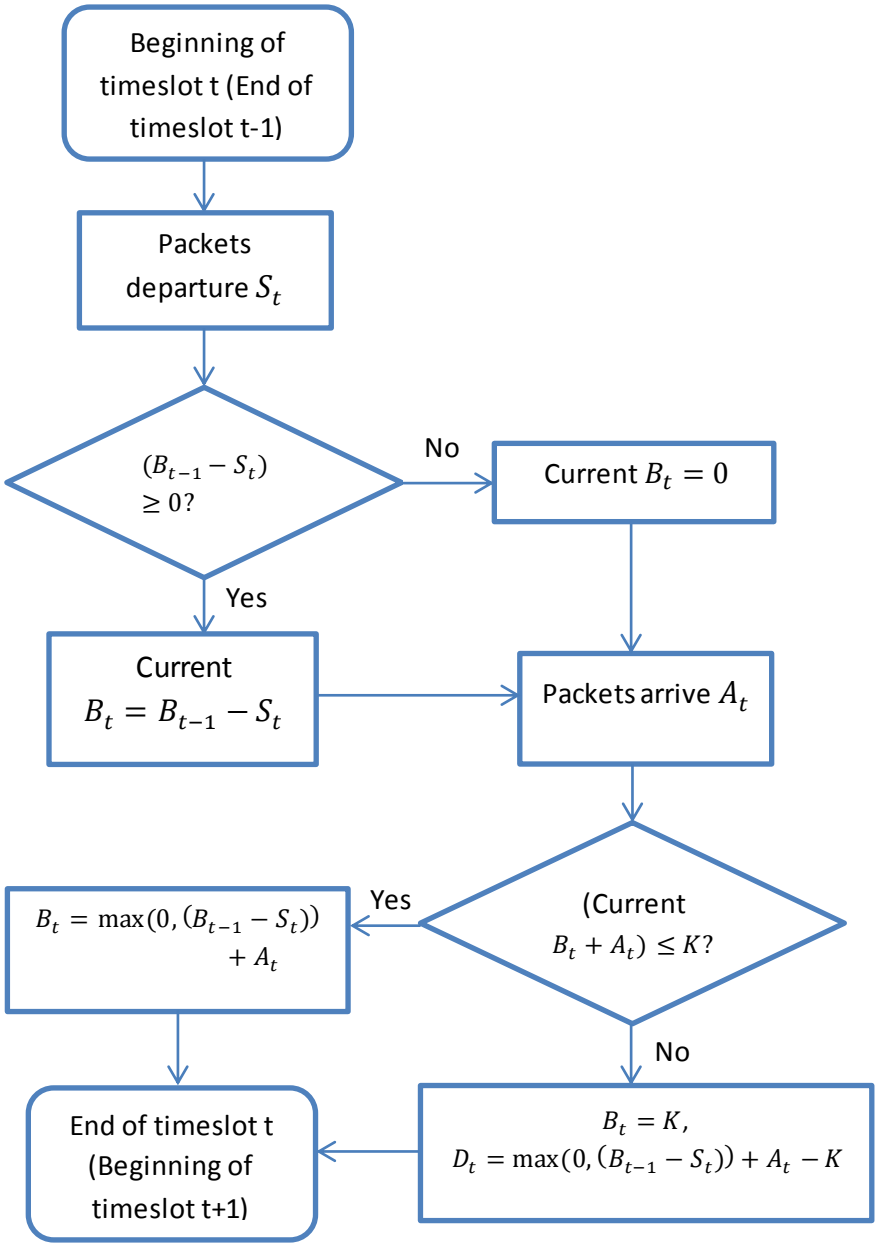

Figure 4.3 Flow chart for obtaining B_t and D_t for M/D/s(t)/K queue model

4.2.2.3 Queue state transition

Let B_t denote the queue state at the end of the t-th timeslot, or equivalently, the queue state at the beginning of (t+1)-th timeslot. In addition, the system serves at most S_t packets at the beginning of the t-th timeslot; or serves B_{t-1} packets if $B_{t-1} < S_t$. After packet departures, A_t packets come into the buffer (with buffer size K) within the t-th timeslot. If the current queue state after packet arrivals is larger than the buffer size K, then $B_t = K$ and excess packets will be dropped due to buffer overflow to form the packet drop process D_t.

The queue state B_t at the end of t-th timeslot after packet arrivals and departures is obtained by (4.16).

$$B_t = \min(K, \max(0, (B_{t-1} - S_t)) + A_t), \quad (4.16)$$

The flow chart of obtaining queue state B_t and packet drop D_t for each timeslot is shown in Figure 4.3.

Then we obtain the probability transition matrix for the M/D/s(t)/K queue model with a range of server working vacations. As s(t) is no longer a fixed value in an M/D/s(t)/K, an embedded FSMC, which considers both queue state transitions and service rate transitions, is built up. As a consequence, the probability transition matrix $\boldsymbol{P}^{M/D/s(t)/K}$ for queue state transition should consider both the queue state process B_t and the service process S_t. In general, the transition matrix is obtained by (4.17) with each element $\boldsymbol{A}_{B_{t-1},B_t}^{M/D/s(t)/K}$ (4.18).

$$\boldsymbol{P}^{M/D/s(t)/K} = \begin{pmatrix} \boldsymbol{A}_{0,0}^{M/D/s(t)/K} & \cdots & \boldsymbol{A}_{0,K}^{M/D/s(t)/K} \\ \vdots & \ddots & \vdots \\ \boldsymbol{A}_{K,0}^{M/D/s(t)/K} & \cdots & \boldsymbol{A}_{K,K}^{M/D/s(t)/K} \end{pmatrix}, \quad (4.17)$$

$$\boldsymbol{A}_{B_{t-1},B_t}^{M/D/s(t)/K} = \begin{pmatrix} p_{(B_{t-1},0),(B_t,0)}^{M/D/s(t)/K} & \cdots & p_{(B_{t-1},0),(B_t,N)}^{M/D/s(t)/K} \\ \vdots & \ddots & \vdots \\ p_{(B_{t-1},N),(B_t,0)}^{M/D/s(t)/K} & \cdots & p_{(B_{t-1},N),(B_t,N)}^{M/D/s(t)/K} \end{pmatrix}, (4.18)$$

And the elements $p^{M/D/s(t)/K}_{(B_{t-1},S_t),(B_t,S_{t+1})}$ of matrix $\boldsymbol{A}^{M/D/s(t)/K}_{B_{t-1},B_t}$ satisfies (4.19).

$$p^{M/D/s(t)/K}_{(B_{t-1},S_t),(B_t,S_{t+1})} = 0, for\ (B_{t-1} - B_t) > S_t, \quad (4.19)$$

More specifically, the probability $p^{M/D/s(t)/K}_{(u,c),(v,d)}$ for transitioning from $(B_{t-1} = u, S_t = c)$ to $(B_t = v, S_{t+1} = d)$ is obtained using (4.20).

$$p^{M/D/s(t)/K}_{(u,c),(v,d)} = p(B_t = v, S_{t+1} = d | B_{t-1} = u, S_t = c) = p(S_{t+1} = d | S_t = c) \times p(B_t = v | B_{t-1} = u, S_t = c) = p^{MCS}_{c,d} \times p(B_t = v | B_{t-1} = u, S_t = c), \quad (4.20)$$

$p^{MCS}_{c,d}$ is the transition probability for the service rate which is obtained using (4.11). And $p(B_t = v | B_{t-1} = u, S_t = c)$ can be obtained by using (4.21).

$$p(B_t = v | B_{t-1} = u, S_t = c) = \begin{cases} P(A_t = v - \max(u - c)), if\ v \in [0, K) \\ 1 - \sum_{v \in [0,K)} p(B_t = v | B_{t-1} = u, S_t = c)\ , if\ v = K' \end{cases} \quad (4.21)$$

$P(A_t = a)$ is the Poisson distribution obtained using (4.13).

Then we can derive the stationary distribution $\boldsymbol{\pi}^{M/D/s(t)/K}$ for queue state and service state for M/D/s(t)/K model with a range of working vacations using (4.22).

$$\boldsymbol{\pi}^{M/D/s(t)/K} = \boldsymbol{\pi}^{M/D/s(t)/K} \times \boldsymbol{P}^{M/D/s(t)/K}, \quad \sum_{B \epsilon [0,K]} \sum_{S \epsilon \boldsymbol{S}} \boldsymbol{\pi}^{M/D/s(t)/K}_{(B,S)} = 1, \quad (4.22)$$

4.2.2.4 Performance analysis

Based on (4.16) and Figure 4.3, we obtain the process for packet drop D_t at the end of t-th timeslot by (4.23).

$$D_t = \max(0, \max(0, (B_{t-1} - S_t)) + A_t - K), \quad (4.23)$$

Thus, we can get the average number of packets dropped $E(D)$ within each timeslot δτ by using (4.24).

$$E(D) = \sum_{a=0}^{\infty} \sum_{B_{t-1}=0}^{K} \sum_{S_t \epsilon \boldsymbol{S}} D_t \times P(A_t = a) \times \pi^{M/D/s(t)/K}_{(B_{t-1},S_t)} = \sum_{a \epsilon [0,\infty), B_{t-1} \epsilon [0,K], S_t \epsilon \boldsymbol{S}} [\max(0, \max(0, (B_{t-1} - S_t)) + A_t - K) \times P(A_t = a) \times \pi^{M/D/s(t)/K}_{(B_{t-1},S_t)}], \quad (4.24)$$

$P(A_t = a)$ is the stationary distribution of packet arrivals specified in (4.13) and $\pi_{(B_{t-1},S_t)}^{M/D/s(t)/K}$ is the element of the stationary distribution $\boldsymbol{\pi}^{M/D/s(t)/K}$ for the queue state in the buffer and the service state (4.22).

Then the PDP for M/D/s(t)/K queue model is obtained by (4.25) based on (4.24).

$$PDP = \frac{Average\ number\ of\ packets\ dropped}{Average\ number\ of\ packets\ arrived} = \frac{E(D)}{\lambda}, \quad (4.25)$$

And the average queue length $E(B)$ for M/D/s(t)/K queue model is obtained by using (4.26) from (4.22).

$$E(B) = \sum_{\substack{B\epsilon[0,K] \\ S\epsilon\boldsymbol{S}}} B \times \pi_{(B_{t-1},S_t)}^{M/D/s(t)/K}, \quad (4.26)$$

By Little's Law, we can obtain the average packet delay $E(W)$, which is the average packet waiting time before getting served, by using (4.27) from (4.25) and (4.26).

$$E(W) = E(B)/[\lambda \times (1 - PDP)], \quad (4.27)$$

Then we can obtain PLP caused by PDP due to buffer overflow and average PER $\overline{PER}$ due to transmission errors based on prescribed PER P_0 by (4.28).

$$PLP = 1 - (1 - PDP) \times (1 - \overline{PER}), \quad (4.28)$$

The throughput within each timeslot can be obtained by (4.29).

$$Average\ throughput = \lambda \times (1 - PLP) = \lambda \times (1 - PDP) \times (1 - \overline{PER}), \quad (4.29)$$

4.2.3 Numerical results

Before starting the performance analysis, we should pre-define some parameters for simulation use. These parameters with their values are listed in Table 4.4,

Table 4.4 Parameter settings for cross-layer analysis

Parameter	Value
Frame duration T_f	2 ms
Total simulation time	200 s
Packet length N_b	1080 bits
Symbol/frame N_s	2160
Doppler frequency f_d	10 Hz
Packet error rate P_0	10^{-2}
Fading channel	Slow Nakagami-m with m=1
Buffer length K	50
Average service rate	4.33 packets/slot
Average SNR $\overline{\gamma}$	15 dB

4.2.3.1 Queue length in the buffer

Firstly, we analyse an M/D/s(t)/K queue model with a range of different levels of working vacations obtained in Figure 4.2. The service rate for M/D/s(t) queue model is no longer a fixed value. The stationary distribution of buffer size for M/D/s(t)/K queue model with different system loads is obtained by equation (4.22) extending M/D/1/K vacation model. Note that we also consider the fading factor, which is the shaping parameter m for Nakagami-m distribution, for wireless channel. In this scenario, we use Nakagami-m (with m=1, which is equivalent to Rayleigh) slow fading channel which means the transmission mode can only be changed between adjacent ones and the transition probability matrix can be obtained by equations (4.17)-(4.21).

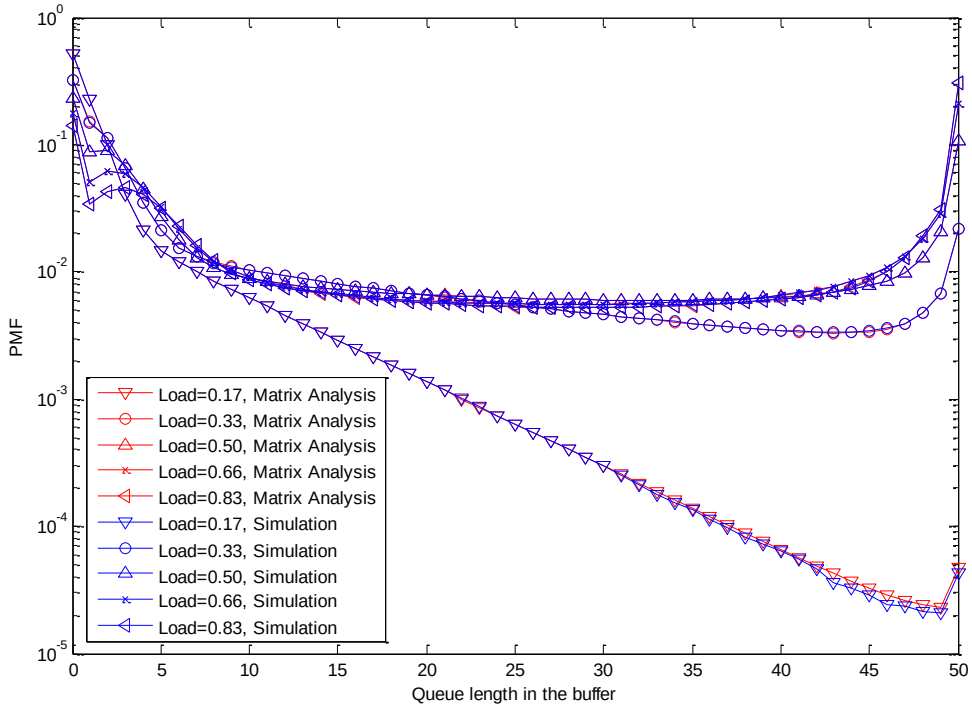

Figure 4.4 Steady-state probability of buffer state for M/D/s(t)/K vacation model

Figure 4.4 validates and obtains the queue length in the buffer with different system loads. We have following observations:

Figure 4.4 shows the impact of fading on queue model with different system loads. In Figure 3.4 which describes the queue state behaviour for an M/D/1/K queue model, the gaps between curves with different system loads become more obvious (often a gap of 2 orders of magnitude when the probability is around 10^{-12}) as the queue length in the buffer increases, however, the result changes when we introduce slow Nakagami-m fading channel into system model. As can be seen in Figure 4.4, when system load gets higher, the gaps become less obvious (e.g. the difference is over 2 orders of magnitude between system load of 0.17 and system load of 0.33 at queue length equals 45; while the difference becomes less than 1 order of magnitude between a load of 0.33 and a load of 0.50 at queue length equals 45). Therefore, we can primarily conclude that the impact of fading changes the probabilistic behaviour of the queue length in the buffer.

On the other hand, the impact of fading for M/D/s(t)/K model can be shown when compared to M/D/1/K vacation model. Figure 4.5 is the comparison when system load=0.17,

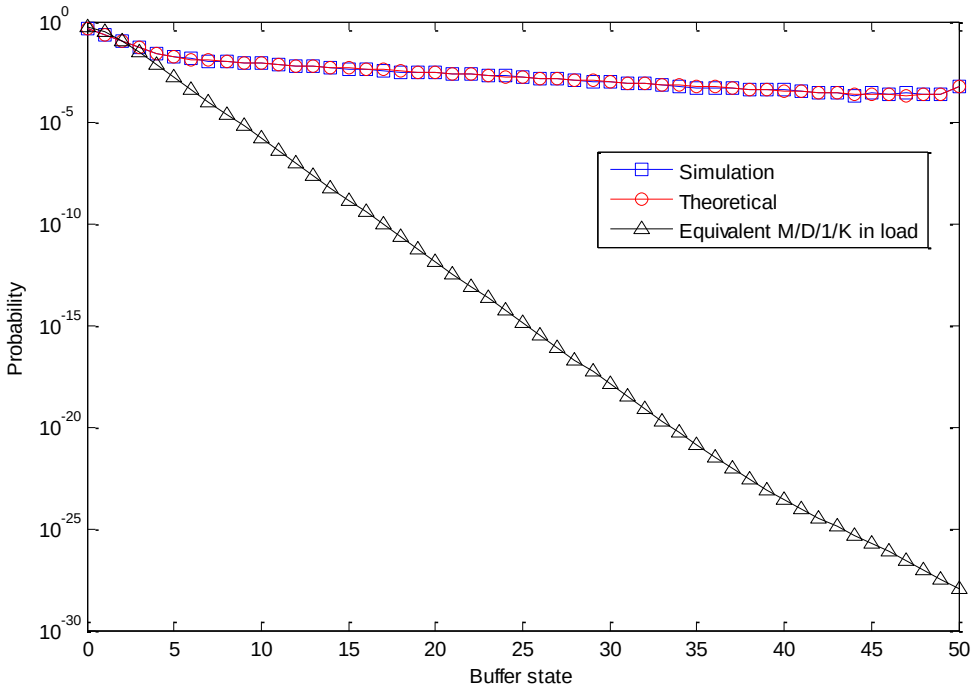

Figure 4.5 Comparison between M/D/s(t)/K and M/D/1/K queue model

We have following observations:

Compared to the black triangle line which is equivalent M/D/1/K vacation model, we can find a gap of over 20 orders of magnitude, which also results from the impact of fading, between these two queue models. M/D/s(t)/K queue model adopts a range of working vacations with different data rates. The SNR for mode 0 is too poor (usually SNR<2dB) to satisfy the requirement for any MCS; while the service rates for mode 1-3 are also lower than the average service rate specified in Table 4.4. Therefore, when the channel condition can only support MCSs with service rates below average service rate, there is higher probability to cause buffer overflow.

Next, we would like to check the impact of shaping parameter m used by Nakagami-m fading channel on M/D/s(t)/K queue model. The probability density function for Nakagami-m fading channel is obtained by equation (4.30):

$$p_\gamma(\gamma) = \frac{1}{\gamma \times \Gamma(m)} \times \left(\frac{m \times \gamma_n}{\overline{\gamma}}\right)^m \times e^{-\frac{m \times \gamma_n}{\overline{\gamma}}}, \quad (4.30)$$

Where m is the shaping parameter, $\overline{\gamma}$ is the average SNR, γ_n is SNR threshold for mode n obtained by equation (4.2) and $\Gamma(.)$ is Gamma function. For m=1, it is a Rayleigh

fading channel; when m>1, Nakagami-m fading channel can provide good coordination to Ricean fading channel; when m<1, the fading condition is worse than the Rayleigh fading channel.

Figure 4.6-4.8 show the impact of different shaping parameters m on probability distributions of queue length under low, medium and high system load (Load=0.2, 0.5, 0.8 respectively). As suggested by [77], the shaping parameter m ranges from 0.5 to 4.5 with an average of 1.56 based on measurements for urban scenarios. Therefore, we choose to simulate and analyze scenarios with m from 0.5 to 2.5 to cover most cases.

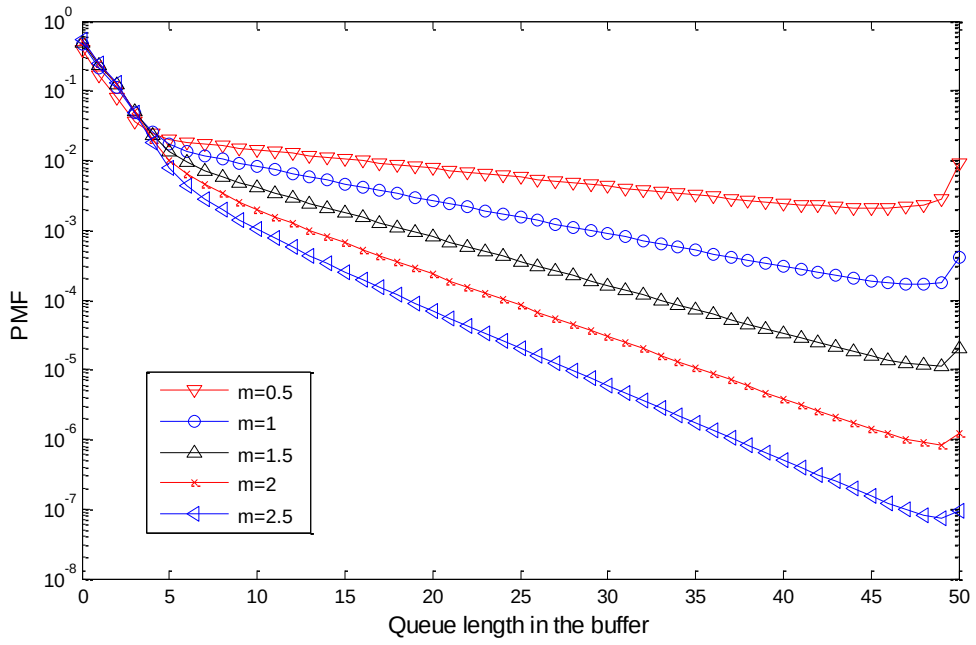

Figure 4.6 Impact of different fading channels on M/D/s(t)/K model when load=0.2

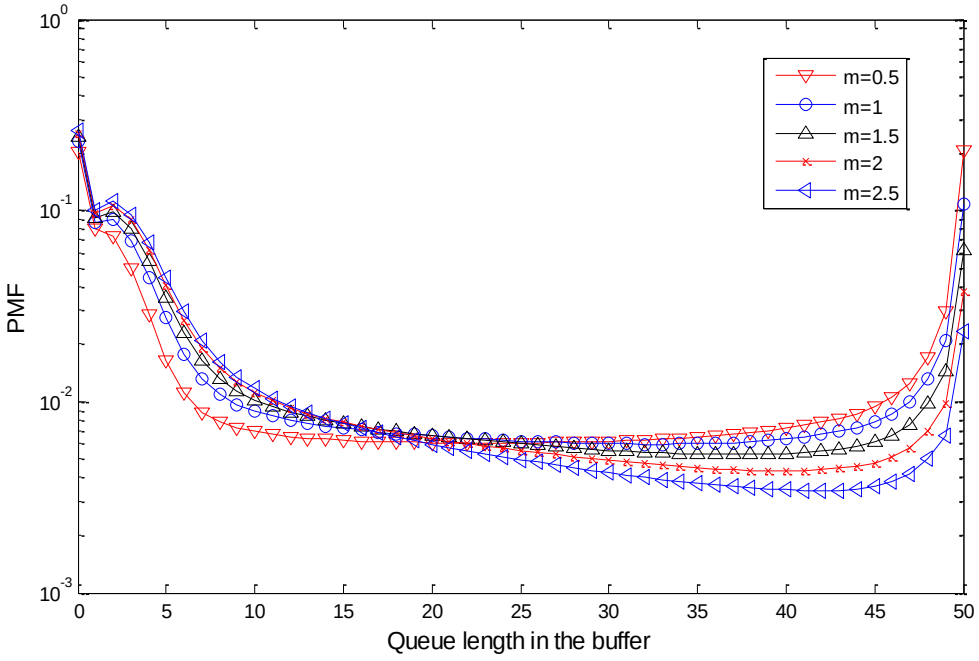

Figure 4.7 Impact of different fading channels on M/D/s(t)/K model when load=0.5

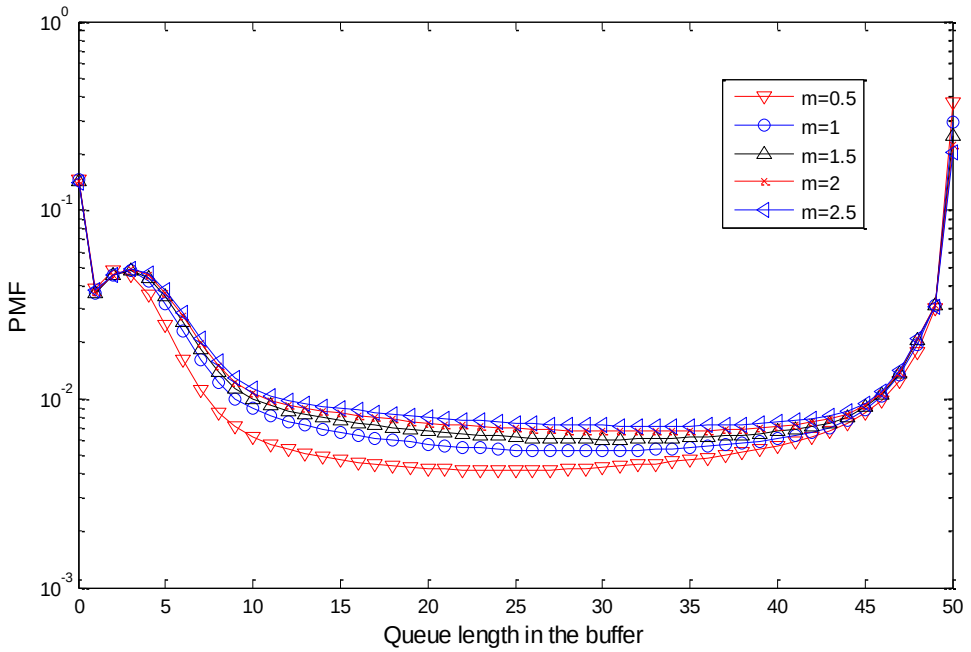

Figure 4.8 Impact of different fading channels on M/D/s(t)/K model when load=0.8

We have following observations:

1. Figure 4.6-4.8 study the impact of fading on queue model. Probabilities are accumulated at the tail of each curve because of limited buffer size. In addition, the probabilities at the tail get larger when the system load gets larger. The shaping parameter m in the Nakagami-m model has a far greater effect on packet queueing at lower loads.

2. Figure 4.6 shows that when system load is low, the impact of different shaping parameters m on M/D/s(t)/K queue model becomes obvious (around 1 order of magnitude between adjacent curves when queue length is 50) as queue length in the buffer increases.

3. Figure 4.6-4.8 show that PMF curves for queue state with different shaping parameters often intercept with each other at different points. When the system load gets higher, the corresponding values of queue lengths for interception points get larger. For example, the interception points are around 5 in queue length when system load = 0.2; while around 18 in queue length when system load=0.5; moreover, around 49 in queue length when system load=0.8.

4.2.3.2 PDP due to buffer overflow

Next, we validate and analyse PDP due to buffer overflow with two aspects: 1. Check PDP with different buffer sizes when m=1 shown by Figure 4.9, and 2. Check PDP with different shaping parameters m when K=50 shown by Figure 4.10.The red lines are obtained by matrix geometric analysis and blue lines are simulation results. We have following observations:

1. Figure 4.9 and 4.10 show that the gaps between curves become more obvious (over 1 order of magnitude when the system load is 0.1) when the system load decreases.

2. Compared to Figure 3.15 which analyses PDP for M/D/1/K vacation models with different buffer sizes, Figure 4.9 shows higher PDPs which indicate the impact of fading. While Figure 4.10 reveals the impact of fading degrees for fading channel on the queue model. More specifically, PDPs get larger when the fading gets worse (decrease in shaping parameter m).

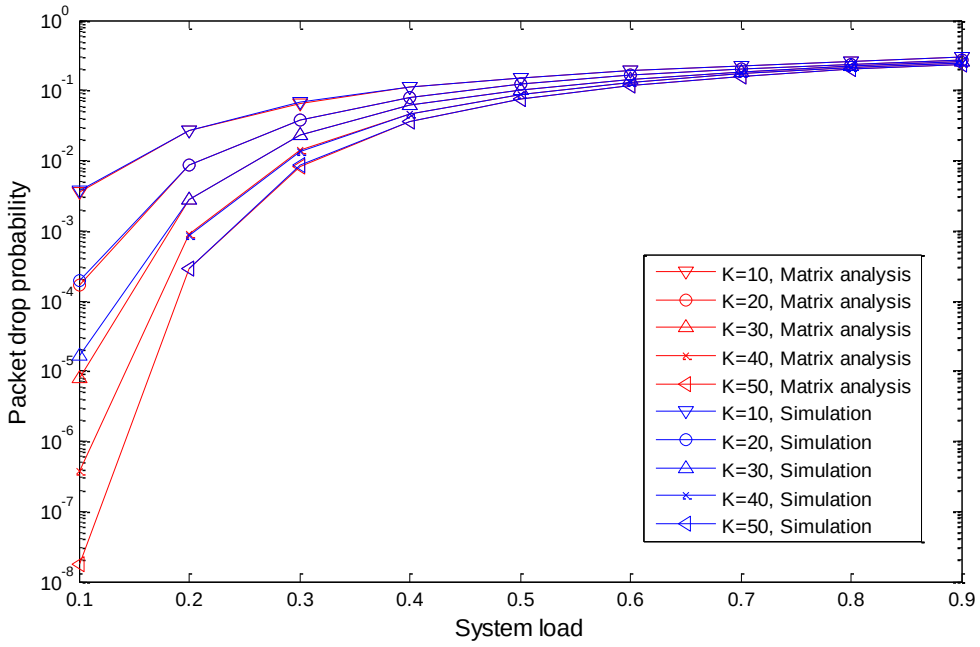

Figure 4.9 PDP with different buffer sizes

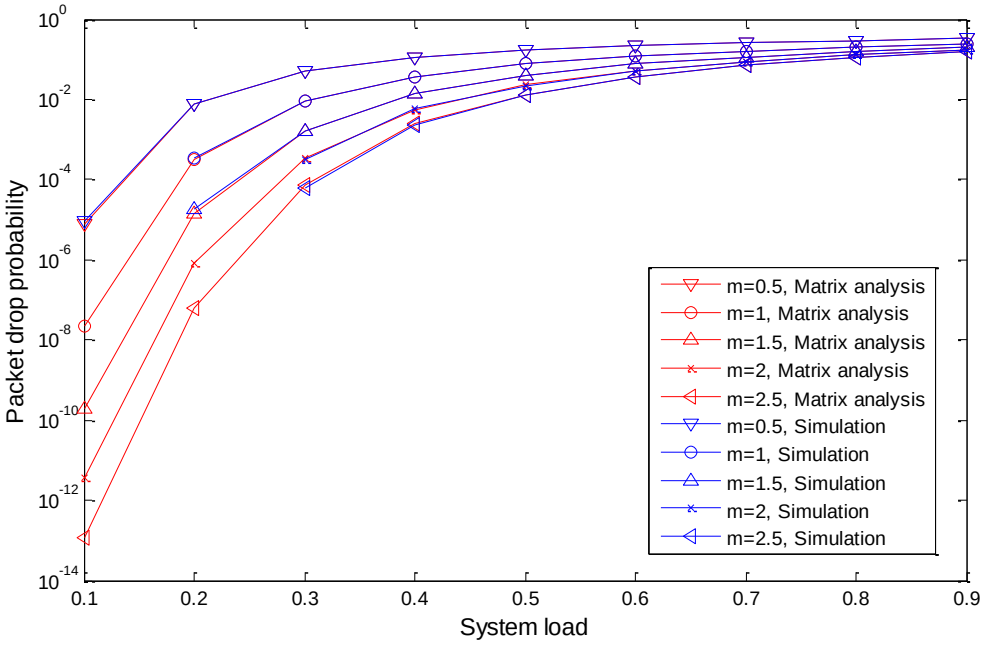

Figure 4.10 PDP with different shaping parameters m

4.2.3.3 Average queueing delay, PLP and throughput

In this section, we validate and analyse average queueing delay, PLP and throughput for the proposed M/D/s(t)/K queue model which are respectively shown by Figure 4.11-4.13. We have following observations:

1. Figure 4.11 shows the average queueing delay for M/D/s(t)/K queue model with a range of working vacations. We see that the gaps between different curves become more obvious (over 1 slot in delay between adjacent curves when the system load is 0.9) when the system load gets larger; this is because we plot delay as an absolute (not relative) value. In addition, average queueing delays with different buffer sizes nearly converges to a point when system load=0.1 where buffer size has little impact. Compared to Figure 3.17, which captures the average queueing delay for M/D/1/K vacation model, M/D/s(t)/K queue model achieves higher average queueing delays. This is also influence by fading.

3. Figure 4.12 captures PLP with different buffer sizes. As shown by equation (4.28), PLP is determined by PDP and average PER, of which PDP has been captured by Figure 4.9 and average PER is approximately equal to 6.14×10^{-4} when we adopt the MCSs listed in Table 4.3. From 4.12, we obtain that PLP is mainly dependent on PDP when PDP is larger than 10^{-3} so that average PER is comparatively negligible; while PLP is mainly dependent on average PER when PDP is lower than 10^{-4}.

4. Figure 4.13 reveals the behaviour of throughput, which is measured in packets per slot, for M/D/s(t)/K queue model. The gaps between curves become more obvious (around 0.1 packet per slot in throughput between adjacent curves when the system load is 0.9) when the system load gets higher. We also get that longer buffer sizes leads to larger throughput; however, a longer buffer size is not always a good way to ensure the system performance. For example, from Figure 4.11, we conclude that longer buffer size results in longer average queueing delay. Therefore, finding a proper buffer size to trade-off between average queueing delay and throughput becomes an interesting field to investigate in, but it's out of this chapter's scope and the discussion on buffer sizes is given in Chapter 6.

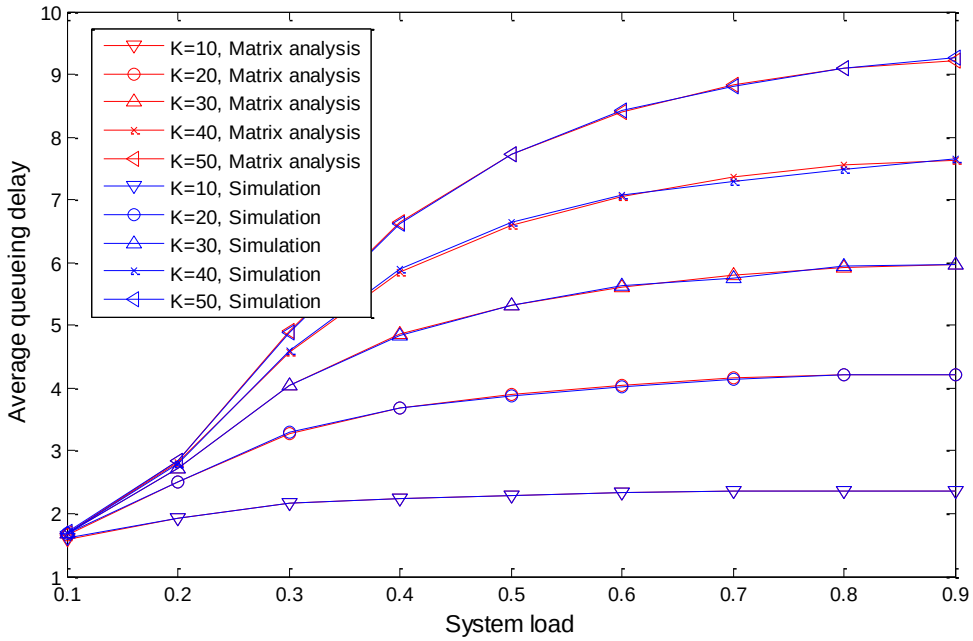

Figure 4.11 Average queueing delay for M/D/s(t)/K queue model

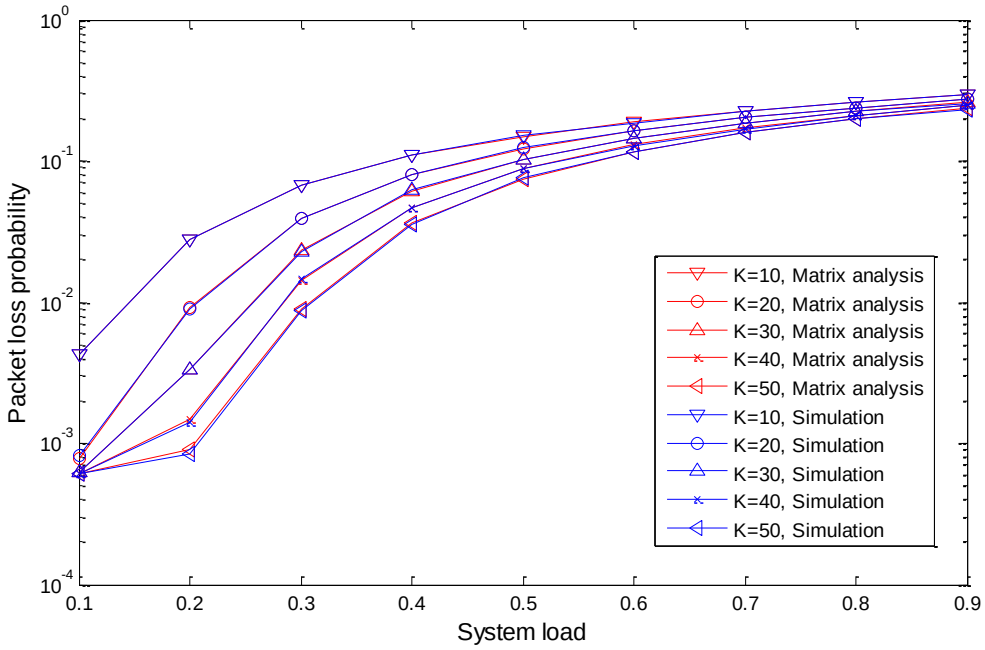

Figure 4.12 PLP for M/D/s(t)/K queue model

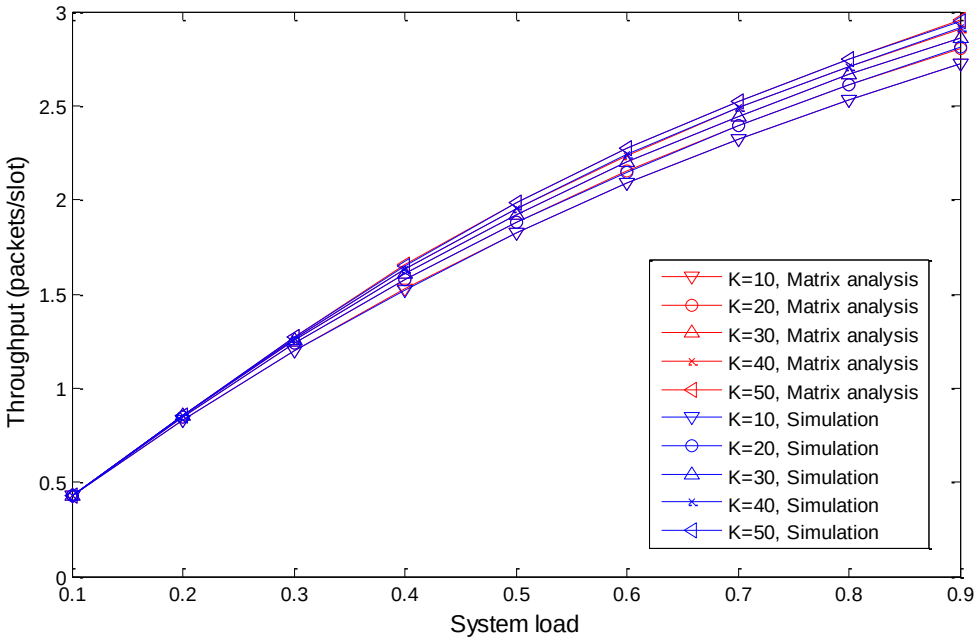

Figure 4.13 Throughput for M/D/s(t)/K queue model

4.3 Summary

In this chapter, a cross-layer design is built up which is suggested by [28], and we summarize it as M/D/s(t)/K queue model with a range of working vacations. At the data-link layer, packets arrived as a Poisson process; while a range of MCSs with various service rates indicating the channel variations are adopted by IEEE 802.11 standard. A Markov chain model for service rate transitions in a slow Nakagami-m fading channel is constructed. Brief introductions related to MCSs are given, and the queueing analysis on the cross-layer design is obtained. After that, we validate the performance analysis, including queue state, PDP, average queueing delay, PLP and throughput, through simulation, and reveal the impact of fading on our proposed M/D/s(t)/K queue model. In addition, the impact of degrees of fading is also discussed in terms of queue length in the buffer and PDP due to buffer overflow.

The work done in this chapter provides a fundamental framework for analysis on slot-based system which needs cross-layer analysis with fading channels, which is a useful approach for performance analysis on wireless networks.

Chapter 5 Cross-layer Analysis with Bursty Arrivals

In this chapter, we introduce the bursty arrivals model including the single on-off arrival model and the aggregated on-off arrival model into the cross-layer analysis. As we stated in Introduction section, Poisson arrivals, which is used for analysing the cross-layer design specified in Chapter 4, is not a reasonable approximation for bursty services. Therefore, we need to look for other traffic models for bursty behaviour. The literature has given abundant proof [46] [47] [78] to show that on-off traffic model is one of the main models used for bursty service approximation, especially in the scenarios where users require multiple services simultaneously. As a result, queue models including on-off/D/s(t)/K and N-burst/D/s(t)/K with a full range of working vacations have been obtained for cross-layer queueing analysis with bursty traffic. Following the similar framework of queueing analysis specified in Chapter 4, metrics including queue length in the buffer, PDP due to buffer overflow, average packet delay, PLP and throughput are obtained for the cross-layer analysis with bursty services. In addition, we compare the differences in performance between the Poisson arrival model and on-off traffic models. These indicate that the traffic model plays an important part in wireless packet network performance evaluation.

5.1 Bursty traffic models

In this section we introduce two well-accepted bursty traffic models in the literature including the single on-off traffic model and the aggregated on-off traffic model.

The single on-off traffic model, which gives a higher constant arrival rate during on periods and a lower constant arrival rate during off periods, is naturally suitable for bursty voice streams. In addition, the superposition of multiple single on-off traffic model, which is also known as aggregated on-off model (or N-burst model, Batch on-off model etc.), is good at approximating bursty data traffics based on extensive observations in the existing literature. As stated in the Introduction section, published work gives strong support to using the on-off traffic model instead of the Poisson arrival model to approximate bursty streams, making on-off arrival model one of the best choices when applied to queueing analysis on bursty streams in a cross-layer design.

Application to cross-layer analysis has not been published previously.

Note that we assume the system as slot-based and that the channels remain invariant within each timeslot. We focus on discrete-time traffic models and build Markov chains to find the probabilities of state transitions.

Also note that there are slight differences for definitions on on-off traffic models among the published works. Therefore, we need to explain the exact meanings of these traffic models to be used in our cross-layer design.

5.1.1 Single on-off traffic model

The single on-off traffic model uses two periods: ON periods and OFF periods. During ON periods, the packets arrive with an arrival rate of λ^{ON}; while during OFF periods, the packet arrival changes to a comparatively lower rate of λ^{OFF}. λ^{OFF} is often assumed to be zero where there is no packet arriving during OFF periods. The state transition rates r^{on-off} (transition rate from ON to OFF period) and r^{off-on} (transition rate from OFF to ON period) between ON and OFF periods can be shown by a two-state Markov chain which is shown in Figure 5.1.

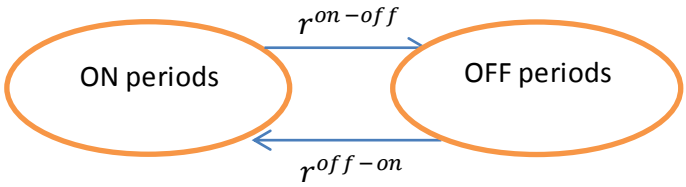

Figure 5.1 State transition rate between ON and OFF periods

The parameter values for the single on-off traffic model are listed in Table 5.1. Note that the values provided in Table 5.1 correspond to individual subscriber Internet scenario as suggested by [78], and these values are captured by real network simulations in order to obtain the distribution of HTTP sessions. Although the experiments carried out in [78] are not intended to be applied at the data-link layer, they provide the basic idea of bursty traffic models, which captures the burstiness of network traffic at all levels.

Table 5.1 Parameter settings for on-off traffic model

Parameter	Value
r^{on-off}, transitions/timeslot	1.445×10^{-2}
r^{off-on}, transitions/timeslot	1.084×10^{-2}
Timeslot duration T_s	2 ms

Based on parameters given in Table 5.1, we can obtain the state transition matrix for the single on-off traffic model. Both r^{on-off} and r^{off-on} are measured in transitions/timeslot and remain constant in the long run simulation. Both ON and OFF periods follow the exponential distribution with rate parameters of r^{on-off} and r^{off-on}.

As a result, the probability to stay in the ON state p_{on-on} is equal to the probability that the duration of ON period T^{ON} is larger than timeslot duration T_s, which is denoted as $P(T^{ON} > T_s)$ and can be obtained by the complement of CDF of exponential distribution (5.1):

$$p_{on-on} = P(T^{ON} > T_s) = e^{r^{on-off}\times T_s}, \quad (5.1)$$

And the probability to transfer from ON to OFF state p_{on-off} is shown by (5.2) following a similar method.

$$p_{on-off} = P(T^{ON} < T_s) = 1 - e^{r^{on-off}\times T_s}, \quad (5.2)$$

And we can also obtain p_{off-on} (Equation (5.3)) and $p_{off-off}$ (Equation (5.4)) respectively.

$$p_{off-on} = P(T^{OFF} < T_s) = 1 - e^{r^{off-on}\times T_s}, \quad (5.3)$$

$$p_{off-off} = P(T^{OFF} > T_s) = e^{r^{off-on}\times T_s}, \quad (5.4)$$

By Equations (5.1)-(5.4), we can construct the two-state Markov chain to give the state transition probabilities. The MC is shown by Figure 5.2.

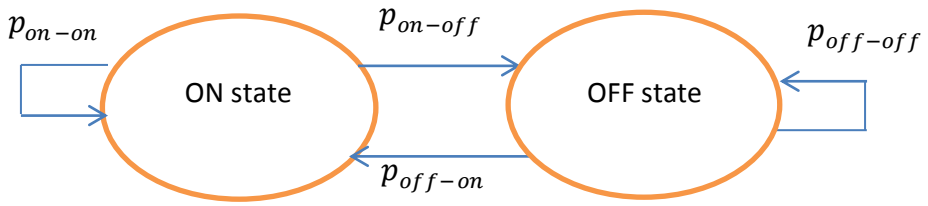

Figure 5.2 Markov chain for state transition probabilities

Correspondingly, we can construct the probability transition matrix $\boldsymbol{P}_{on-off}$ for single on-off traffic model shown by (5.5).

$$\boldsymbol{P}_{on-off} = \begin{pmatrix} e^{r^{on-off}\times T_s} & 1-e^{r^{on-off}\times T_s} \\ 1-e^{r^{off-on}\times T_s} & e^{r^{off-on}\times T_s} \end{pmatrix}, \quad (5.5)$$

As we know, typical voice codecs generate voice packets at a constant rate, thus, single on-off traffic model is used for approximating the behaviours of burstiness of voice and silence periods suppressed by codec with a constant arrival rate c^{ON} during ON periods.

5.1.2 Aggregated on-off traffic model

The aggregated on-off traffic model, which is good at approximating bursty data streams, is constructed by the superposition of multiple single on-off traffic models. The structure of aggregated on-off traffic model is shown by Figure 5.3.

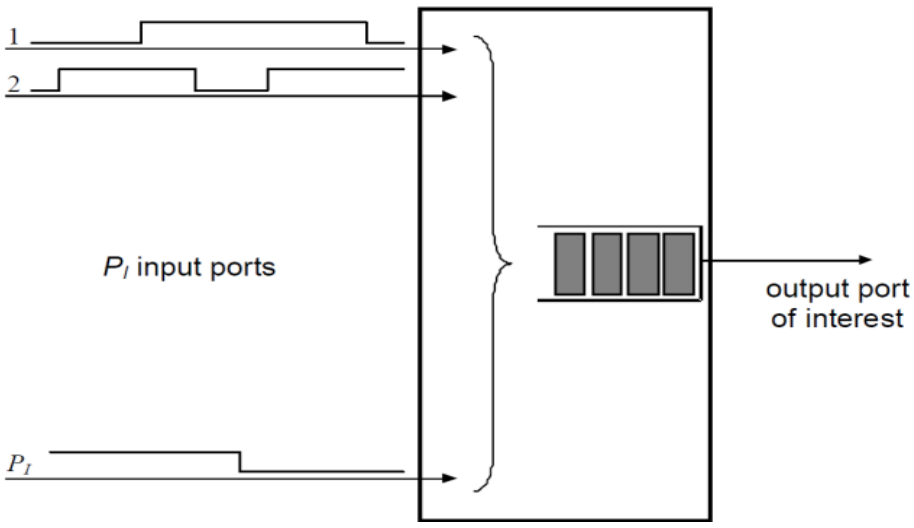

Figure 5.3 Aggregated on-off traffic model

As shown in Figure 5.3, there are a total of P_I independent single on-off streams at input ports. Each single on-off stream P_s input c_{P_s} packets per timeslot during ON periods with a constant arrival rate. In addition, each single on-off stream follows an independent state probability transition matrix. Therefore, the arriving packets within each timeslot for the aggregated on-off traffic model form a value set with finite elements which are dependent on single constant arrival rate value set $\{c_{P_s}\}$. As a result, the aggregated arrival rate can vary over time but still be limited to a finite dataset, which enables us to use the matrix geometric method for queueing analysis and ensure a limited degree of complexity.

5.2 Cross-layer analysis with bursty arrivals

5.2.1 System model

In this section, we apply bursty traffic models including the single on-off traffic model and the aggregated on-off traffic model, instead of Poisson arrival model, into the proposed cross-layer design specified in Chapter 4. We consider an end-to-end wireless link with single antenna at both transmitter and receiver side. The block diagram is shown in Figure 5.4.

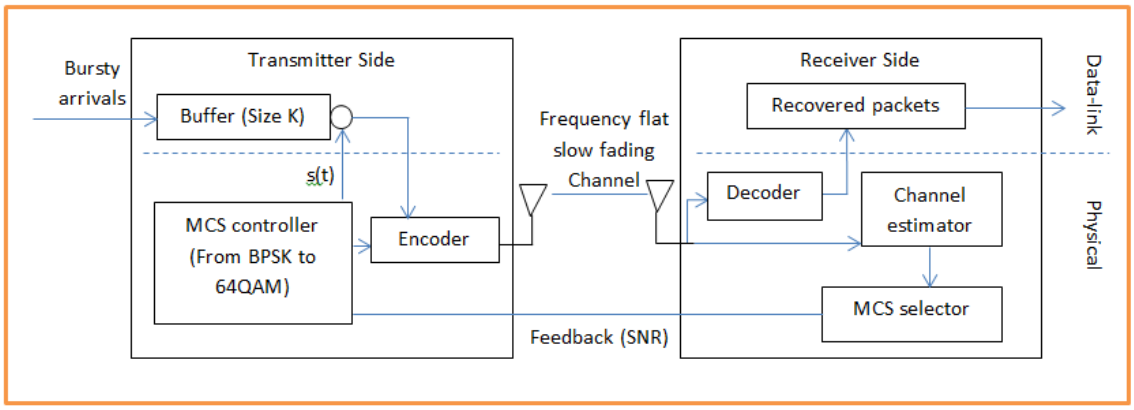

Figure 5.4 Block diagram for cross-layer analysis with bursty arrivals

At the transmitter side, the data-link layer packets with aggregated on-off arrivals are going into a queue with finite buffer size K. The packets are converted into frames and are served in a first-in-first-out (FIFO) manner, and we regard each frame as a timeslot.

The frame duration, which is equal to timeslot duration, T_s is fixed, and the number of packets transmitted per timeslot depends on the MCS selected. We assume a slow and frequency flat Nakagami-m fading channel, so that the mode used during one frame duration (also referred to as a timeslot) remains the same. The MCS may change at the end of a timeslot, in accordance with a Markov process. At the receiver side, a channel estimator measures the current Channel State Information (CSI), and an MCS selector at the receiver side then determines the MCS used during next frame duration and feeds back to the transmitter side through an error-free feedback channel. (The functional block diagram for the cross-layer design is similar to Figure 4.7)

Next, we adopt a series of MCSs to adjust to the channel variations. To make a reasonable comparison between bursty arrivals and Poisson arrivals, we select the same MCSs used in Chapter 4 for queueing analysis. The parameter settings for the proposed cross-layer analysis with bursty traffic are listed by Table 4.3.

We determine SNR thresholds $\{\gamma_n\}$ for the system to use mode n by Equations (4.1)-(4.2).

We also assume a frequency-flat slow Nakagami-m fading channel, which suggests the channel condition remains invariant during each timeslot. Using equations (4.3)-(4.11), we obtain the probability transition matrix for service rate $\boldsymbol{P}^{MCS}$ and construct a six-state Markov chain correspondingly. The Markov chain for service rate transitions is shown by Figure 4.8.

In addition, we generalize the proposed cross-layer analysis with single on-off traffic model as an on-off/D/s(t)/K queue model, and the cross-layer analysis with aggregated on-off traffic model as N-burst/D/s(t)/K queue model for further queueing analysis.

5.2.2 Queueing analysis

In this section, we focus on the queueing analysis of the proposed on-off/D/s(t)/K and N-burst/D/s(t)/K queue models, of which the system accepts multiple on-off arriving (fixed length) packets, and serves the incoming packets with a varied number of servers determined by the AMC and the fading channel, and has a finite buffer length of K packets. Arrival process, service process and queue state transitions are discussed for

the proposed queue models, and finally we construct FSMCs for queue state transition probabilities. Moreover, we address the feasibility of obtaining the stationary distribution of queue state based on queue state transition matrix. Finally, the metrics including queue state, PDP, average packet delay, PLP and throughput are discussed.

5.2.2.1 Arrival process

For single on-off source, the arrival process A_t^{on-off} is independent of the queue state and the service process (which is determined by the AMC algorithm). The transition matrix $\boldsymbol{P}_{on-off}$ for ON and OFF states is obtained using (5.5). Then, we can get the probabilities for the traffic model to be in ON and OFF states by solving (5.6).

$$\boldsymbol{\pi}^{on-off} = \boldsymbol{\pi}^{on-off} \times \boldsymbol{P}_{on-off}, \pi^{ON} + \pi^{OFF} = 1, \text{(5.6)}$$

Note that we assume a constant arrival rate c^{ON} during ON periods for single on-off traffic model. Therefore, the average arrival rate for single on-off traffic $\overline{A_t^{on-off}}$ can be calculated by (5.7).

$$\overline{A_t^{on-off}} = \pi^{ON} \times c^{ON}, \text{(5.7)}$$

For aggregated on-off arrivals, the arrival process $A_t^{N-burst}$ is also independent of the queue state and the service process. As shown in Figure 5.3, a user can require at most P_I streams, each of which P_s follows an on-off process with c_{P_s} packets/slot when it is in the ON state and 0 packets/slot in the OFF state. Each of these on-off processes follows an independent MC with probability transition matrix of $\boldsymbol{P}_{on-off}^{P_s}$ shown by (5.8) which is similar to (5.5).

$$\boldsymbol{P}_{on-off}^{P_s} = \begin{pmatrix} p_0^{P_s} & 1-p_0^{P_s} \\ 1-p_1^{P_s} & p_1^{P_s} \end{pmatrix}, \text{(5.8)}$$

Where state 0 is OFF, and state 1 is ON. $p_0^{P_s}$ is the probability for the process to stay in the OFF state, and $p_1^{P_s}$ is the probability to stay in the ON state.

Then we can obtain the stationary distribution for ON and OFF states of each on-off process by solving (5.9).

$$\boldsymbol{\pi}_{P_s} = \boldsymbol{\pi}_{P_s} \times \boldsymbol{P}_{on-off}^{P_s}, \quad \sum_{P_s \epsilon [1,P_I]} \boldsymbol{\pi}_{P_s} = 1, \quad (5.9)$$

We obtain $\pi_{P_s}^{ON}$ and $\pi_{P_s}^{OFF}$ as the stationary probability for stream $\mathrm{P_s}$ to be in ON state and OFF state respectively.

All of the single on-off arrivals of these P_I streams will combine together, and the outcome outputs an aggregated on-off arrival process. The average arrival rate for aggregated on-off arrival process $\overline{A_t^{N-burst}}$ can be calculated as (5.10).

$$\overline{A_t^{N-burst}} = \sum_{P_s=1}^{P_I} \pi_{P_s}^{on} \times c_{P_s}, (5.10)$$

Note that if $p_0^{P_s} = 1 - p_1^{P_s}$ for all P_I streams and the values of $\{c_{P_s}\}$ are all equal, the probability of a number of k single streams being in the ON or OFF state N^{ON} and N^{OFF} in any timeslot follows the binomial distribution (5.11):

$$\begin{cases} \Pr(N^{ON} = k) = \binom{P_I}{k} (\pi_{P_s}^{ON})^k (\pi_{P_s}^{OFF})^{P_I - k} \\ \Pr(N^{OFF} = k) = \binom{P_I}{k} (\pi_{P_s}^{OFF})^k (\pi_{P_s}^{ON})^{P_I - k} \end{cases}, (5.11)$$

In this case, the arrival process is memoryless because whether the state is in ON or OFF is actually independent of the states in the past; otherwise, we have to calculate $\Pr(N^{ON} = k)\ and\ \Pr(N^{OFF} = k)$ carefully by analyzing each individual $\boldsymbol{P}_{on-off}^{P_s}$. Assuming $\{\boldsymbol{T}_k\}$ is the set of all the combination of k streams in ON state with $\binom{P_I}{k}$ elements, and denote each combination as $\boldsymbol{T}_k^i$, and each complementary combination as $C(\boldsymbol{T}_k^i)$, then (5.12):

$$\begin{cases} \Pr(N^{ON} = k) = \sum_{i=1}^{\binom{P_I}{k}} (\prod_{m \epsilon \boldsymbol{T}_k^i} \pi_m^{ON} \times \prod_{n \epsilon C(\boldsymbol{T}_k^i)} \pi_n^{OFF}) \\ \Pr(N^{OFF} = k) = \sum_{i=1}^{\binom{P_I}{k}} (\prod_{m \epsilon \boldsymbol{T}_{P_I-k}^i} \pi_m^{ON} \times \prod_{n \epsilon C(\boldsymbol{T}_{P_I-k}^i)} \pi_n^{OFF}) \end{cases}, (5.12)$$

5.2.2.2 Service process

The same as the service process detailed in Chapter 4, the service process S_t is determined by the AMC algorithm operating at the physical layer in order to adjust to CSI. The number of servers can only be chosen from a set $\boldsymbol{S}$, of which (5.13),

$$S_t \epsilon \boldsymbol{S}, \boldsymbol{S} = \{s_n, n \epsilon [0, N]\}, s_n = b \times R_n, (5.13)$$

Where $\mathrm{R_n}$ is data bits per symbol specified in Table 4.3. Normally, we would like to keep $\{\mathrm{s_n}\}$ as integer values, thus, b is usually set to 2. Based on Table 4.3, for the modes with convolutional coding, $\boldsymbol{S} = \{0,1,2,3,6,9\}$.

In order to keep the system stable, we should maintain an average overall arrival rate less than the average service rate, that is, the system load ρ cannot be larger than 1, otherwise the queue will tend to grow to infinity and most of the new incoming packets will be dropped due to buffer overflow. System load constraints ρ_{on-off} and $\rho_{N-burst}$ are shown by (5.14) and (5.15) for single and aggregated on-off traffic models.

$$\rho_{on-off} = \frac{\overline{A_t^{on-off}}}{\overline{S_t}} = \frac{\pi^{ON} \times c^{ON}}{\sum_{n=1}^{N} \pi_n^{MCS} \times s_n} \leq 1, \text{(5.14)}$$

$$\rho_{N-burst} = \frac{\overline{A_t^{N-burst}}}{\overline{S_t}} = \frac{\sum_{P_S=1}^{P_I} \pi_{P_S}^{ON} \times c_{P_S}}{\sum_{n=1}^{N} \pi_n^{MCS} \times s_n} \leq 1, \text{(5.15)}$$

Where $\{\pi_n^{MCS}\}$ is the stationary probability distribution for staying in each mode of the service process.

5.2.2.3 Queue state transition

We denote B_t as the queue state at the end of a timeslot 't', or equivalently, the queue state in the beginning of timeslot 't+1'. At the beginning of timeslot t, the system serves at most S_t packets, or serves B_{t-1} packets if $B_{t-1} < S_t$. Then the on-off arriving packets at timeslot t, A_t, comes into the queue. If the current queue state after arrivals is larger than the buffer size K, then $B_t = K$ and excess packets will be dropped due to buffer overflow. The queue state transition process can be calculated by (5.16).

$$B_t = min(K, \max(0, (B_{t-1} - S_t)) + A_t), \quad \text{(5.16)}$$

Then we can obtain the probability transition matrix for queue state with each element $p_{u,v}$ (5.17).

$$p_{u,v} = p(B_t = v | B_{t-1} = u), 0 \leq u, v \leq K, \text{(5.17)}$$

For single on-off traffic model, $A_t \in \{0, c^{ON}\}$, since the model stays at either OFF state with 0 arrival rate or ON state with a constant arrival rate of c^{ON}.

For the aggregated on-off traffic model, we consider the summation of instantaneous arrival rates for all single on-off streams. More specifically, we denote by $\boldsymbol{A}$, which is a vector, all possible summations of combinations of single on-off streams $\{c_{P_s}\}$, and $A_t \in \boldsymbol{A}$. In this case, single on-off traffic is actually a special case for aggregated on-off traffic with $\boldsymbol{A} = \{0, c^{ON}\}$. Then we can merge these two traffic models into one case for further queueing analysis.

Since the next states for both of the arrival and service processes are only dependent on the current states; we have to consider all possible single cases for both arrival and service processes. That is (5.18),

$$p_{u,v} = \sum_{A_t, A_{t+1} \in \boldsymbol{A}, S_t, S_{t+1} \in \boldsymbol{S}} p(A_{t+1} = b, B_t = v, S_{t+1} = d | A_t = a, B_{t-1} = u, S_t = c), \quad (5.18)$$

For each $p_{(a,u,c),(b,v,d)}$ (5.19) of the probability transition matrix $\boldsymbol{P}_B$,

$$p_{(a,u,c),(b,v,d)} = p(A_{t+1} = b, B_t = v, S_{t+1} = d | A_t = a, B_{t-1} = u, S_t = c)$$

$$= p(A_{t+1} = b | A_t = a) \times p(S_{t+1} = d | S_t = c) \times p(B_t = v | A_t = a, B_{t-1} = u, S_t = c), \quad (5.19)$$

$p(A_{t+1} = b | A_t = a)$ is the transition probability for on-off traffic. For single on-off traffic, $p(A_{t+1} = b | A_t = a) = \boldsymbol{P}_{on-off}(a, b)$ where $\boldsymbol{P}_{on-off}$ is detailed by (5.5) and $\boldsymbol{P}_{on-off}(a, b)$ is the element of $\boldsymbol{P}_{on-off}$ with row a and column b.

For aggregated on-off traffic, if all of $c_{P_s} = 1$, then $p(A_{t+1} = b | A_t = a)$ (5.20) can be obtained from (5.11) and (5.12):

$$p(A_{t+1} = b | A_t = a) = \Pr(N^{ON} = a) \times \Pr(N^{ON} = b), \quad (5.20)$$

More generally, assuming $\{\boldsymbol{U}_l\}$ is the set of all combinations for the summation of arrival rate equals to $l \in \boldsymbol{A}$. Denote each combination as $\boldsymbol{U}_l^i$, and each complementary combination as $C(\boldsymbol{U}_l^i)$. Then $p(A_{t+1} = b | A_t = a)$ can be obtained by (5.21),

$$p(A_{t+1} = b | A_t = a)$$

$$= \left[\sum_{\boldsymbol{U}_a^i \in \boldsymbol{U}_a} \left(\prod_{m \epsilon \boldsymbol{U}_a^i} \pi_m^{ON} \times \prod_{n \epsilon C(\boldsymbol{U}_a^i)} \pi_n^{OFF}\right)\right] \times \left[\sum_{\boldsymbol{U}_b^i \in \boldsymbol{U}_b} \left(\prod_{j \epsilon \boldsymbol{U}_b^i} \pi_j^{ON} \times \prod_{k \epsilon C(\boldsymbol{U}_b^i)} \pi_k^{OFF}\right)\right], \quad (5.21)$$

$p(S_{t+1} = d | S_t = c)$ is the transition probability for service rate. And $p(B_t = v | A_t = a, B_{t-1} = u, S_t = c)$ (5.22):

$$p(B_t = v | A_t = a, B_{t-1} = u, S_t = c) = \begin{cases} 1, if\ v = min(K, \max(0, (u - c)) + a) \\ 0, if\ v \neq min(K, \max(0, (u - c)) + a) \end{cases}, \quad (5.22)$$

Next, we discuss the feasibility of getting the stationary distribution of the queue state $p_{u,v}$. As shown in [QL,SZ,GB], the stationary distribution $\boldsymbol{\pi}$ exists and is unique if the probability transition matrix of the enlarged FSMC is irreducible, homogeneous and positive recurrent. Therefore, we can have the lemma below:

Lemma: *The Markov chain of (a, u, c), where* $(a, u, c) \in \boldsymbol{A} \times \boldsymbol{B} \times \boldsymbol{S}$*, has only one closed communicating class, and therefore is positive recurrent.*

Proof: Firstly, we need to show there exists a multi-transition path with non-zero transition probability from state $(a, u, c) \rightarrow (b, v, d)$, where $(a, u, c), (b, v, d) \in \boldsymbol{A} \times \boldsymbol{B} \times \boldsymbol{S}$.

1) If $v = min(K, \max(0, (u - c)) + a)$, we can find a direct path from $(a, u, c) \rightarrow (b, v, d)$ with non-zero transition probability by (5.19).

2) If $v \neq v' = min(K, \max(0, (u - c)) + a)$, there exists a path with non-zero transition probability from $(a, u, c) \rightarrow (b', v', d')$, where either b' or d' is equal to 0; then we can always find a multi-transition path from $(b', v', d') \rightarrow (b, v, d)$ with each intermediate state (b'', v'', d''), where either b'' or d'' is equal to 0, to offset the difference between v' *and* v.

Then we can draw the conclusion that the finite state set $(a, u, c) \in \boldsymbol{A} \times \boldsymbol{B} \times \boldsymbol{S}$ forms a closed communicating class where every pair of states $(a, u, c), (b, v, d)$ in the set communicates with each other. Therefore, the state transition matrix is irreducible by definition. In addition, since the FSMC of state transition is independent of time, the state transition matrix is homogeneous by definition.

Finally, we can conclude that the Markov chain of state transition is positive recurrent, because [79] asserts that the finite state irreducible homogeneous Markov chain is positive recurrent. *(End of Proof)*

Based on the lemma stated above, we can draw the conclusion that the stationary distribution of queue state π_{P_B} exists and is unique, and $\pi_{P_B} \geq 0$.

Based on (5.17)-(5.22), we can get the stationary distribution of queue state by solving (5.23),

$$\boldsymbol{\pi}_{P_B} = \boldsymbol{\pi}_{P_B} \times \boldsymbol{P}_B, \quad \sum_{a \in \boldsymbol{A}} \sum_{c \in \boldsymbol{S}} \sum_{u \in [0,K]} \boldsymbol{\pi}_{P_B} = 1, \text{ (5.23)}$$

5.2.2.4 Performance analysis

We would like to analyze the PDP due to buffer overflow, the average queueing delay, the PLP (caused by PDP and transmission PER) and the system throughput for the proposed cross-layer analysis. Firstly, we have to obtain the packet drop process, denoted as D_t. As mentioned above, the excess packets are dropped within each timeslot 't' if the buffer if full, thus, we can obtain the expression for D_t (5.24),

$$D_t = \max(0, A_t - K + max(0, B_{t-1} - S_t)), \text{ (5.24)}$$

Thus, the average number of dropped packets during timeslot 't' is (5.25):

$$E(D) = \sum_{A_t \in \boldsymbol{A}} \sum_{B_{t-1}=0}^{K} \sum_{S_t=s_0}^{S_n} D_t \times P(A_t = a, B_{t-1} = u, S_t = c), \text{ (5.25)}$$

Together with the expression we obtained in (5.7) and (5.10), we can get the PDP due to buffer overflow as follows (5.26),

$$PDP = \frac{Average\ number\ of\ packets\ dropped}{Average\ number\ of\ packets\ arrived} = \frac{E(D)}{\overline{A_t}}, \text{ (5.26)}$$

Where $\overline{A_t}$ stands for either $\overline{A_t^{on-off}}$ or $\overline{A_t^{N-burst}}$. Next, we can obtain the average queueing delay $\overline{W}$ based on Little's Law [74]. The expression for $\overline{W}$ is shown as follows (5.27),

$$\overline{W} = \frac{E(B)}{(1-PDP) \times \overline{A_t}}, \text{ (5.27)}$$

Where $E(B)$ is the average queue length which can be obtained by (5.28) from (5.23),

$$E(B) = \sum_{B=0}^{K} (\sum_{a \in \boldsymbol{A}} \sum_{c \in \boldsymbol{S}} \boldsymbol{\pi}_{P_B}) \times K, \text{ (5.28)}$$

Then, we can get the PLP (5.29) which is determined by *PDP* (5.26) and average PER $\overline{PER}$,

$$PLP = 1 - (1 - PDP) \times (1 - \overline{PER}), \text{ (5.29)}$$

And finally, we can obtain the system throughput *T* by (5.30),

$$T = \overline{A_t} \times (1 - PLP), \text{ (5.30)}$$

5.2.3 Numerical results

In this section, we present numerical results for the probabilities associated with the queue state in the buffer, the PDP due to buffer overflow, the average queueing delay, the PLP and the throughput based on analytical expressions specified in 5.2.2. We validate the cross-layer analysis with bursty traffic models including single on-off model and aggregated on-off model by Monte-Carlo simulation. We also make comparisons between cross-layer models with Poisson arrivals and bursty arrivals with the same system loads by adjusting the parameters of the on-off probability transition matrix.

Similar to the parameter settings in Chapter 4, we assume frame duration $\mathrm{T_f} = 2$ ms, prescribed PER $\mathrm{P_0} = 10^{-2}$ (Note that $\mathrm{P_0}$ is the upper bound, and average PER $\overline{PER}$ is much less than $\mathrm{P_0}$, approximately equal to 6.14×10^{-4} with the MCSs adopted in Table 4.3), system average SNR=15 dB, buffer length K=50 packets for queue state analysis and K=[10 20 30 40 50] for other analyses (as suggested in [34] and [23]). The average service rate $\overline{S_t} \approx 4.3$ packets/slot for our chosen channel model.

The probability transition matrix $P_{on-off}^{P_s}$ should be carefully chosen in order to approximate the burstiness of each single on-off stream. As suggested in [78], a practical packet generation model for wireless networks is given by a discrete-time batch Markovian arrival process (DBMAP) for which on-off traffic should follow the probability transition matrix (5.31), and the values of matrix (5.31) are obtained by Table 5.1 and equations (5.1)-(5.4),

$$P_{on-off}^{P_s} = \begin{pmatrix} 0.9892 & 0.0108 \\ 0.0143 & 0.9857 \end{pmatrix}, \text{ (5.31)}$$

The matrix (5.31) suggests that the probabilities for single on-off arrival staying in the ON and OFF state are so high (0.9857 and 0.9892 respectively) that the state rarely changes, which is a reasonable approximation for bursty behavior, since arriving packets will keep coming during a long period of time staying in the ON state, and then no packet will come during another long period of time staying in the OFF state. As a comparison, we also introduce another probability transition matrix (5.32) with the same stationary distribution $\boldsymbol{\pi}_{P_s}$ as (5.31).

$$P_{on-off}^{P_s} = \begin{pmatrix} 0.5697 & 0.4303 \\ 0.5697 & 0.4303 \end{pmatrix}, \quad (5.32)$$

The matrix (5.32) suggests that whether the stream is in ON or OFF state is memoryless. Therefore, the matrix (5.32) cannot approximate bursty traffic streams, and the traffic behavior is like Poisson traffic's.

In addition, we assume all of the single on-off probability transition matrices are independent and identical with c_{P_s}=1 packet/slot. Therefore, we can obtain the single on-off stream arrival rate$= c_{P_s} \times \pi_{P_s}^{on} = 0.4303$ packet/slot. Therefore, the system load $\rho \approx 0.1$ for a single on-off arrival traffic, and so the system can support roughly up to 10 traffic streams.

In order to maintain the same system loads for comparisons, the Poisson arrival rate is adjusted to the same value as the one for aggregated on-off arrivals. In our case, the Poisson arrival rate is a multiple of 0.4303 packets/slot.

5.2.3.1 Bursty traffic validation

In this section, we validate the cross-layer analysis with bursty traffic models including single on-off model and aggregated on-off model by Monte-Carlo simulation. We validate the system with a range of system loads from 0.1 to 0.9, which suggests that c^{ON} ranges from 1 to 9 packets/slot for single on-off traffic model, or the number of streams ranges from 1 to 9 for the aggregated on-off traffic model.

Figure 5.5 and Figure 5.6 give the validation of queue length in the buffer, which lays the foundation for further performance analysis on PDP, PLP etc., for two bursty traffic models. We have the following observations:

The queue state results for both single and aggregated on-off traffic oscillate about mean values. However, practical buffer dimensioning would aim to use a smoothed p.m.f.; one solution for this would be,

$$\begin{cases} qs(k) = q(k), & when\ k = 0\ or\ K \\ qs(k) = \dfrac{q(k-1) + q(k) + q(k+1)}{3}, & else \end{cases} \quad (5.33)$$

Where $qs(k)$ is the smoothed version of original p.m.f. of queue state. But we will still use the original p.m.f. for the accuracy of other metrics based on queue state.

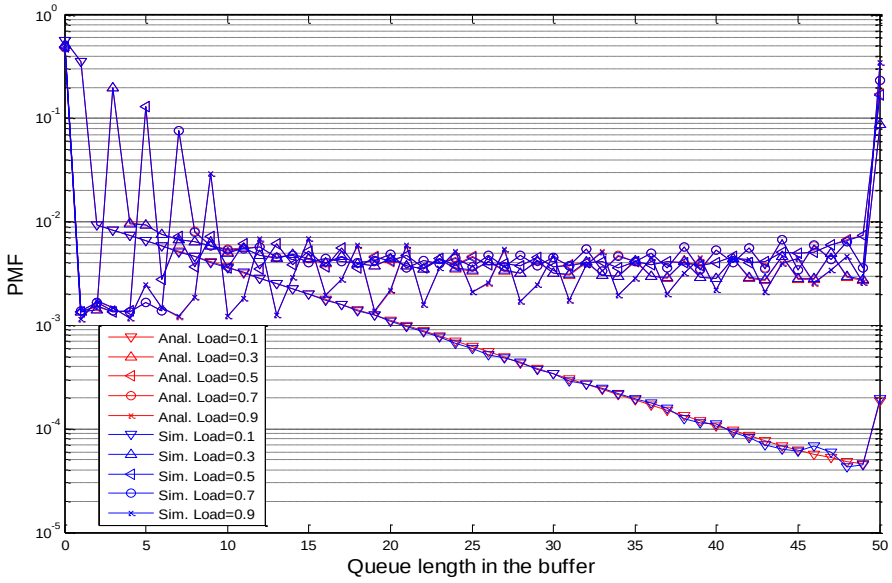

Figure 5.5 Validation for single on-off traffic model

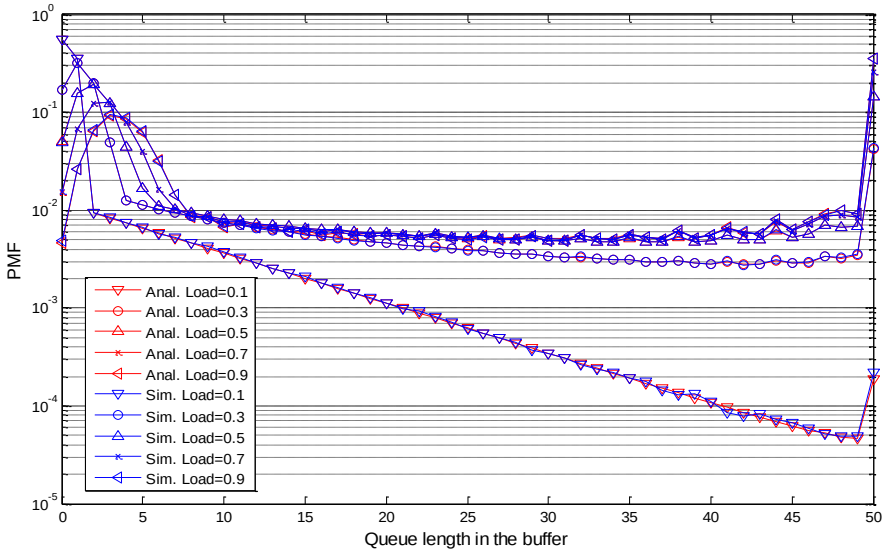

Figure 5.6 Validation for aggregated on-off traffic model

5.2.3.2 Queue state comparisons

Figure 5.7 and Figure 5.8 show the probability of queue state in the buffer for bursty arrivals and Poisson arrivals. We have the following observations:

1. One property in common for results with two different arrival processes is that the queue state probabilities are often increasing when reaching the buffer length K. One of the reasons is because the buffer is finite; thus, probability is accumulated at the tail. On the other hand, the behavior of cross-layer design in wireless environment is influenced by fading, and sometimes the channel state is so poor that we have to use MCSs with lower data rates, resulting in an increase in the probabilities for staying at longer queue lengths.

2. Unlike Poisson arrivals which often achieve the maximum value at the head when queue state=0, aggregated on-off arrivals usually feature a lower empty buffer probability, which is understandable because the buffer won't be empty as long as one single on-off stream is in the ON state. However, for single on-off arrival process, empty buffer probability usually achieves the maximum at the head.

3. The proposed cross-layer analysis using on-off arrival processes achieves a comparatively higher probability at the tail than the one using Poisson arrival with the same system loads, especially when system load is low (usually below 0.5).

The higher probability at the tail obtained from bursty arrivals are because of the bursty behavior of on-off traffic model (5.31). As a result, the probabilities for large queue length in the buffer gets higher for bursty arrivals since the continuous arrival of packets with a period of time fill the buffer faster than arriving packets with a constant arrival rate provided by Poisson arrival. This is of high importance for real network design.

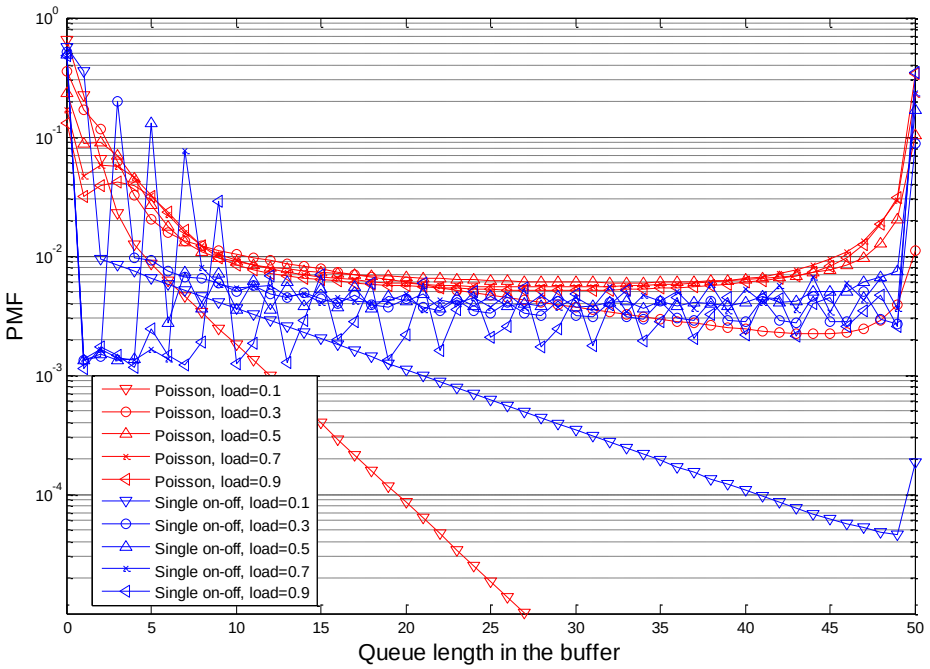

Figure 5.7 Queue state comparisons for single on-off and Poisson

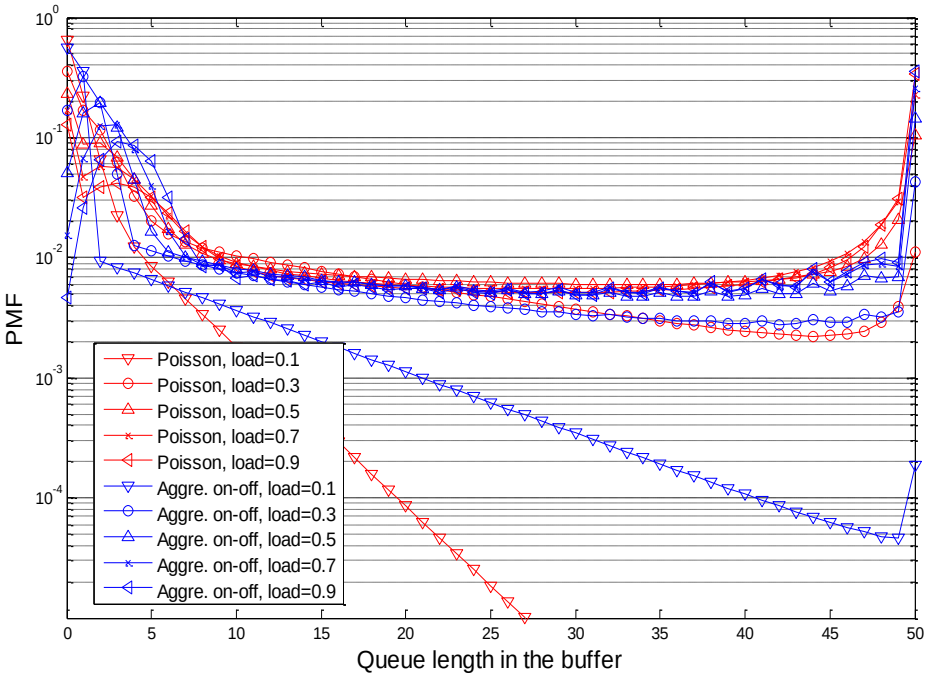

Figure 5.8 Queue state comparisons for aggregated on-off and Poisson

5.2.3.3 PDP comparisons

Figure 5.9 - Figure 5.12 show the PDP due to buffer overflow for bursty arrivals and Poisson arrivals with two transition matrix (5.31) and (5.32) under a range of buffer sizes K. We have the following observations:

1. The PDPs for the proposed cross-layer analysis with bursty traffic shown in Figure 5.9 and Figure 5.10, which approximate the bursty behavior for streams with single and aggregated on-off arrivals, are larger than the ones with Poisson arrivals with the same system load. The difference is much more apparent when system load is low from 10^{-1} to 10^{-3} absolute difference in probability values (at these buffer lengths).

However, the PDPs shown in Figure 5.11 and Figure 5.12 which doesn't capture the burstiness for streams with single and aggregated on-off arrivals show little difference when compared with Poisson arrivals.

3. The results also suggest that the traffic arrival process is highly important in determining the PDPs, even for a fading channel. Therefore, in order to obtain more accurate results which are more suitable for specific network system approximation, we should study the buffer performance by using a model that more accurately represents the traffic patterns.

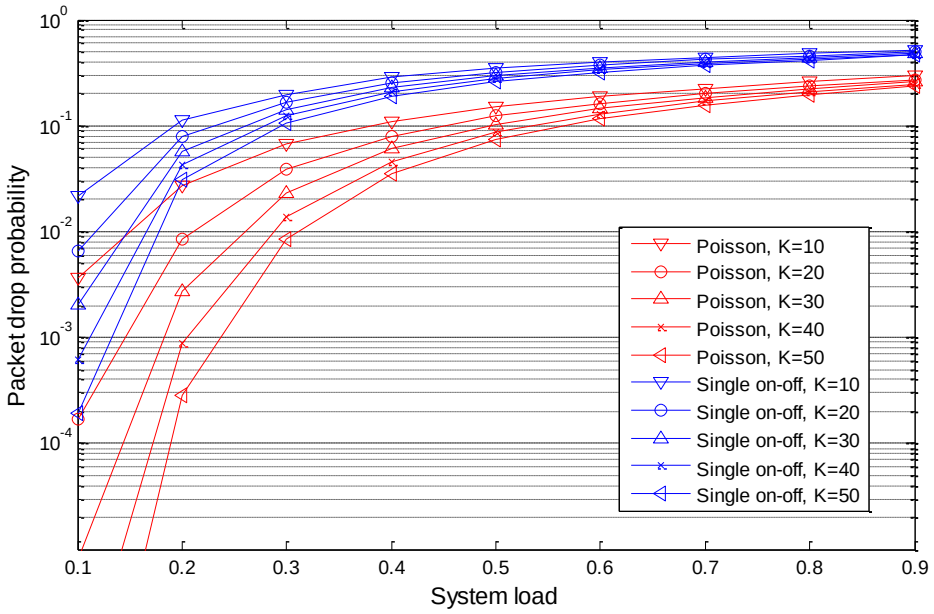

Figure 5.9 PDP comparisons for single on-off and Poisson using (5.31)

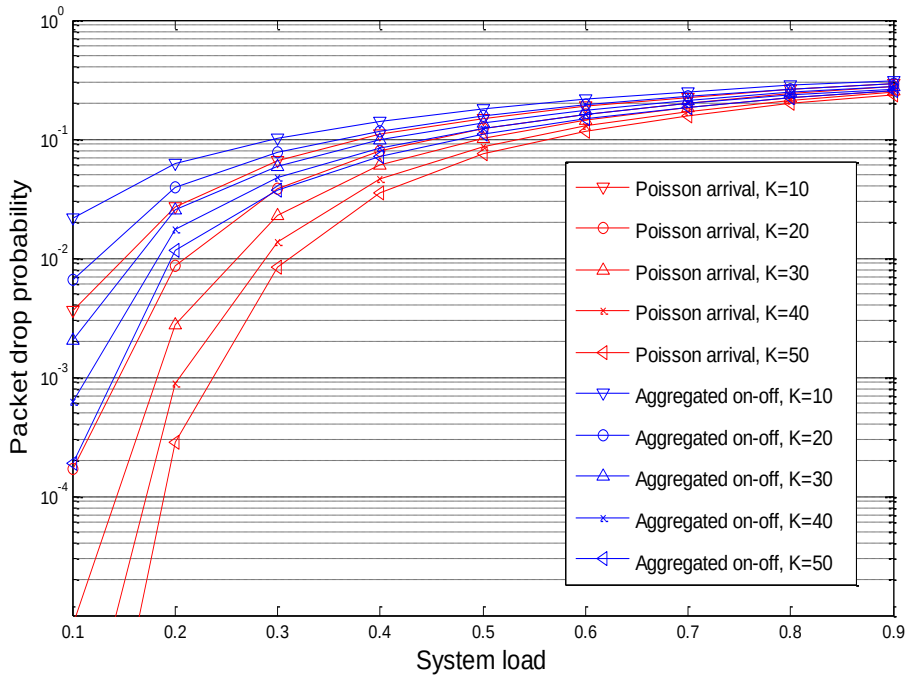

Figure 5.10 PDP comparisons for aggregated on-off and Poisson using (5.31)

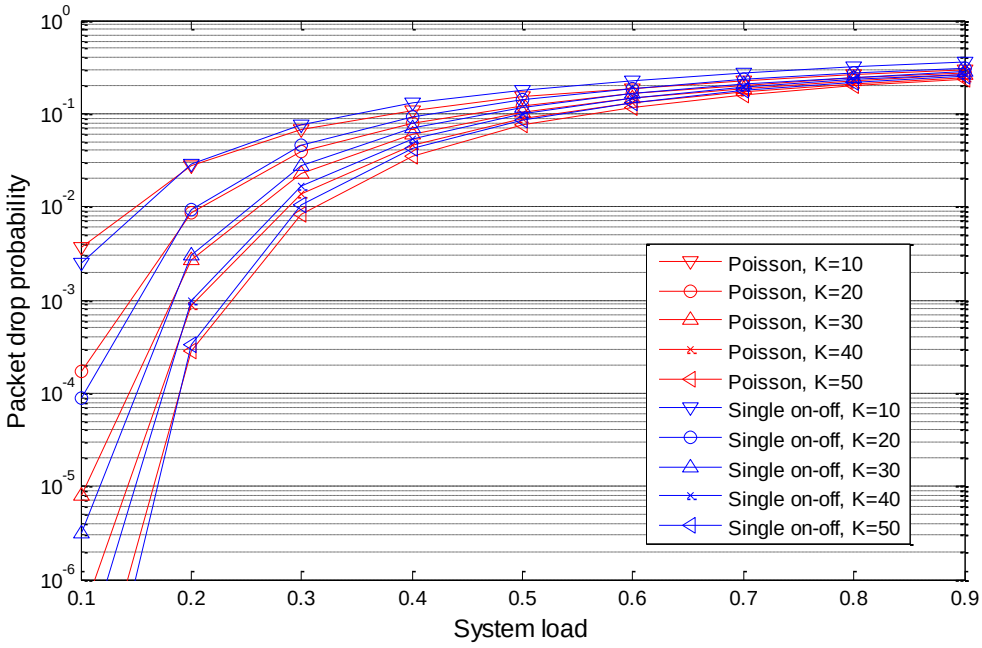

Figure 5.11 PDP comparisons for single on-off and Poisson using (5.32)

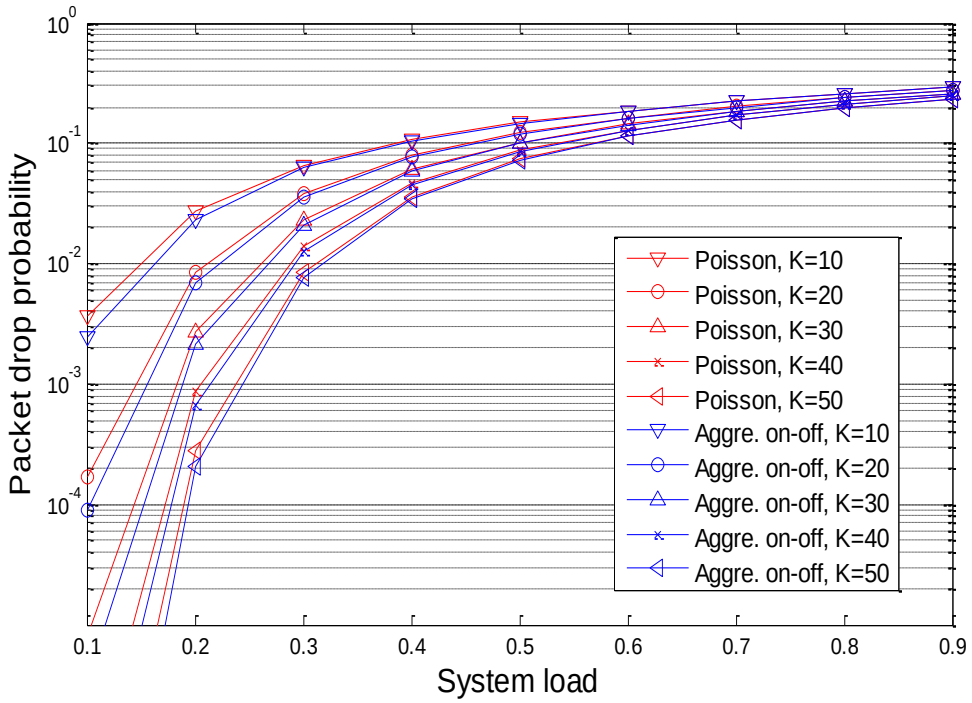

Figure 5.12 PDP comparisons for aggregated on-off and Poisson using (5.32)

5.2.3.4 Delay, PLP and throughput comparisons

Because we have known that single on-off arrival can be treated as a special case of the aggregated on-off arrivals model, and we have shown enough evidence on differences between single on-off arrival and Poisson arrival in terms of queue state and PDP, we only provide the comparisons between aggregated on-off arrival and Poisson arrival in this section. Figure 5.13 - Figure 5.15 illustrates the comparisons for bursty arrivals and Poisson arrivals in terms of average queueing delay, PLP and throughput. And we have the following observations:

1. The comparison for average queueing delay suggests that aggregated on-off arrival model results in higher delays compared to the Poisson arrival model, especially when loads are low. The results of queueing delay show gaps of over 1 slot duration in delay between adjacent curves among the same traffic model, which suggests a problem of choosing a proper buffer size for the cross-layer analysis to tradeoff between PDP due to buffer overflow and average queueing delay. However, this problem is out of this chapter's scope and we won't detail it.

2. The comparison for PLP suggests that the aggregated on-off arrival model results in

higher PLP compared to the Poisson arrival model. However, it is also important to note that when load $\rho = 0.1$, $\overline{PER}$ dominates the PLP for Poisson arrival; while PDP dominates the PLP in other cases with parameters set in this chapter. We can predict that $\overline{PER}$ dominates the PLP in more cases when buffer size K>50. And the minimum for PLP is approaching 6.14×10^{-4} but always larger than 6.14×10^{-4} with the MCSs adopted in Table 4.3.

3. The comparison for throughput suggests that the aggregated on-off arrival model achieves lower throughput compared to the Poisson arrival model especially when loads are high, as expected. By equation (5.30), we see that throughput is determined by PLP. The throughput difference becomes apparent (around 0.1 packet per slot in throughput between Poisson and aggregated on-off traffic models) when PLP is larger than 10^{-1}.

4. The comparisons shown by Figure 5.13 - Figure 5.15 also give evidence of the importance of selecting proper traffic models for different traffic patterns. The aggregated on-off arrival model is more appropriate than Poisson arrival model in bursty service approximations.

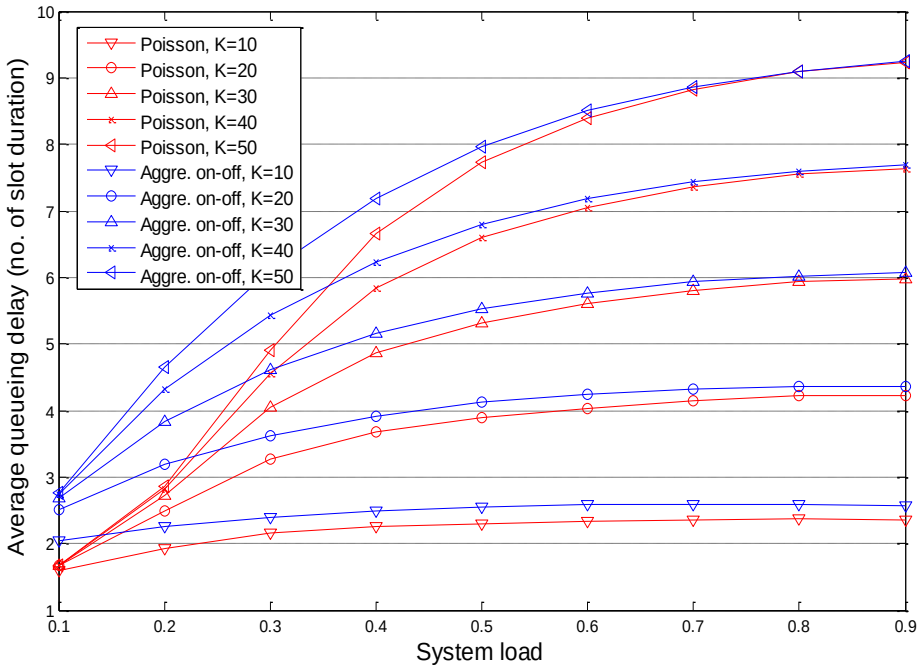

Figure 5.13 Delay comparisons for aggregated on-off and Poisson

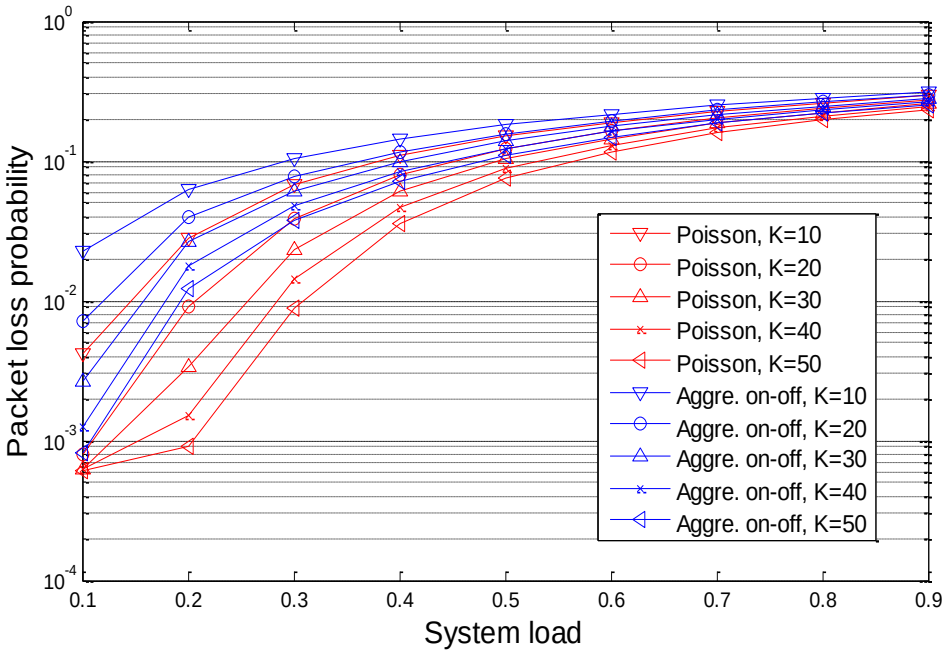

Figure 5.14 PLP comparisons for aggregated on-off and Poisson

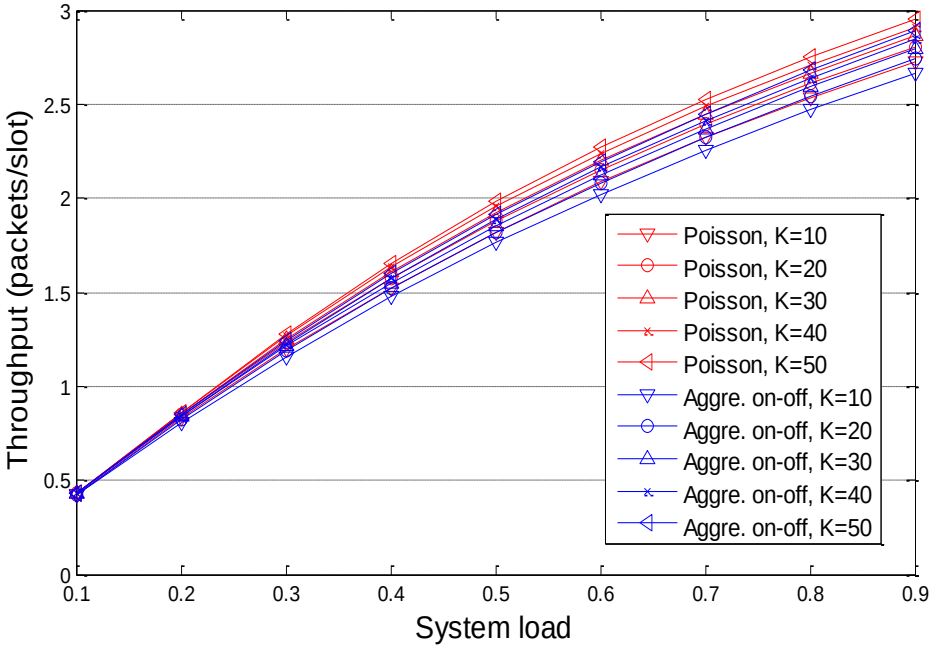

Figure 5.15 Throughput comparisons for aggregated on-off and Poisson

5.3 Summary

In this chapter, a new cross-layer analysis model combining bursty arrivals, including single on-off traffic model and aggregated on-off traffic model, and AMC with finite buffer for analyzing bursty services is proposed and analyzed. More specifically, we consider an end-to-end wireless link with single antenna at both transmitter and receiver side. At the transmitter side, the data-link layer packets with bursty arrivals are input into a queue with finite buffer size K. We analyze the queueing behaviors for bursty arrivals through the matrix geometric method. We obtain an enlarged FSMC for the proposed cross-layer analysis model and build up the transition matrix with a tuple of three parameters (arrival state, queue state and service state) accordingly. The queueing behaviors for burst arrivals are analyzed, and the results for performance analysis are compared with those generated by the usual Poisson traffic model.

Our numerical results clearly indicate that the traffic characteristics have a powerful effect on system performance. This means that future studies of wireless networks with fading channels must incorporate both the fading channel and a viable model (not just Poisson) of the packet arrival process(es) in order to achieve valid performance evaluation.

Chapter 6 Cross-layer Analysis for IEEE 802.11ac

In this chapter, we apply our proposed cross-layer analysis to a well-known WLAN standard – IEEE 802.11ac. Since the parameter settings of our proposed cross-layer analysis detailed in Chapter 4 and 5 only satisfies the specifications of 802.11a, we extend the existing cross-layer analysis, especially on the air interface part at the physical layer, to meet the requirements of the state-of-the-art. The system model with spatial multiplexing MIMO is introduced in this chapter, which is an extension of the cross-layer analysis with single channel as specified in Chapter 4 and 5. At the data-link layer, we use both Poisson arrivals and the bursty arrivals, which were introduced in Chapter 4 and Chapter 5 respectively, to approximate different traffic types. Similar to Chapters 4 and 5, we obtain the performance analysis on queue length in the buffer, PDP due to buffer overflow, average queueing delay, PLP and throughput (sub-channel throughput) through queueing analysis with the matrix geometric method after validating the system model.

6.1 Cross-layer analysis for IEEE 802.11ac

6.1.1 System model

We consider a point to point frequency flat Rayleigh fading SU-MIMO channel with M_t transmit antennas and M_r receive antennas. The channel is modeled as (6.1).

$$\boldsymbol{y} = \sqrt{E_S}\boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}, \quad (6.1)$$

Where E_S is the transmit energy per antenna; $\boldsymbol{y}$ is $M_r \times 1$ receive vector; $\boldsymbol{x}$ is $M_t \times 1$ receive vector with each element x_k selected from unit energy constellation; $\boldsymbol{n}$ is $M_r \times 1$ circular symmetric complex additive white Gaussian noise vector with variance $N_0\boldsymbol{I}_{M_r}$ where $\boldsymbol{I}_{M_r}$ is the identity matrix; $\boldsymbol{H}$ is the $M_r \times M_t$ channel gain matrix with each element h_{ij} the channel gain from the *j*th transmit antenna and *i*th receive antenna.

We adopt zero-forcing detector at the receiver side. The received $M_r \times 1$ symbol vector $\hat{\boldsymbol{y}}$ after pseudo-inverse operation is shown in (6.2),

$$\hat{\boldsymbol{y}} = \boldsymbol{G}\boldsymbol{y} = \boldsymbol{G}\left(\sqrt{E_S}\boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}\right) = \boldsymbol{x} + \boldsymbol{G}\boldsymbol{n}, \quad (6.2)$$

Where $\boldsymbol{G} = \frac{1}{\sqrt{E_S}}\boldsymbol{H}^{\dagger} = \frac{1}{\sqrt{E_S}}(\boldsymbol{H}^H\boldsymbol{H})^{-1}\boldsymbol{H}^H$ denotes the pseudo-inverse operation to channel gain matrix $\boldsymbol{H}$ so that $\sqrt{E_S}\boldsymbol{G}\boldsymbol{H} = \boldsymbol{I}$; $\boldsymbol{H}^H$ denotes the Hermitian transpose or conjugate transpose of channel gain matrix $\boldsymbol{H}$.

Therefore, the SNR on the *k*th spatial stream γ_k can be calculated by (6.3),

$$\gamma_k = \frac{p(x_k {x_k}^H)}{p([(\boldsymbol{Gn})(\boldsymbol{Gn})^H]_{kk})} = \frac{E_S}{N_0[\boldsymbol{H}^{\dagger}\boldsymbol{H}^{\dagger H}]_{kk}} = \frac{E_S}{N_0[(\boldsymbol{H}^H\boldsymbol{H})^{-1}]_{kk}} = \frac{\gamma_0}{[(\boldsymbol{H}^H\boldsymbol{H})^{-1}]_{kk}}, \quad (6.3)$$

Where $\gamma_0 = \frac{E_S}{N_0}$; $p(\cdot)$ is the symbol or noise power; $[\cdot]_{kk}$ is the *k*th diagonal entry of the matrix. Note that $p(x_k {x_k}^H) = 1$ since each symbol x_k is picked from a unit energy constellation. Basically, we can use (6.7) to obtain the instantaneous SNR for each spatial stream given the channel gain matrix $\boldsymbol{H}$. Furthermore, we can obtain the closed-form expressions for γ_k if we can identify the distribution of $[(\boldsymbol{H}^H\boldsymbol{H})^{-1}]_{kk}$.

Next, we adopt the Kronecker model as the spatial correlation model. Generally, the channel gain matrix $\boldsymbol{H}$ for the Kronecker model is obtained by (6.4) with Rayleigh fading,

$$\boldsymbol{H} = \boldsymbol{R}_r^{1/2}\boldsymbol{H}_w(\boldsymbol{R}_t^{1/2})^{\mathsf{T}}, \quad (6.4)$$

Where the elements of $\boldsymbol{H}_w$ are independently identical complex Gaussian random numbers, distributed with zero mean and unit variance; $\boldsymbol{R}_t$ is the transmit correlation matrix; $\boldsymbol{R}_r$ is the receive correlation matrix; T is the transpose of the matrix. As a result, the elements of channel gain matrix $\boldsymbol{H}$ follow a circular symmetric complex Gaussian distribution with covariance matrix $\boldsymbol{R}_t \otimes \boldsymbol{R}_r$ (6.5),

$$\boldsymbol{H} \sim CN(\boldsymbol{0}, \boldsymbol{R}_t \otimes \boldsymbol{R}_r), \quad (6.5)$$

Where $\otimes$ denotes the Kronecker product.

Based on the adoption of spatial multiplexing with zero-forcing detector and spatial correlation model, we introduce two SU-MIMO scenarios which are suitable for 802.11ac. The first scenario only assumes transmit correlations providing there is rich scattering at the receiver side which leads to no receive correlations; the second scenario assumes both transmit and receive correlations to obtain a more general model. Note that $M_t \leq M_r$. More specifically, we focus on a 4×4 MIMO model since

there are at most 4 antennas for the user to implement and at most 4 spatial streams, but the system model is ready to be extended to any other MIMO configurations.

We assume that AP antennas are not correlated with user antennas, which is indicated by the Kronecker model (6.4) in which transmit correlation matrix $\boldsymbol{R}_t$ and receive correlation matrix $\boldsymbol{R}_r$ are independent of each other. We also assume that the antennas between different users are not correlated since the real distances between users are usually large enough to ignore inter user antenna correlation. As a result, the Kronecker model can be reduced as (6.6) for the first scenario,

$$\boldsymbol{H} = \boldsymbol{H}_w(\boldsymbol{R}_t^{1/2})^{\mathsf{T}}, \text{ (6.6)}$$

Recall from (6.3) that the closed-form expressions for the SNR on the *k*th spatial stream γ_k can be obtained if we can identify the distribution of $[(\boldsymbol{H}^H\boldsymbol{H})^{-1}]_{kk}$. Since each row of the channel gain matrix $\boldsymbol{H}$ detailed in (6.6) follows an M_t-variate normal distribution with zero mean, $(\boldsymbol{H}^H\boldsymbol{H})$ is a complex Wishart matrix. Therefore, the SNR on the *k*th spatial stream γ_k with zero-forcing detectorcan be obtained by (6.7) which is given by [61],

$$f(\gamma_k) = \frac{[\boldsymbol{R}_t^{-1}]_{kk}e^{-\frac{\gamma_k[\boldsymbol{R}_t^{-1}]_{kk}}{\gamma_0}}}{\gamma_0\Gamma(M_r-M_t+1)}\left(\frac{\gamma_k[\boldsymbol{R}_t^{-1}]_{kk}}{\gamma_0}\right)^{M_r-M_t}, \text{ (6.7)}$$

However, for the second scenario, in which intra user antenna correlation should be taken into consideration, each row of the channel gain matrix $\boldsymbol{H}$ given by (6.4) does not follow a multi-variate normal distribution. Therefore, it is at least very hard, even if possible, to obtain the closed-form expressions for the SNR on the *k*th spatial stream γ_k given by (6.3). Therefore, we instead construct and verify the flat fading Rayleigh fading channels with transmit and receive correlation, and capture the distributions of γ_k by using Monte-Carlo simulation for the second scenario. Since the distributions of γ_k are dependent on the values of $\boldsymbol{R}_t$ and $\boldsymbol{R}_r$, the results will be given in section 6.1.3 with specific $\boldsymbol{R}_t$ and $\boldsymbol{R}_r$ for the second scenario.

Note that all equations given in this section lay the foundation of the queueing analysis on 802.11ac, which is the main contribution of this chapter, for the next section.

6.1.2 Queueing analysis

Once the distributions of γ_k are determined, we can obtain the stationary probability π_n^k for the kth stream to be in mode n given by (6.8) which is similar to (4.6),

$$\pi_n^k = \int_{\gamma_n}^{\gamma_{n+1}} f(\gamma_k)\, d\gamma_k, \quad (6.8)$$

Where $f(\gamma_k)$ is detailed by (6.7); $\{\gamma_n\}$ is the SNR threshold which is determined by (4.1)-(4.2). Note that because we need to meet the specifications of 802.11ac with higher order MCSs and OFDM operation, the original parameter settings for MCSs specified in Table 4.3 is no longer applied; instead, we determine new SNR thresholds for MCSs chosen from Table 2.2. Since different settings for the data-link level simulation detailed in Figure 1.2 result in different SNR threshold values, and it often takes quite a long time to get the PER curve for each MCS through simulation, we use one specific parameter settings for PER analysis and further SNR threshold determination, which is given by Table 6.1, throughout this chapter. In fact, we can obtain SNR threshold for any reasonable system parameter settings (see Appendix A).

Table 6.1 Parameter settings for PER analysis

Parameter	Value	Parameter	Value
Packet length	1080 bits	Symbols per frame	2160
SNR range	-5~35 dB	Channel bandwidth	20 MHz
No. of data subcarriers	52	No. of FFT points	64
Bits/symbol	[1,2,4,6,8]	Interleaving block	$(2 \times bits/sym) \times 26$
Numerical system	Grey code	Modulation	Rectangular QAM
OFDM symbol duration	3.2 μs	GI	800 ns

Then we select 5 MCSs (for details, see Appendix A) from MCS 0~9, the chosen MCSs together with their parameter settings are listed in Table 6.2,

Table 6.2 Parameter settings for 802.11ac MCSs

Mode n	n=0	n=1	n=2	n=3	n=4	n=5
MCS Index	-	0	2	4	6	8
Modulation	-	BPSK	QPSK	16QAM	64QAM	256QAM
Coding rate	-	1/2	3/4	3/4	3/4	3/4
Coded bits per symbol	0	1	2	4	6	8
Data bits per symbol	0	0.5	1.5	3	4.5	6
Service rate (packets/slot)	0	1	3	6	9	12
a_n	0	2.898	2.690	2.973	2.934	2.999
g_n	0	0.7383	0.2041	0.04633	0.01158	0.003091
$\gamma_{pn}(dB)$	-	1.5872	6.8559	13.7139	19.6825	25.6085

Based on the parameter settings given in Table 6.2, we can obtain SNR thresholds $\{\gamma_n\}$ from equations (4.1)-(4.2). Together with (6.7), we can obtain π_n^k given by (6.8).

Based on π_n^k, we can obtain the probability transition matrix for the *k*th streams service rate $\boldsymbol{P}^k$. Since we assume a slow Rayleigh fading channel for each stream, the service rate can only be changed between adjacent modes. Then we can build up the probability transition matrix similar to the one given by Equation (4.10). We can calculate the transition probabilities of the *k*th stream $\{p_{n,n-1}^k\}$ and $\{p_{n,n+1}^k\}$ using (6.9), which is similar to equation (4.8) for N=5 (since we choose 5 MCSs for queueing analysis),

$$\begin{cases} p_{n,n+1}^k = \frac{N_{n+1} \times T_f}{\pi_n^k}, if\ n \in [0, N-1], n \in Z \\ p_{n,n-1}^k = \frac{N_n \times T_f}{\pi_n^k}, if\ n \in [1, N], n \in Z \end{cases}, \quad (6.9)$$

Where π_n^k is obtained by (6.8); T_f is calculated by (6.10),

$$T_f = T_s \times ceil(\frac{Symbols\ per\ frame}{Number\ of\ data\ subcarriers}), \quad (6.10)$$

Where $T_s = OFDM\ symbol\ duration + GI\ duration = 4\ \mu s$, $ceil(\cdot)$ is the ceiling function. Other parameters can be obtained in Table 6.1.

Therefore, the remaining problem for (6.9) is that of obtaining the level crossing rate (LCR) for each mode $\{N_n, n \in [1, N], n \in Z\}$. The LCR is the value quantifying the frequency of the envelope of SNR going across some chosen thresholds in either a positive or a negative direction when using the fading channels. In our case the thresholds are the obtained SNR thresholds $\{\gamma_n\}$ detailed above. The usual method to obtain the LCR is to check the envelope crossing a given level R in the positive direction (6.11) given by [67],

$$N_R = \int_0^\infty \dot{r}_k f(\dot{r}_k, r_k = R) d\dot{r}_k = \int_0^\infty \dot{r}_k f(\dot{r}_k | r_k = R) f(r_k = R) d\dot{r}_k, \quad (6.11)$$

Where $\dot{r}$ is the derivative of r with respect to time. And $f(r = R)$, which is given by (6.12), can be obtained by (6.7) with the transformation $f(r_k)|dr_k| = f(\gamma_k)|d\gamma_k|$ and $\gamma_k = \frac{r_k^2}{2}$ for the first scenario,

$$f(r_k) = \frac{r_k}{\gamma_0/[\boldsymbol{R}_t^{-1}]_{kk} \times \Gamma(M_r - M_t + 1)} e^{-\frac{r_k^2}{2\gamma_0/[\boldsymbol{R}_t^{-1}]_{kk}}} \left(\frac{r_k^2}{2\gamma_0/[\boldsymbol{R}_t^{-1}]_{kk}}\right)^{M_r - M_t}, \quad (6.12)$$

However, for the rest of the integral $\dot{r}_k f(\dot{r}_k | r_k = R)$ in (6.16), we are not able to find a way to solve it in the presence of transmit (and receive) correlation, because we cannot identify the distribution of $\dot{r}_k$, and neither can we find a possible solution in the literature with similar scenario settings. Through simulation, we find that zero-forcing detecting operation given by (6.2) has changed the LCRs, although the distributions for r_k remains quite similar to Rayleigh fading ones when $M_r = M_t$. As a result, we use Monte-Carlo simulation to determine LCR with transmit (and receive) correlations.

All of the above are the major changes to the service process to meet the specifications of 802.11ac; while all the other queueing analysis, which has little changes compared with 802.11ac, including: 1. arrival process with Poisson and bursty traffic models, 2. Queue state transition analysis and 3. Performance analyses on PDP, delay, PLP and throughput, have been detailed in Chapter 4 and 5.

6.1.3 Numerical results

In this section, we present numerical results for the probabilities associated with the queue state in the buffer, the PDP due to buffer overflow, the average queueing delay,

the PLP and the throughput. We study the proposed cross-layer analysis with both Poisson and bursty traffic models. We also make comparisons between the scenarios with and without receive antenna correlations.

Similar to the parameter settings in Chapter 4 & 5, we assume a frame duration of $T_f = 2$ ms and a prescribed PER $P_0 = 10^{-2}$.

We set the system average SNR=35 dB for each spatial stream, compared to 15 dB set in Chapter 4 & 5. There is a huge increase in average SNR. After analysis, we get three possible reasons for the increment of average SNR:

1. We use higher-order MCS, that is, 256 QAM with 3/4 coding rate, to increase the maximum data rate, which requires better channel conditions compared to MCSs adopted in Chapter 4 & 5.

2. We insert 800 ns GI for each OFDM symbol. The insertion of GI results in the increase in SNR in order to obtain the same PER.

3. The trace-back length for the Viterbi decoder we use for simulation is quite short. We set trace-back length = 35 because conventionally the trace-back length is 5 times longer than the constraint length, which has been set to 7, used by a 1/2 convolutional encoder. However, the trace-back length has a strong impact on the PER. The longer the trace-back length is, the smaller the PER will be. But longer trace-back length also results in larger registers to store the path metrics used by the Viterbi decoder. As a result, a longer trace-back length takes more time for a simulation of PER. The trace-back length we use might be too short compared to the one used by real devices. Therefore, we have to increase SNR to obtain the same PER.

The next problem is to select the proper buffer length used for analysis and simulation. There is not so much information on buffer sizing in published work, although some inspiring attempt has been made in literature [71]-[73], the problem still remains open for investigation. However, the selection of the proper buffer size is out of this thesis's scope, and we just try to find out the buffer size in use by simulation tools and testbeds. As suggested by [71], ns-2 simulator uses a default buffer size of 50 packets for the queue object (which is also why papers like [22] uses 50 packets as the buffer size, and

we follow this setting in Chapter 4 & 5); while the open source MadWifi drivers for the Atheros chipset adopts 200 packets for the buffer size; and the ath5k drivers equally divide 200 packets buffer size into four queues representing four different traffic types; the ath9k drivers uses 512 instead 200 packets for similar division for 802.11n standard. To offset the differences in these buffer size settings, we adopt a range of buffer lengths, from 50, 100, 150, 200 to 250 packets.

Next, the proper transmit and receive correlation matrices $\boldsymbol{R}_t$ and $\boldsymbol{R}_r$ need to be adopted. Consistent with those adopted in [62], we set $\boldsymbol{R}_t$ and $\boldsymbol{R}_r$ using (6.13). Note that the correlation matrices are not necessarily the same, and the elements for correlation matrices are also dependent on real antenna configurations, which may vary in different scenarios.

$$\boldsymbol{R}_t = \boldsymbol{R}_r = \begin{pmatrix} 1 & 0.57e^{-2.25i} & 0.17e^{0.02i} & 0.29e^{-2.94i} \\ 0.57e^{2.25i} & 1 & 0.57e^{-2.25i} & 0.17e^{0.02i} \\ 0.17e^{-0.02i} & 0.57e^{2.25i} & 1 & 0.57e^{-2.25i} \\ 0.29e^{2.94i} & 0.17e^{-0.02i} & 0.57e^{2.25i} & 1 \end{pmatrix}, \quad (6.13)$$

Since the correlation matrices specified by (6.13) are symmetric, we obtain that the performances for stream 1 & 4 are the same; while the performances for stream 2 & 3 are the same. Therefore, we only need to check stream 1 and stream 2 for simplicity.

Based on correlation matrices, we can obtain the average service rate $\overline{S_t}$ for our chosen channel model. Table 6.3 lists the average service rate for the two scenarios.

Table 6.3 Average service rate for 802.11ac MCSs

	Scenario 1	Scenario 2
Stream 1	8.80 packets/slot	8.10 packets/slot
Stream 2	8.08 packets/slot	7.35 packets/slot

Note that the average service rates for scenario 1 when there is only transmit correlation are larger than ones for scenario 2 when both transmit and receive correlations exist with the same average SNR for each spatial stream. Thus, the

existence of correlation degrades the channel conditions, which can also be verified by equation (6.7) where the *k*th diagonal entry $[\boldsymbol{R}_t^{-1}]_{kk}$ is a real number larger than 1.

After we obtain the average service rates for the queueing analysis, the arrival rates for both Poisson and bursty traffic models should be identified for various system loads. Poisson arrival rates can be easily adjusted to the specified average service rates to obtain different system loads; while bursty arrival rates need more manipulations. If we use the probability transition matrix for each single on-off traffic stream specified by equation (5.31), for which we still adopt in this section to approximate the burstiness for arrival traffic, we obtain the single on-off stream arrival rate=0.4303 packet/slot. Therefore, we get several definite values of system loads for analytical simplicity, although we can obtain any arbitrary system load by changing the elements in equation (5.31).

6.1.3.1 SNR distribution validation for each stream

Before performance analysis, we validate the SNR distribution for scenarios 1 & 2. The analytical expression for the SNR distribution for scenario 1 has been given by equation (6.7), and Figure 6.1 has given the CDF for SNR distribution by both theoretical analysis and simulation. From Figure 6.1, we obtain that streams 1 & 4 have the same CDF for SNR distribution; while stream 2 & 3 have the same CDF. Moreover, analytical and simulation results match quite well, which validates the SU-MIMO channel we built. In addition, the matching results prove that using Monte-Carlo simulation for SU-MIMO channel construction is feasible in practice. Therefore, we use Monte-Carlo simulation to generate a CDF for the SNR distribution for scenario 2 for which we are not able to obtain an analytical expression.

Figure 6.2 shows the simulation results for each stream. Least square regression analysis with exponential fitting is used for obtaining an asymptotic expression of the CDF for the SNR distribution. The reason we use exponential fitting is because simulation results are shaped similar to the CDF of an exponential distribution, and an exponential expression is easier to take into further operations such as integration for the calculation of average PER.

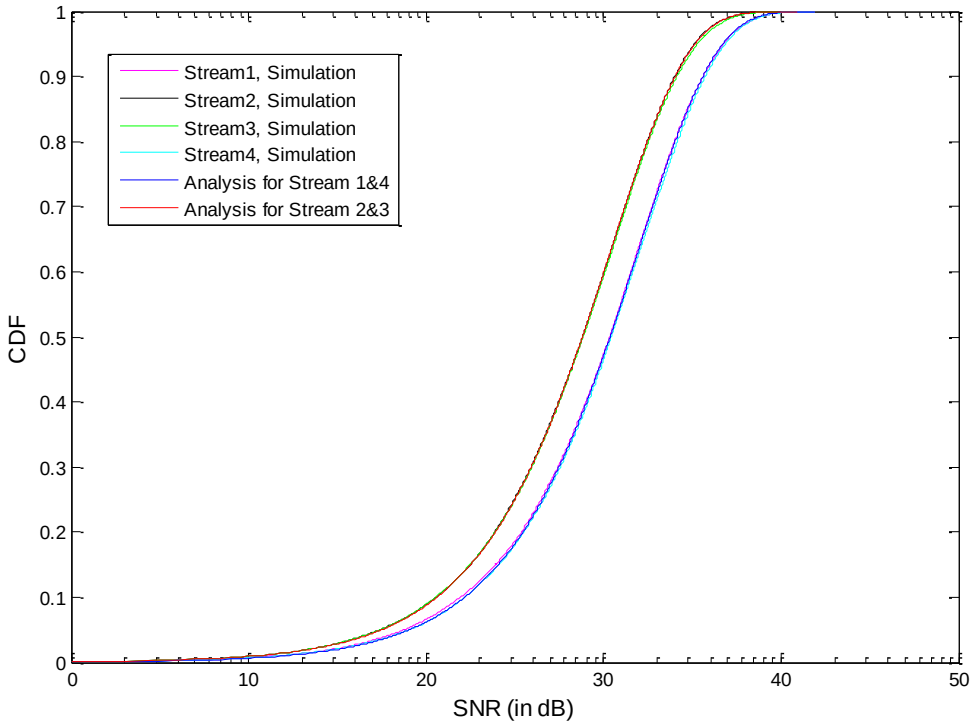

Figure 6.1 CDF of SNR for each stream for scenario 1

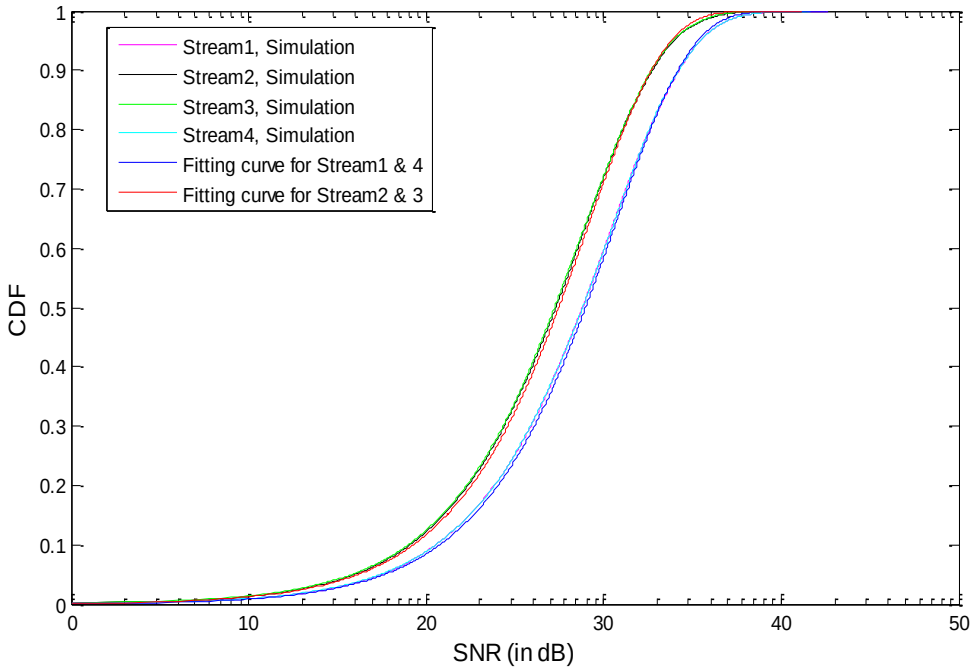

Figure 6.2 CDF of SNR for each stream for scenario 2

6.1.3.2 Queue state in the buffer

Figure 6.3-6.5 show the probabilities associated with queue state in the buffer for both Poisson and bursty arrivals for scenario 1 & 2 when the buffer size K=100. System load = 0.7 for Poisson arrival in both scenario 1 &2; while system load = 0.3409, 0.3713, 0.3713, 0.3409 for bursty arrival of each stream in scenario 1 and system load = 0.3740, 0.4082, 0.4082, 0.3740 in scenario 2. We have the following observations:

1. For Figure 6.3 and 6.4, we see that the shapes of curves with different average service rates are also different even if they share the same system load, which shows the impact of fading on the queue models.

2. For Figure 6.5, we see that different scenarios can share the same curves with different average service rates if we do not change the elements of the probability transition matrix for each single on-off traffic stream specified by equation (5.31), which shows the impact of the burstiness of the traffic on the queue models.

Note that other performance analysis are easily validated after we validate the queue state in the buffer, thus, we won't validate other performance metrics. In addition, since scenario 2 (considering receive correlations) is more general than scenario 1, the results for scenario 1 are mainly used for comparisons with ones for scenario 2 to avoid unnecessary repetitions.

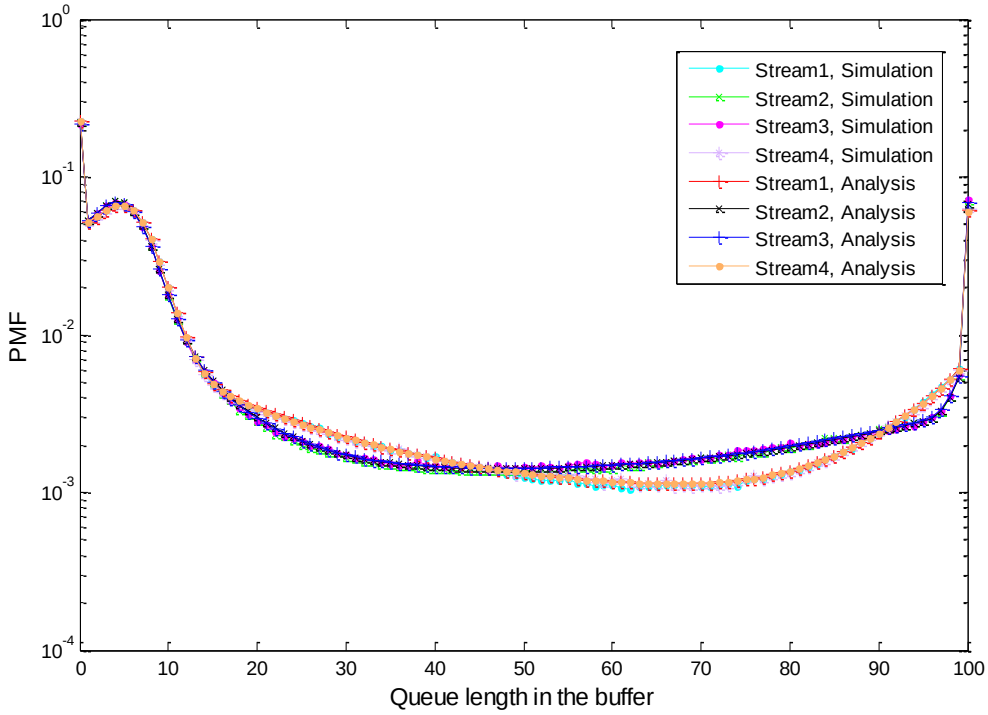

Figure 6.3 Queue state in the buffer with Poisson arrival for scenario 1

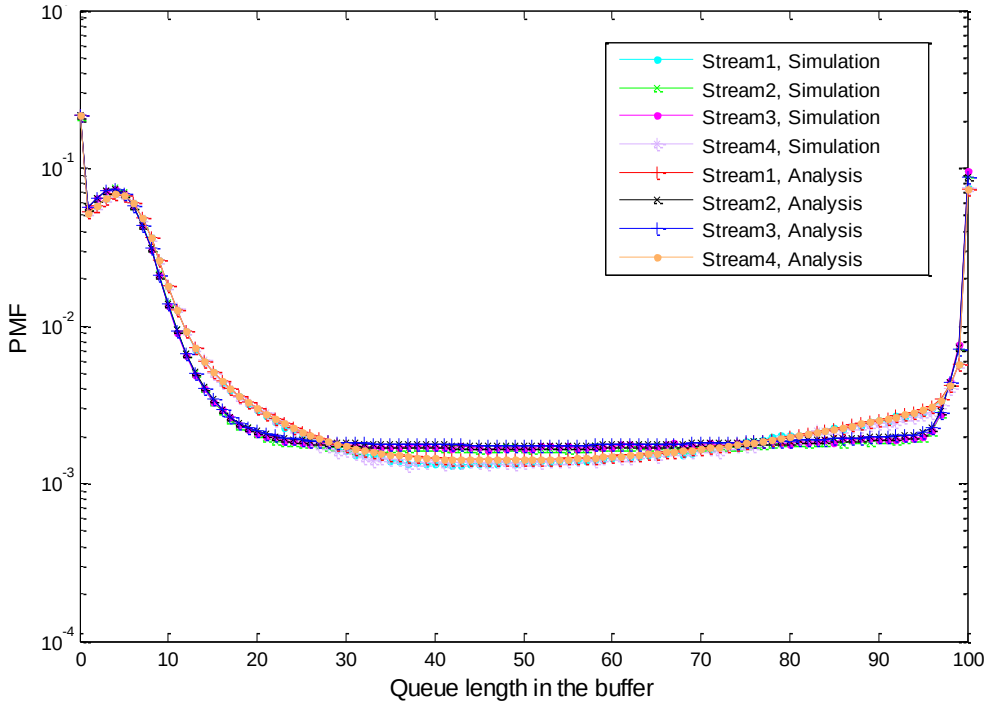

Figure 6.4 Queue state in the buffer with Poisson arrival for scenario 2

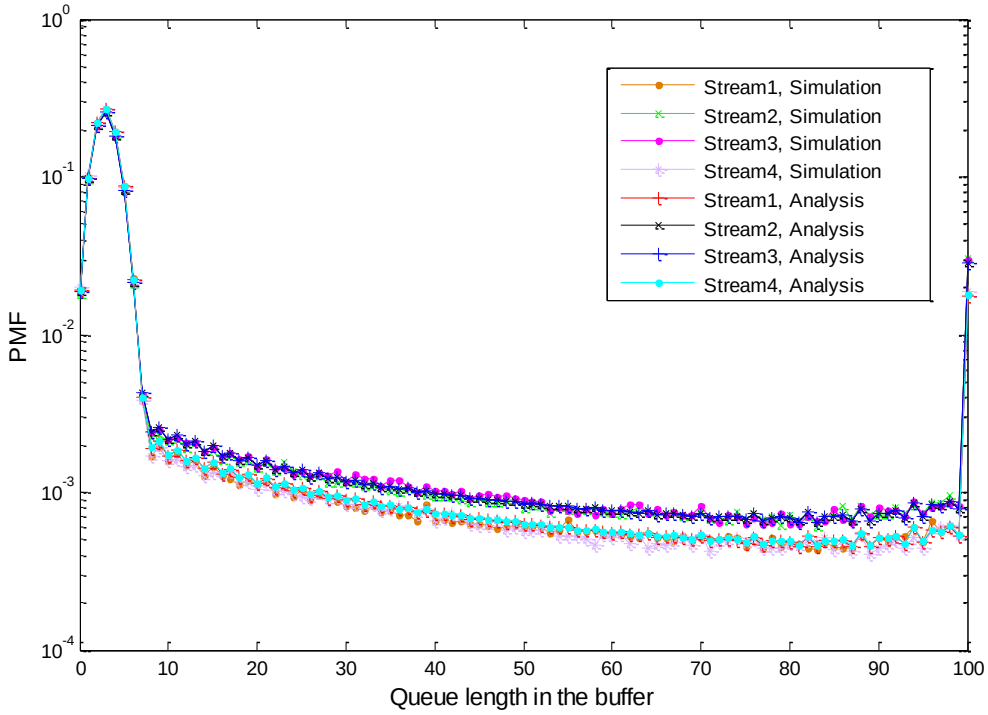

Figure 6.5 Queue state in the buffer with bursty arrival

6.1.3.3 PDP due to buffer overflow

Figure 6.6-6.9 captures the behaviors of PDP due to buffer overflow with both Poisson and bursty arrivals. We have following observations:

1. Figure 6.6 and 6.7 shows the comparisons for streams with different average service rates. We see that the stream with larger average service rate (stream 1 in our case) always achieves a lower PDP with the same system load and buffer size, which shows the impact of fading on the queue models.

2. Figure 6.8 and 6.9 illustrates the comparisons on the same stream for different scenarios. We find that the stream without receive correlations always achieves a lower PDP with the same system load and buffer size. This situation is understandable because streams without receive correlations always obtain higher average service rates under the same average SNR, so the same conclusion can be drawn as in observation 1. The comparisons illustrate the impact of antenna correlations.

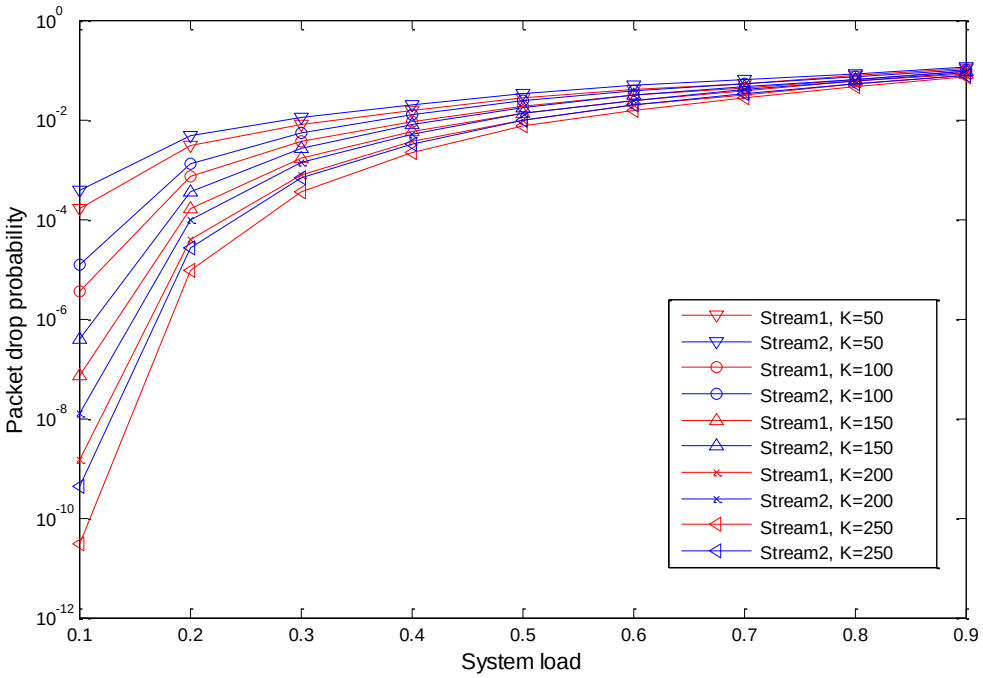

Figure 6.6 PDP with Poisson arrival for scenario 2

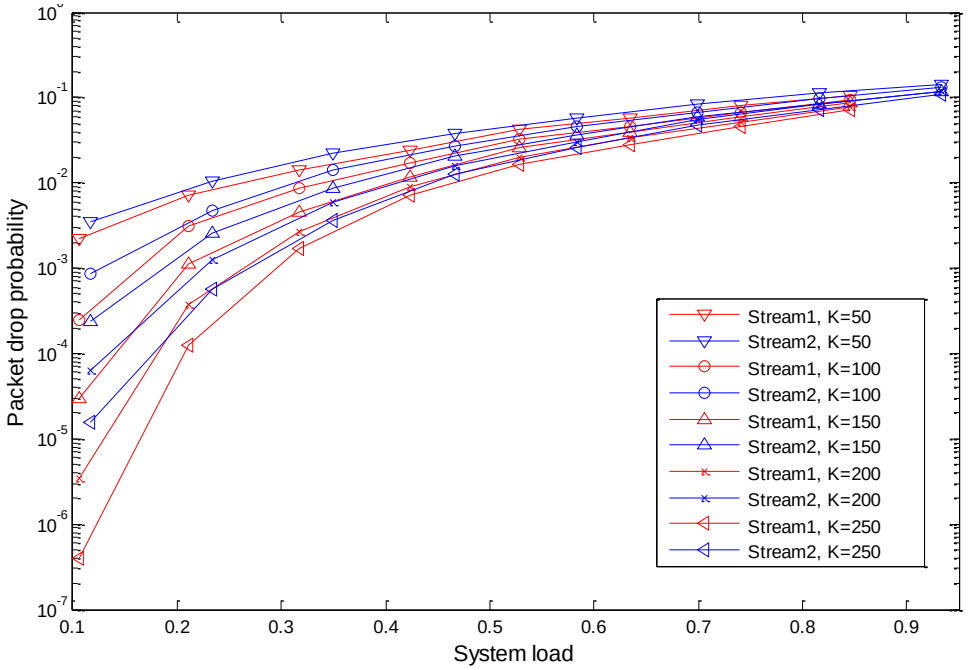

Figure 6.7 PDP with bursty arrival for scenario 2

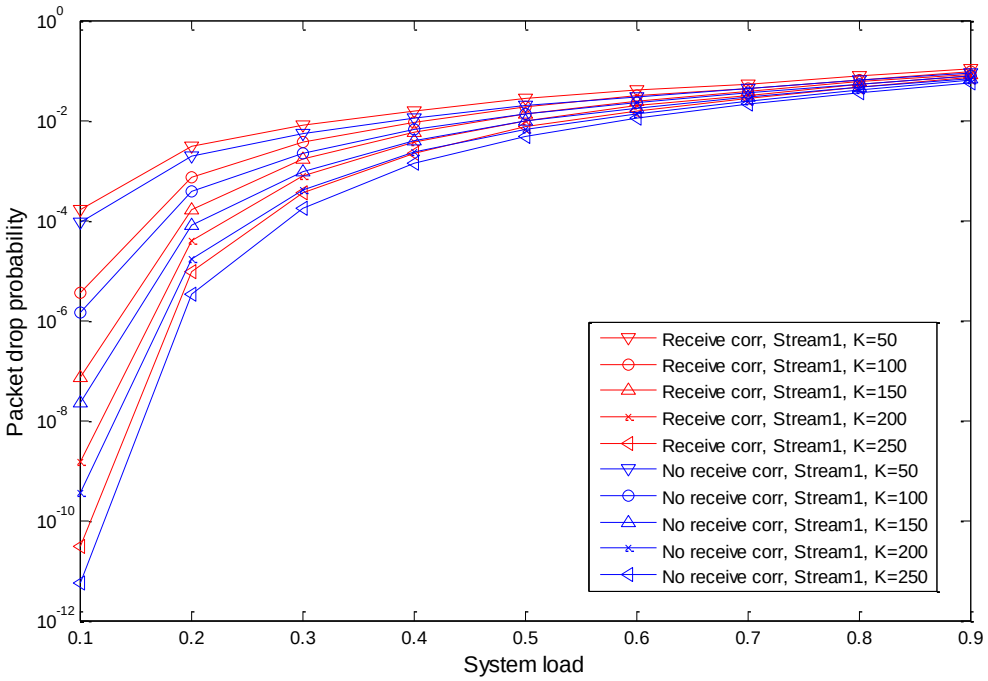

Figure 6.8 PDP comparisons with Poisson arrival

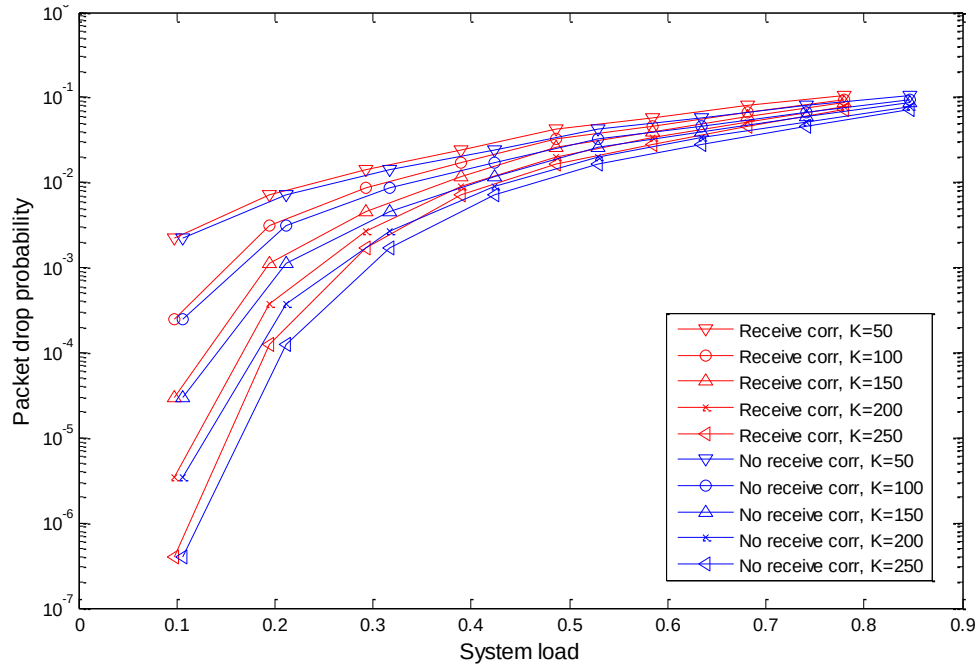

Figure 6.9 PDP comparisons with bursty arrival

6.1.3.4 Average queueing delay

Figure 6.10-6.13 shows the analysis of average queueing delay with both Poisson and bursty arrivals. We have the following observations:

1. Figure 6.10 and 6.11 compare streams with different average service rates in the presence of transmit and receive correlations. We see that the stream with lower average service rate (stream 2 in our case) always has a larger average queueing delay with the same average SNR for each stream, which shows the impact of fading on the queue models.

2. Figure 6.12 and 6.13 show the comparisons on the same stream for different scenarios. We see that the stream with receive correlations always results in a larger average queueing delay with the same system load, buffer size and average SNR. This is also because the stream with receive correlations has lower average service rates compared to the stream without receive correlations, which shows the impact of antenna correlations on queue models.

3. We do not analyze the case when system load grows larger than 0.9; however, we can still predict the curves for average queueing delay become flat because of the finite buffer sizes.

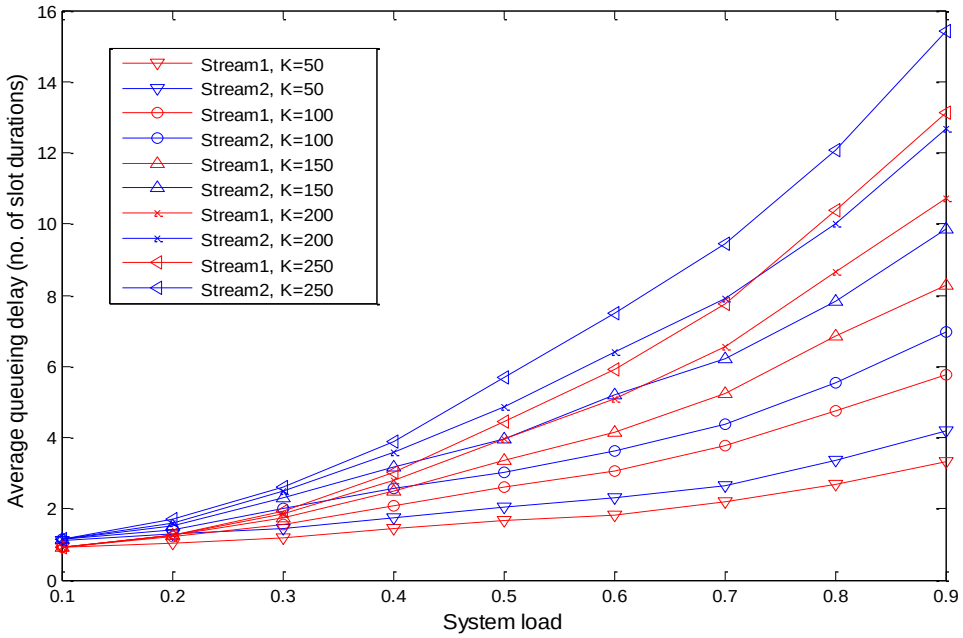

Figure 6.10 Average queueing delay with Poisson arrival for scenario 2

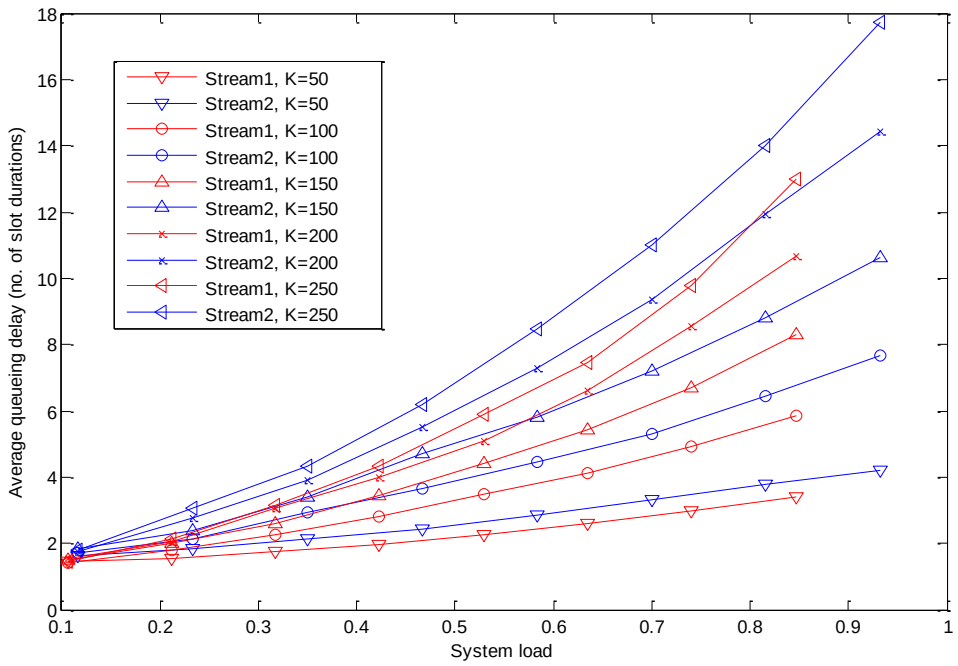

Figure 6.11 Average queueing delay with bursty arrival for scenario 2

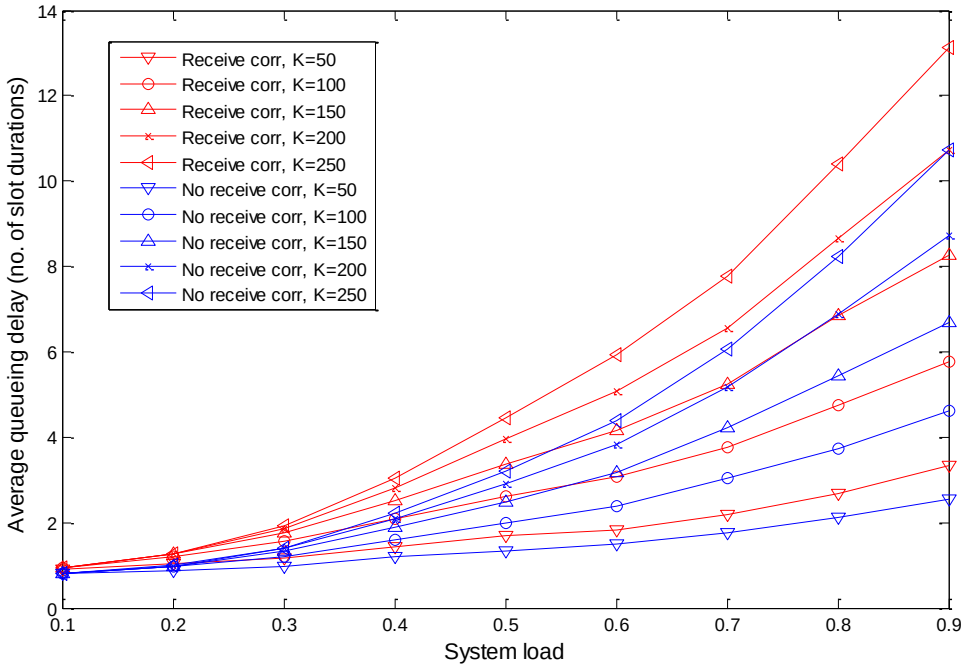

Figure 6.12 Average queueing delay comparisons with Poisson arrival

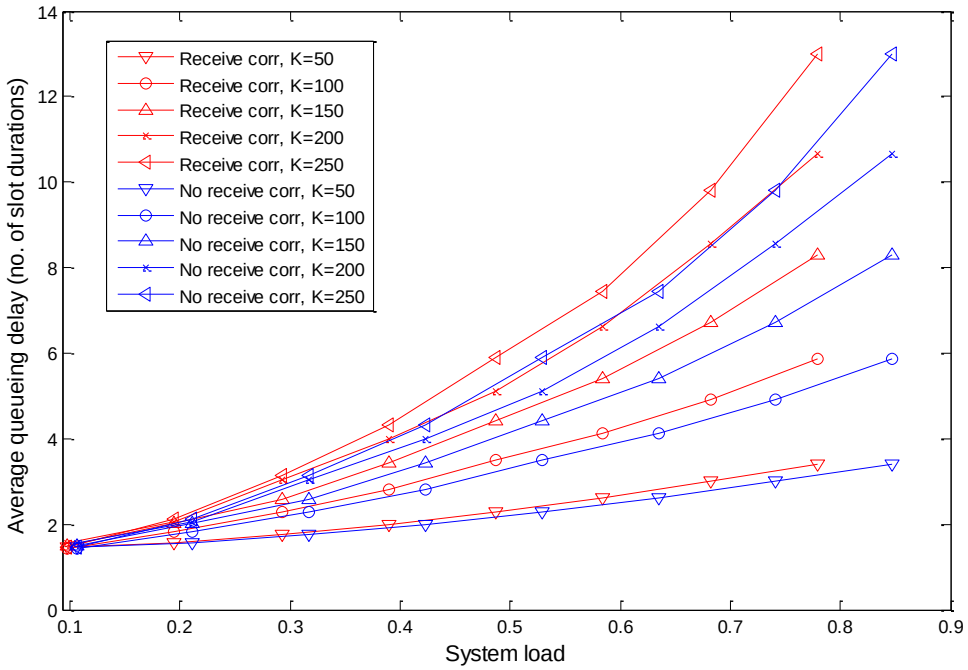

Figure 6.13 Average queueing delay comparisons with bursty arrival

6.1.3.5 PLP

Figure 6.14-6.17 illustrate the PLP for the cross-layer analysis for both Poisson and bursty arrivals. The average PERs for streams 1 & 2 are equal to 2.2996×10^{-6} *and* 3.5909×10^{-6} respectively for scenario1; while they are equal to 3.4587×10^{-6} *and* 5.3616×10^{-6} for scenario 2. We have following observations:

1. As analyzed in Chapter 4 and 5, PLP is determined by PDP and average PER by equation (4.28). In this case, PLP is dominated by PDP when PDP is larger than 10^{-5}; while PLP is dominated by average PER when PDP is lower than 10^{-6}. Therefore, PLPs will converge to the values of average PERs when PDPs are lower than 10^{-6}.

2. Figure 6.14 and 6.15 show the PLP with different average service rates. The stream with higher average service rate achieves lower PLP, again illustrating the impact of fading on queue models.

3. Figure 6.16 and 6.17 show the comparisons on PLP for different scenarios. The stream without receive correlations gives lower PLP, which again verifies the impact of antenna correlations on queue models.

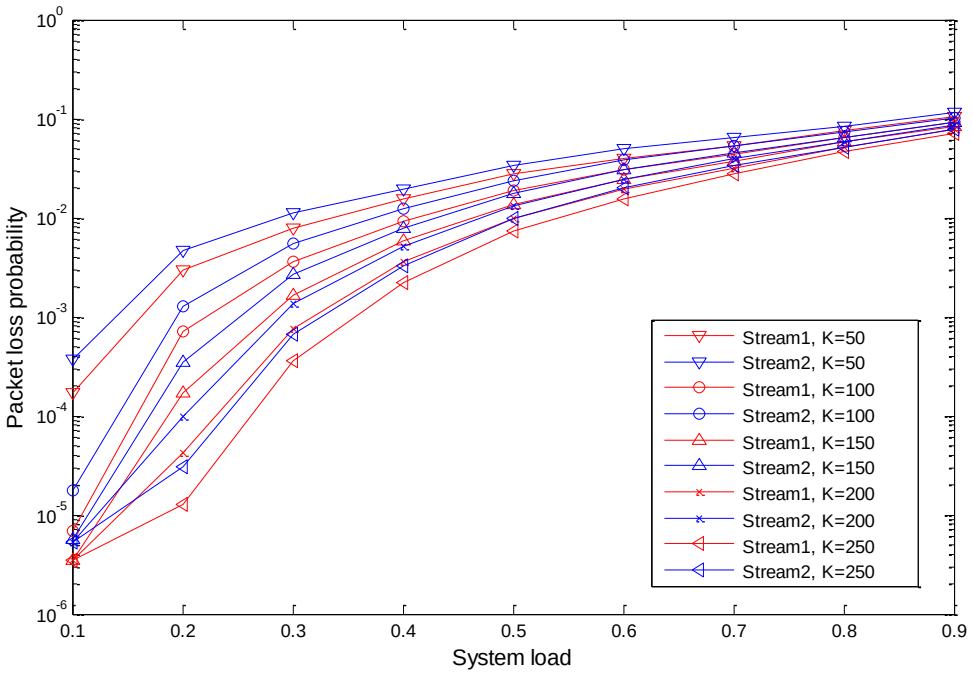

Figure 6.14 PLP with Poisson arrival for scenario 2

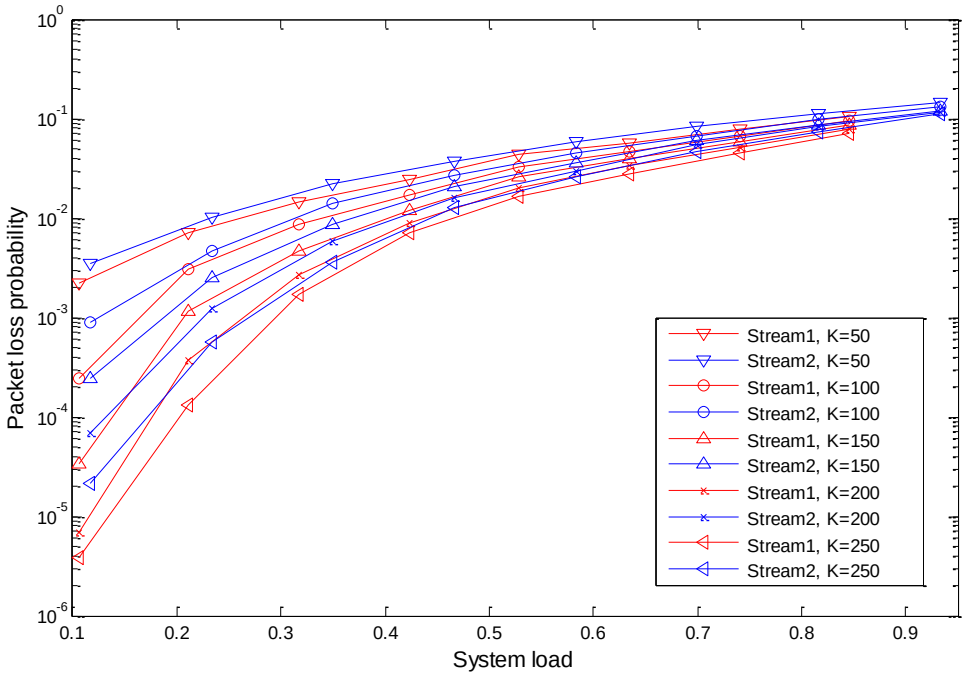

Figure 6.15 PLP with bursty arrival for scenario 2

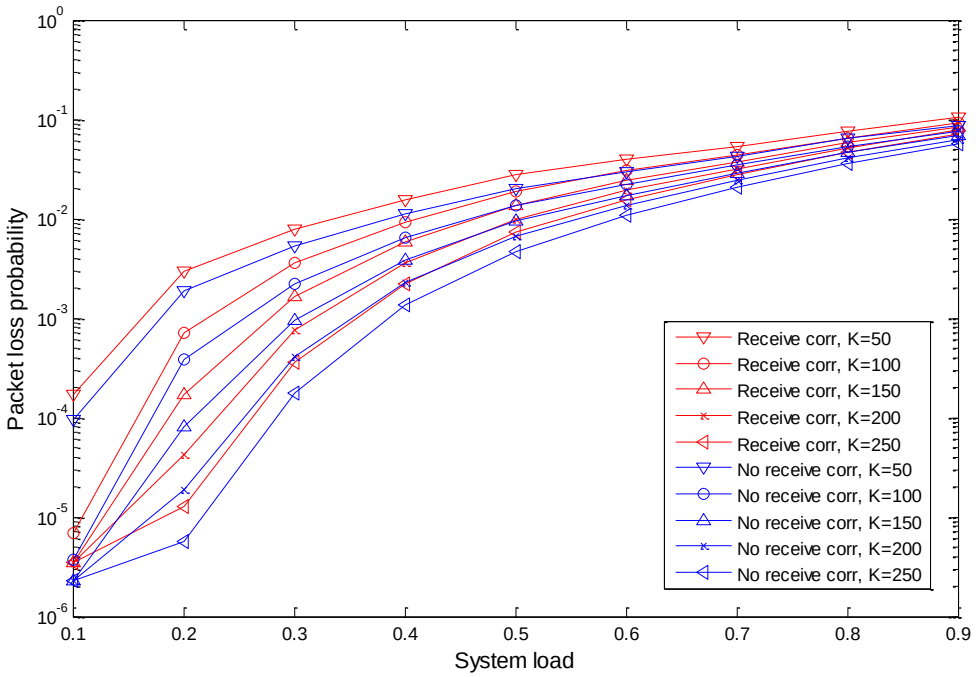

Figure 6.16 PLP comparisons with Poisson arrival

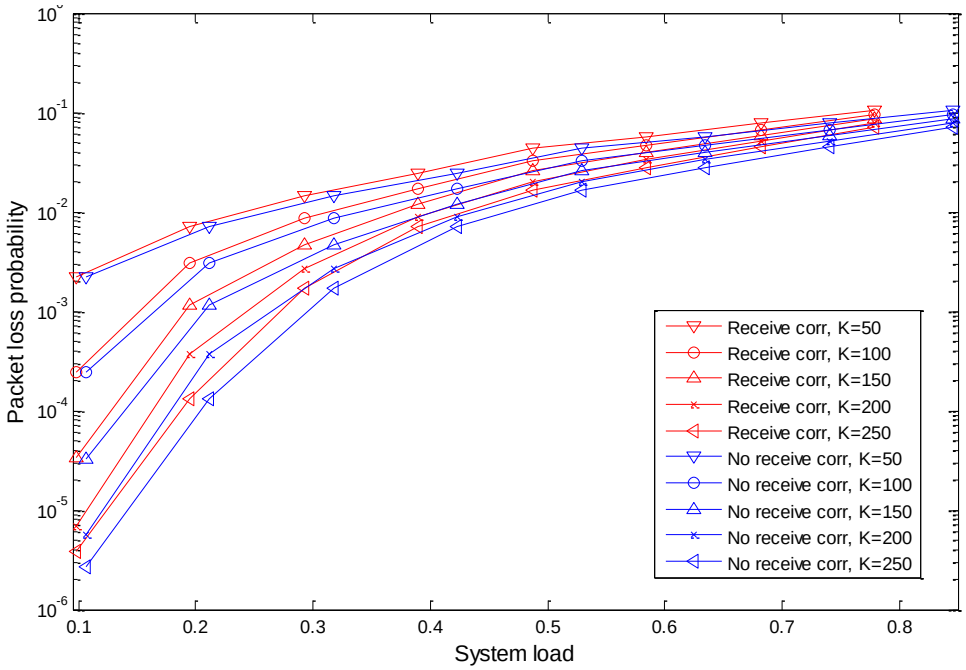

Figure 6.17 PLP comparisons with bursty arrival

6.1.3.6 Throughput

Figure 6.18 and 6.19 show the throughput, which is a measure of packets transmitted per slot, for each stream when the buffer size K=200. We have following observations:

1. We see that the stream without receive correlations achieves higher throughput, which again shows the impact of antenna correlations on queue models.

2. The spatial multiplexing technology used by the SU-MIMO channel model can dramatically increase the overall throughput for the user exploiting the spatial order. Theoretically, the user equipped with 4 antennas can achieve 4 times larger throughput than a single antenna user without antenna correlations; however, the presence of antenna correlations degrades the throughput for each stream which results in the reduction in overall throughput in the ideal case. (e.g. for the receive correlation only case with Poisson arrivals at load 0.9, the total throughput is approximately $2 \times (7.50 + 6.80) = 28.60$ packets/slot, but ideally the total throughput is $2 \times (8.80 + 8.08) = 33.76$ packets/slot by values in Table 6.3)

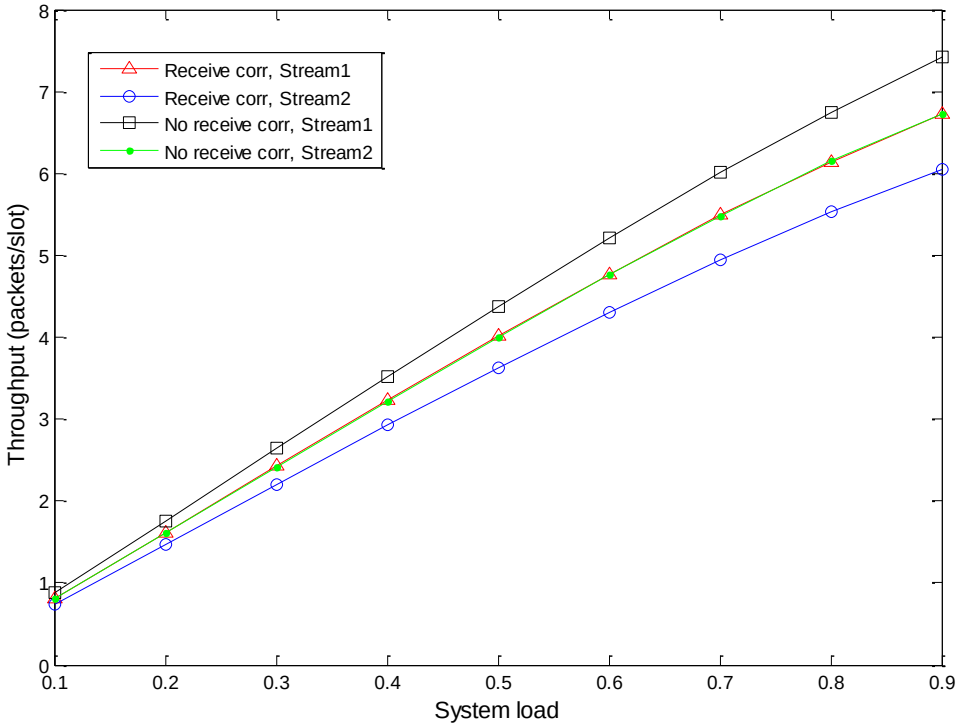

Figure 6.18 Throughput comparisons with Poisson arrival

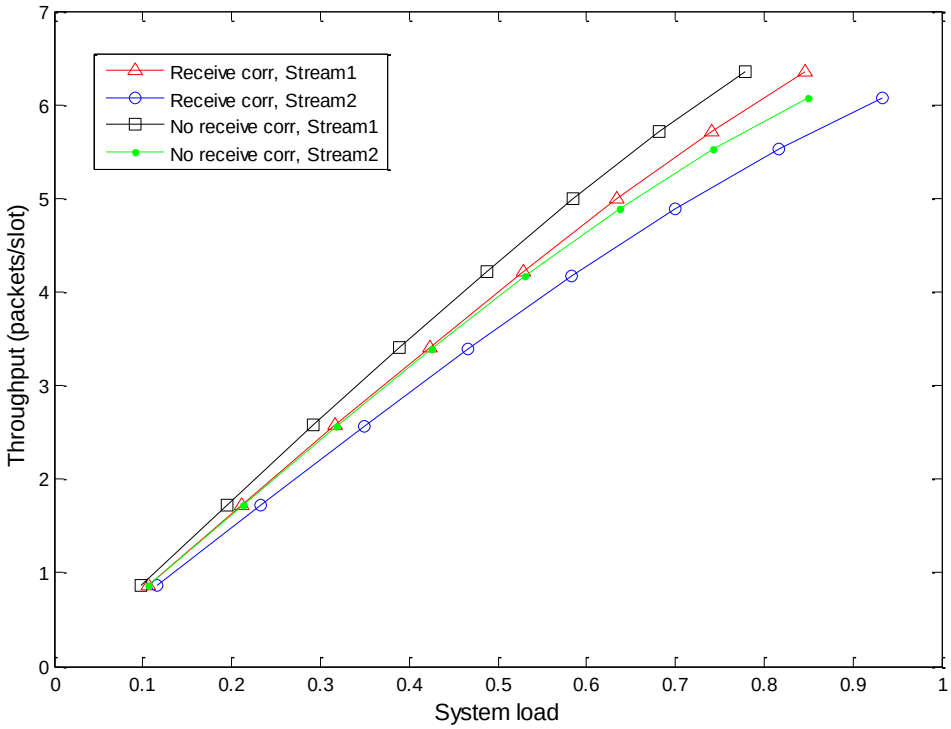

Figure 6.19 Throughput comparisons with bursty arrival

6.2 Summary

In this chapter, cross-layer analysis on IEEE 802.11ac with higher-order MCSs, wider bandwidth and more spatial streams is given. Finally, we successfully extend the original single channel single antenna scenario (specified in Chapters 4 & 5) to the SU-MIMO scenario which is able to be analyzed by the matrix geometric method. The modifications of queueing analysis, especially on the service process, are detailed for performance analysis. We adopt Poisson and bursty traffic arrival processes, which are discussed throughout Chapters 4 & 5, to approximate behaviors of different traffic types using 802.11ac. In addition, the presence of antenna correlations, which play an important role in practical MIMO systems, are discussed and evaluated in this chapter. Performance metrics including queue state in the buffer, PDP due to buffer overflow, average queueing delay, PLP and throughput are validated and analyzed by simulations. By performance comparisons, we not only illustrate the impact of fading and traffic types, but also that of the antenna correlations on the queue models. All the work done in this chapter is oriented at performance evaluation for the IEEE 802.11ac standard.

Chapter 7 Conclusion & Future Work

7.1 Conclusion

In this thesis, extensions and modifications, both at the data-link layer and the physical layer, to the cross-layer operation incorporating packet queueing effects are investigated, evaluated and discussed. The generalized G/D/s(t)/K queue model for both Poisson and bursty arrivals incorporating AMC is proposed for queueing analysis. To evaluate the packet queueing distributions, delays and losses, the matrix geometric method is adopted throughout the thesis. The main contributions are included in Chapter 5 and Chapter 6 based on the fundamental work done in Chapter 3 and Chapter 4.

7.1.1 Fundamental work

In Chapter 3, the M/D/1/K queue models with and without vacations are investigated respectively. Metrics including queue state in the buffer, PDP due to buffer overflow and average queueing delay are obtained. In addition, the comparisons between queue models with and without vacations are investigated, and the impact of vacations is examined, which lays the foundation of the extension for cross-layer analysis incorporating AMC.

In Chapter 4, the cross-layer design incorporating queueing effects and AMC is examined. This chapter is mainly based on the work already done in the literature [28]. The equivalent M/D/s(t)/K queue model with a range of working vacations, which correspond to MCSs adopted by AMC, is obtained for queueing analysis on a single channel, single user scenario. As an extension to Chapter 3, metrics including PLP and average throughput are investigated for performance analysis. As a result, the impact of fading, which determines the variations of service rate controlled by AMC, is examined.

7.1.2 Contributions

In Chapter 5, the cross-layer analysis with bursty arrivals at the data-link layer is obtained. Bursty traffic processes, including ON-OFF process and aggregated ON-OFF process, are adopted for queueing analysis. Compared to the Poisson arrivals adopted in Chapter 4, bursty arrivals can better approximate traffic types such as voice, video

and data centre in practical scenarios. The feasibility of applying the matrix geometric method in cross-layer analysis with bursty arrivals is discussed. In addition, performance analyses are obtained to reveal the impact of bursty arrivals compared to Poisson arrivals. Numerical results show the importance of choosing proper traffic models in practical scenarios.

In Chapter 6, the cross-layer analysis is extended at the physical layer to meet the specifications of IEEE 802.11ac standard with high-order MCSs, more channel bandwidth and more spatial streams. Spatial multiplexing SU-MIMO with ZF detector in the presence of both transmit and receive correlations is considered and successfully modelled by the matrix geometric method with both Poisson and bursty traffic models. The PER analysis for candidate MCSs is obtained, resulting in the re-adoption of MCSs and re-determination of SNR thresholds for the service process. Numerical results reveal the impact of antenna correlations as well as traffic models on system performance.

7.2 Future work

This thesis investigates cross-layer analysis, incorporating queueing effects and AMC both at the data-link layer and the physical layer, by queueing analysis using the matrix geometric method. In general, there are two paths that could be followed by the future work.

On one hand, the contributions of this thesis are mainly oriented at cross-layer analysis incorporating AMC. More specifically, we only investigate how the physical layer mechanisms like AMC impact on the queueing behaviour at the data-link layer; there is still a lack of studies on how the queue state at the data-link layer can affect the physical layer decisions. Therefore, the cross-layer solution/algorithm to make interactive decisions between the physical layer and the data-link layer can be investigated and developed as the continuing work. An outline of how this could work is elaborated in section 7.2.1.

On the other hand, the cross-layer analysis is extended to partly meet the specifications of IEEE 802.11ac standard in this thesis. In order to obtain more practical performance

analysis for 802.11ac, potential extensions can focus on the completion of traffic models, algorithms, protocols and mechanisms for 802.11ac, both at the data-link layer and the physical layer.

7.2.1 Cross-layer algorithm

In order to achieve a functioning cross-layer algorithm between the data-link and the physical layer, we need to specify the interactions and decision making between these two layers.

From Chapter 4 to 6, we have provided a method to evaluate how AMC at the physical layer determines the service process of the queue at the data-link layer. The service rate transition matrix is constructed to identify the transitions of MCSs adopted by AMC. Additional steps are needed to be followed in order to achieve a cross-layer algorithm. More specifically, these steps will address how the physical layer mechanisms can access the information at the data-link layer to obtain better QoS performance.

The rest of this section focusses on how the queue state at the data-link layer can be used to optimize MCS selection decisions at the physical layer.

As indicated by equation (4.28) and (4.29), both of the PLP and throughput are dependent on average PER and PDP due to buffer overflow. Roughly speaking, using higher-order MCS increases in PER under the same CSI, but decreases in the number of packets dropped if the buffer tends to be full; on the other hand, if the buffer tends to be empty, using lower-order MCS reduces PER while ensuring no packet dropped. Therefore, if the physical layer can obtain queue state information at the data-link layer, then the cross-layer algorithm can make MCS decisions for each timeslot that maximizes the throughput or minimize the PLP for the long run.

The matrix geometric method we adopt from Chapter 4 to 6 can obtain the queue state behaviour under any specific system load and any specific measured traffic burstiness (i.e. bursty transition matrix such as equation (5.5)) by equation (4.16) on a slot-by-slot basis. Recall it for convenience,

$$B_t = \min(K, \max(0, (B_{t-1} - S_t)) + A_t), \quad (4.16)$$

Originally, S_t is totally determined by CSI; a possible cross-layer algorithm can also take queue state B_{t-1} and the average number of arrival packets with a limited length of sliding window $\overline{A^W}$ into consideration. Note that we adopt a late arrival system (LAS) for cross-layer analysis throughout the thesis, which means packets in the buffer are served before new packets are arrived within each timeslot. Therefore, we use $\overline{A^W}$ instead of A_t in determining S_t in each timeslot for the cross-layer algorithm. $\overline{A^W}$ is given by (7.1) with an integer number of W,

$$\overline{A^W} = \frac{1}{W} \sum_{i=t-W+1}^{t} A_i \,, (7.1)$$

As a result, a possible cross-layer algorithm determining S_t for each timeslot is given as below; note that the service rate can only be chosen from a vector ***S*** and *S(n)* denotes the service rate in mode n.

1. Get the service rate $S_t' = S(n)$ which is totally determined by CSI with mode n
2. (The case when the buffer tends to go empty)
   Check if $B_{t-1} < S_t'$. If so, go to step 3; if not, the updated service rate $S_t'' = S(i) = S_t'$, then go to step 4
3. FOR $i = n: 0$
   IF $B_{t-1} \geq S(i)$
   $S_t'' = S(i)$ BREAK
   ELSE
   CONTINUE
   ENDIF
   END

4. (The case when the buffer tends to cause overflow)

   Calculate the predicted number of packets dropped due to buffer overflow $\overline{D_t}$ within the timeslot by $\overline{D_t} = \max(0, \max(0, (B_{t-1} - S_t'')) + \overline{A^W} - K)$, and calculate the average packets dropped by transmission error $\overline{E_t}$ within the timeslot by $\overline{E_t} = S_t'' \times PER_i(\gamma)$, where $PER_i(\gamma)$ is obtained by equation (4.1). Therefore, the average of total packets dropped $\overline{D_{total}(\imath)} = \overline{D_t} + \overline{E_t}$ for mode *i*.

5. IF $i < N$

   FOR $j = i : N$

   CALCULATE $\overline{D_{total}(J)}$ and $\overline{D_{total}(J+1)}$

   IF $\overline{D_{total}(J)} > \overline{D_{total}(J+1)}$

   $S_t = S(j+1)$

   ELSE

   $S_t = S(j)$ BREAK

   ENDIF

   END

   ELSE

   $S_t = S(j)$

   ENDIF

6. OUTPUT S_t as the determined service rate for the timeslot after the cross-layer algorithm

In summary, the above proposal for a cross-layer algorithm exploits the queue state and traffic arrival behaviours at the data-link layer to affect the MCS decisions at the physical layer on a slot-by-slot basis. Such a cross-layer algorithm could be used to optimize the throughput, which would be one of the main contributions for any such future work.

7.2.2 Extensions to the full specifications of 802.11ac

7.2.2.1 Extensions to the data-link layer

Since queueing effects are the only data-link layer factor considered in this thesis, more data-link layer mechanisms could be applied into the cross-layer analysis.

As introduced in section 2.3.2, an extensive amount of work has investigated the cross-layer design incorporating ARQ; moreover, [23] has provided the framework for the cross-layer design incorporating both queueing effects and ARQ using the matrix geometric method, and it is possible to incorporate ARQ into the cross-layer analysis investigated in Chapter 6. A tuple with four elements, including queue state, service state, traffic arrival state and ARQ retransmission state, would need to be considered for constructing the probability transition matrix in order to obtain the stationary distribution for the queue length in the buffer. The hidden challenge would be the operation time for solving the left eigenvector of the probability transition matrix because, in such a proposed algorithm, the matrix would get much larger.

Besides ARQ, more mechanisms at the data-link layer such as a CSMA/CA mechanism could be considered for a similar cross-layer analysis in 802.11-based networks. This collision avoidance mechanism is applied to the uplink scenario when multiple users try to set up transmission to an AP when that AP is only able to communicate with a single user. The published work has provided some heuristic approaches to achieve a queueing analysis incorporating CSMA/CA. [37] incorporated the classical Distributed Coordination Functions (DCF) as the medium access method for queueing analysis, while [38] applied Distributed Queueing Collision Avoidance (DQCA) MAC protocol, which adopted two queues: one for solving collisions and the other for data transmissions, for queueing analysis. The challenge to incorporate CSMA/CA mechanism in the cross-layer analysis would be the complexity of the queueing analysis.

In addition, as suggested by [36] [48] [37], an MMPP traffic model is also good at capturing burstiness of packet traffic as an alternation to using N-ON-OFF processes. In order to get a better understanding of the bursty traffics, an MMPP can be used as the bursty traffic model for the cross-layer analysis. A tuple with queue state, service state and arrival state could be used for constructing the probability transition matrix.

However, the problem of finding an ideal bursty traffic model is still not addressed; the underlying solution is keeping track of new experimental achievements in this related field.

7.2.2.2 Extensions to the physical layer

In Chapter 6, only a SU-MIMO system is considered at the air interface; however, 802.11ac also supports MU-MIMO. [33] claimed to extend to MU-MIMO scenario, but the system model provided in this paper was equivalent to SU-MIMO. As far as we know, in order to achieve a MU-MIMO operation model, a precoding operation such as beamforming should be applied at the transmitter (i.e. the AP for 802.11-based networks) to separate spatial streams for different users. In addition, more detectors other than ZF detector, such as the minimum mean squared error (MMSE) detector, could be considered for equalizing at the receiver side. The challenge is in how to obtain the SNR distribution for each spatial stream by ZF or MMSE detector with precoded channel gain matrix, and this would also make interesting future work.

Another interesting extension is in antenna selection for users. As specified in 802.11ac, an AP can be equipped with at most 8 antennas, while a user device can be equipped with at most 4 antennas. When there are sufficient antennas available for the AP, it can perform antenna selection for performance optimization. [62] provided two antenna selection strategies: one to maximize the average throughput and the other to minimize the average SER. The strategies are achieved by investigating diagonal entries of the inverse of transmit correlation matrix; however, these strategies only work for the ZF detector with transmit correlation only. In addition, the available antennas for AP after antenna selection can transmit the same data to achieve diversity order or go into sleep mode to save energy. Therefore, further investigation could be focused on performance optimizations on the cross-layer analysis.

References

[1] K. Madan, M. Saleh, *On M/D/1 queue with general server vacations*, Information and Management Sciences, vol. 12, No. 2, pp 25-37, 2001

[2] L. Servi, S. Finn, *M/M/1 queues with working vacations,* Elsevier Performance Evaluation 50, pp. 41–52, 2002

[3] D. Wu and H. Takagi, *M/G/1 queues with multiple working vacations,* Elsevier Performance Evaluation, Vol. 63, Issue 7, pp 654-681, July, 2006

[4] J. Li, N. Tian and W. Liu, *Discrete time GI/Geo/1 queue with multiple working vacations,* Queueing Systems, Vol. 56, No. 1, pp 53-63, 2007

[5] N. Tian, Z. Ma and M. Liu, *Discrete time Geom/Geom/1 queue with multiple working vacations,* Elsevier Applied Mathematical Modelling, pp 2941–2953, 2008

[6] M. Alouini, A. Goldsmith, *Adaptive modulation over Nakagami fading channels*, IEEE Vehicular Technology Conference, USA, June, 1997

[7] TSG-RAN WG1£17, *Adaptive modulation and coding (AMC),* Stockholm, Sweden, Oct, 2000

[8] S. Chung and A. Goldsmith, *Degrees of freedom in adaptive modulation: A unified view*, IEEE Transactions on Communications, vol. 49, no. 9, pp. 1561–1571, Sep. 2001.

[9] K. Song, A. Ekbal, S. Chung and J. Cioffi, *Adaptive modulation and coding for bit-interleaved coded OFDM,* IEEE Transactions on Wireless Commu, vol. 5, No. 7, July, 2006

[10] J. Cai, Q. Dou, *The performance study of MCS selection for IEEE 802.11n system*, Master's degree thesis for National Yunlin University, June, 2010

[11] H. Wang and N. Moayeri, *Finite-state Markov channel—A useful model for radio communication channels*, IEEE Transactions on Vehicular Technology, vol. 44, pp. 163–171, Feb. 1995

[12] Q. Zhang, S. Kassam, *Finite-state Markov model for Rayleigh fading channels,* IEEE Transactions on Communications, 1999

[13] J. Razavilar, K. Liu, S. Marcus, *Jointly optimized bit-rate delay control policy for wireless packet networks with fading channels,* IEEE Transactions on Communications, 2002

[14] R. Ertel, J. Reed, *Generation of two equal power correlated rayleigh fading envelops,* IEEE Communication Letters, 1998

[15] B. Natarajan, C. Nassar, V. Chandrasekhar, *Generation of correlated Rayleigh fading envelopes for spread spectrum applications,* IEEE Communication Letters, 2000

[16] W. Al-Hussaibi, F. Ali, *Generation of correlated Rayleigh fading channels for accurate simulation of promising wireless communication systems,* Simulation Modelling Practice and Theory, 2012

[17] S. Shakkottai, T. Rappaport, P. Karlsson, Cross-layer design for wireless networks, IEEE Communication Magazine, 2003

[18] H. Jiang, W. Zhuang, X. Shen, *Cross-layer design for resource allocation in 3G wireless networks and beyond,* IEEE Communications Magazine, 2005

[19] F. Foukalasm V. Gazis, N. Alonistioti, *Cross-layer design proposals for wireless mobile networks a survey and taxonomy,* IEEE Communications Surveys and Tutorials, 2008

[20] Q. Zhang, Y. Zhang, *Cross-Layer Design for QoS Support in Multihop Wireless Networks,* Proceedings of the IEEE, 2008

[21] B. Fu, Y. Xiao, H. Deng, H. Zeng, *A survey of cross-layer designs in wireless networks,* IEEE Communications Surveys and Tutorials, 2014

[22] Q. Liu, S. Zhou, G. Giannakis, *Cross-Layer Combining of Adaptive Modulation and Coding With Truncated ARQ Over Wireless Links*, IEEE Transactions on Wireless Commun, Sep. 2004

[23] X. Wang, Q. Liu and G.Giannakis, *Analyzing and optimizing adaptive modulation coding jointly with ARQ for QoS-guaranteed traffic,* IEEE Transactions on Vehicular Technology, vol. 56, No. 2, March, 2007

[24] G. Aniba, S. Aissa, *Cross-layer designed adaptive modulation algorithm with packet combining and truncated ARQ over MIMO Nakagami fading channels,* IEEE Transactions on Wireless Communicationws, 2011

[25] Y. Yang, H. Ma, S. Aissa, *Cross-layer combining of adaptive modulation and truncated ARQ under cognitive radio resource requirements,* IEEE Transactions on Vehicular Technology, 2012

[26] G. Zhang, L. Sun, H. Wen, B. Wu, X. Zhu, L. Zhou, *A Cross Layer Design Combining of AMC with HARQ for DSRC Systems,* International journal of Distributed Sensor Networks, 2013

[27] D. Wang, L. Lu, Y. Fang, J. Wu, *Cross-layer design for LTE system with jointly AMC and ARQ on fading channel,* IEEE WCNC, 2013

[28] Q. Liu, S. Zhou, G. Giannakis, *Queuing With Adaptive Modulation and Coding Over Wireless Links: Cross-Layer Analysis and Design*, IEEE Transactions on Wireless Commun, May 2005

[29] X. Zhang, J. Tang, H. Chen, S. Ci, M. Guizani, *Cross-layer-based modeling for quality of service guarantees in mobile wireless networks,* IEEE Communications Magazine, 2006

[30] J. Harsini, and F. Lahouti, *Queuing with Adaptive Modulation over MIMO Wireless Links for Deadline Constrained Traffic: Cross-Layer Analysis and Design*, IEEE ICC, 2007

[31] S. Zhou, K. Zhang, Z. Niu and Y. Yang, *Queuing Analysis on MIMO Systems with Adaptive Modulation and Coding*, IEEE ICC 2008

[32] X. Liao, L. Kong, J. Jiang, P. Zhang, *Queuing Analysis on Adaptive Transmission in MIMO Systems with Imperfect CSI,* IEEE Vehicular Technology Conference Fall, 2009

[33] M. Rashid, E. Hossain, V. Bhargava, *Cross-layer analysis of downlink V-BLAST MIMO transmission exploiting multiuser diversity,* IEEE Transactions on Wireless Communications, 2009

[34] K. Zheng, Y. Wang, L. Lei and W. Wang, *Cross-layer queuing analysis on multi-hop relaying networks with adaptive modulation and coding*, IET Communications, 2010

[35] J. Gong, S. Zhou, Z. Niu, *Queuing on Energy-Efficient Wireless Transmissions with Adaptive Modulation and Coding,* IEEE ICC, 2011

[36] N. Wang, T. Gulliver, *Cross layer AMC scheduling for a cooperative wireless communication system over Nakagami-m fading channels,* IEEE Transactions on Wireless Communications, 2012

[37] C. Park, H. Jung, D. Han, *Queueing Analysis of IEEE 802.11 MAC Protocol in Wireless LAN,* ICN/ICONS/MCL, 2006

[38] E. Kartsakli, A. Cateura, L. Alonso, J. Alonso-Zarate, *Cross-layer enhancement for wlan systems with heterogeneous traffic based on DQCA,* IEEE Communications Magazine, 2008

[39] G. Redieteab, L. Cariou, P. Christin, J. Helard, *Cross-layer multichannel aggregation for future WLAN systems,* IEEE International Conference on Communication Systems, 2010

[40] I. Elhanany, D. Sadot, *Queueing Analysis of Markov Modulated ON/OFF Arrivals with Geometric Service Times,* The 22nd Convention of Electrical and Electronics Engineers in Israel, 2002

[41] D. Niyato, E. Hossain, *Queueing analysis of OFDM/TDMA systems,* IEEE Globecom'05, December, 2005

[42] S. Vassilaras, *A cross-layer optimized adaptive modulation and coding scheme for transmission of streaming media over wireless links*, Wireless Networks, 2010

[43] G. Redieteab, L. Cariou, P. Christin, J. Helard, *SUMU-MIMO in IEEE 802.11 ac PHY+ MAC performance comparison for single antenna stations,* IEEE Wireless Telecommunications Symposium, 2012

[44] J. Pitts, J. Schormans, *Introduction to IP and ATM Design and Performance: With Applications Analysis Software, Chapter 15: Resource Reservation,* Chichester: John Wiley &Sons, Edition 2, 2000

[45] L. Lipsky, M. Jobmann, M. Greiner, H. Schwefel, *Comparison of the analytic N-burst model with other approximations to telecommunications traffic,* IEEE International Symposium on NCA, pp. 122-132, 2001

[46] X. Yang, *Designing Traffic Profiles for Bursty Internet Traffic,* IEEE Globecom'02, vol. 3, 2002

[47] J. Zhang, *Bursty traffic meets fading, a cross-layer design perspective*, Proceedings of the annual Allerton conference on communication control and computing, vol. 40, No. 3, 2002

[48] B. Chandrasekaran, *Survey of Network Traffic Models,* Washington University in St. Louis CSE, 2009

[49] A. Zvironiene, Z. Navickas, R. Rindzevicius, *Bursty Traffic Simulation by ON - OFF Model,* Electronics and Electrical Engineering, 2015

[50] ETSI Technical specification, *Broadband radio access networks (BRAN); HIPERLAN Type 2; Physical layer*, April, 2000

[51] A. Doufexi, S. Armour, M. Butler, A. Nix, D. Bull, J. McGeehan, and P. Karlsson, *A comparison of the HIPERLAN/2 and ieee 802.11a wireless LAN standards*, IEEE Commun. Mag., vol. 40, no. 5, pp. 172–180, May, 2002

[52] Shankland, Stephen, *Study: Expect a billion 802.11ac Wi-Fi devices in 2015*. Cnet, February, 2011.

[53] QUALCOMM, Incorporated, *IEEE802.11ac: The next evolution of Wi-Fi Standards*, May, 2012

[54] A.Viterbi, *Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm*, IEEE Transactions on Information Theory, vol. IT-13, pp. 260-269, April, 1967.

[55] G. Al-Rubyai, *Bit error and packet error probability for rectangular Mary-QAM in OFDM transmission system over Rayleigh fading channels,* Journal of Engineering and Development, vol. 11, No. 3, Dec, 2007

[56] MIT 6.02 draft lecture notes, Lecture 8: Convolutional coding, 2010

[57] MIT 6.02 draft lecture notes, Lecture 9: Viterbi decoding of convolutional codes, 2010

[58] F. Tong, *Data encoding methods and apparatus*, UK patent application, GB 2506491, April, 2014

[59] Q. Zhang, *Maximal-ratio combining over Nakagami fading channels with an arbitrary branch covariance matrix,* IEEE Transactions on Vehicular Technology, 1999

[60] H. Bolcskei, A. Paulraj, *Performance of space-time code in the presense of spatial fading correlation,* Conference record for 34th Asilomar conference on Singals, Systems and Computers, 2002

[61] D. Gore, R. Heath Jr, A. Paulraj, *On performance of the zero forcing receiver in presence of transmit correlation,* IEEE International Symposium on Information Theory, 2002

[62] D. Gore, R. Heath Jr, A. Paulraj, *Transmit selection in spatial multiplexing systems,* IEEE Communication Letters, 2002

[63] M. Kiessling, J. Speidel, *Analytical performance of MIMO zero-forcing receivers in correlated Rayleigh fading environments,* IEEE Workshops on Signal Processing Advances in Wireless Communications, 2003

[64] R. Xu, F. Lau, *Performance analysis for MIMO systems using zero forcing detector over fading channels,* IEE Proceedings-Communications, 2006

[65] C. Wang, E. Au, R. Murch, W. Mow, R. Cheng, V. Lau, *On the performance of the MIMO zero-forcing receiver in the presence of channel estimation error,* IEEE Transactions on Wireless Communications, 2007

[66] F. Adachi, M. Feeney, J. Parsons, *Effects of correlated fading on level crossing rates and average fade durations with predetection diversity reception,* IEE Proceedings on Communications, Radar and Signal Processing, February, 1988

[67] M. Yacoub, J. Bautista, L. de Rezende Guedes, *On higher order statistics of the Nakagami-m distribution,* IEEE Transactions on Vehicular Technology, 1999

[68] X. Dong, N. Beaulieu, *Average level crossing rate and fade duration of maximal ratio diversity in unbalanced and correlated channels,* IEEE WCNC, March, 2002

[69] D. Li, V. Prabhu, *Average level crossing rates and average fade durations for maximal-ratio combining in correlated Nakagami channels,* IEEE WCNC, March, 2004

[70] N. Zlatanov, Z. Hadzi-Velkov, G. Karagiannidis, *Level Crossing Rate and Average Fade Duration of the Double Nakagami-m Random Process and Application in MIMO Keyhole Fading Channels,* IEEE Communications Letters, 2009

[71] K. Jamshaid, B. Shihada, L. Xia, P. Levis, *Buffer Sizing in 802.11 Wireless Mesh Networks,* IEEE 8th International Conference on Mobile Adhoc and Sensor Systems, 2011

[72] T. Li, D. Leith, D. Malone, *Buffer Sizing for 802.11 Based Networks,* IEEE/ACM Transactions on Networking (TON), 2011

[73] A. Showail, K. Jamshaid, B. Shihada, *Buffer sizing in wireless networks: challenges solutions and opportunities,* IEEE Communications Magazine, 2014

[74] L. Kleinrock, *Queueing Systems.* Vol. 1: Theory. John Wiley & Sons, New York, 1975

[75] R. Ertel, P. Cardieri, K. Sowerby, T. Rappaport, J. Reed, *Overview of spatial channel models for antenna array communication systems,* IEEE Personal Communications, 1998

[76] K. Cho, D. Yoon, *On the general BER expression of one- and two dimensional amplitude modulations,* IEEE Transactions on Communications, July, 2002

[77] L. Rubio, J. Reig, N. Cardona, *Evaluation of Nakagami fading behaviour based on measurements in urban scenarios,* AEU-International Journal of Electronics and Communications, 2007

[78] S. Andreev, A. Anisimov, Y. Koucheryavy and A. Turlikov, *Practical Traffic Generation Model for Wireless Networks*, ERCIM workshop on Mobility, May, 2010

[79] P. Bremaud, *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues,* New York: Springer-Verlag, 1999

Appendix

A. **Determining SNR thresholds for MCSs**

From Equation (4.2), we obtain that SNR thresholds are dependent on prescribed PER. Therefore, before we get SNR thresholds for MCSs, we need to analyse the PER curve with different SNRs. It has been discussed in [22] that the PER for convolutional coded modulations cannot be obtained directly from BER although the exact BER for QAM has been given by [76]. Thus, we use Monte-Carlo simulation to get the PER with a range of SNRs, and use least square regression analysis with exponential fitting (as specified by (4.2)) to approximate PER behaviour for each MCS.

We choose MCSs adopted by 802.11ac, which is listed in Table 6.1, as an example. System parameter settings are given in Table 6.4. Then, we get the PER curves for different MCSs versus SNR shown by Figure A1.

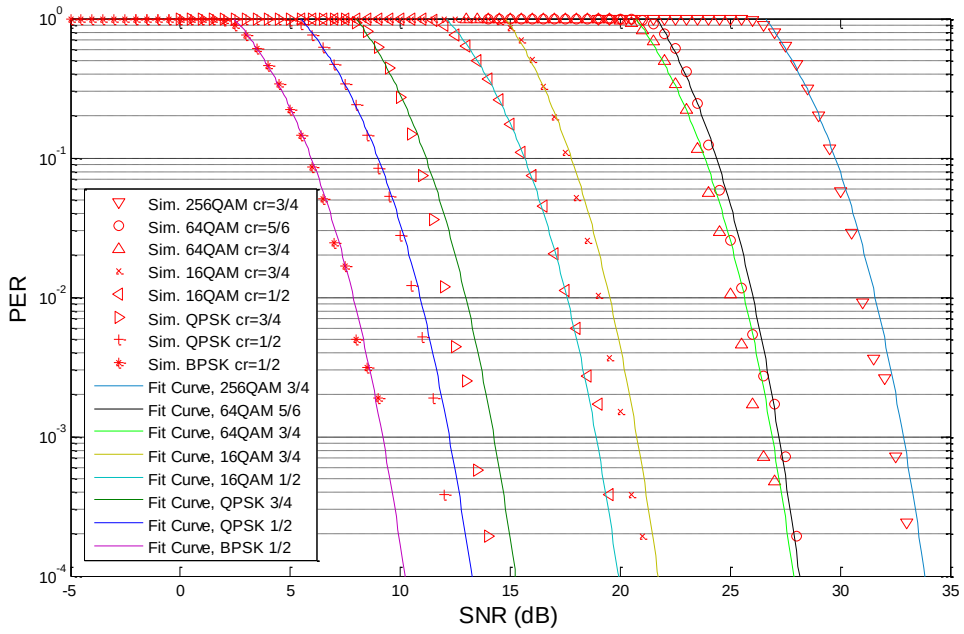

Figure A1 PER curve for MCSs used by 802.11ac

We find that there is no need to use all MCSs adopted by 802.11ac for queueing analysis, because some MCSs (such as 64QAM 3/4 and 64QAM 5/6) behaves nearly the same in PER analysis. In addition, published work such as [7] claims to use fewer MCSs with slightly throughput change to reduce complexity. Thus, we select MCS 0, 2, 4, 6, 8 for further queueing analysis instead of all of the MCSs adopted by 802.11ac.

Below is a chart provided by WLAN Pro to describe the SNR to MCS data rate mapping under different Wi-Fi standards (Figure A2). These are simply a generic approximation for Wi-Fi devices but still provide valuable information to estimate the range to use each MCS. We focus on mappings for 802.11ac with 20 MHz channel bandwidth, and find that our values generated by Monte-Carlo simulation (Table 6.5 and Figure A1) kind of agree to the values captured by real Wi-Fi devices.

MCS Value Achieved by Clients at Various Signal to Noise Ratio Levels (SNR)

Protocol	Channel	1	2	3	4	5	6	7	8	9	10
802.11b	20MHz	None	None	None	MCS 0	MCS 0	MCS 0	MCS 1	MCS 1	MCS 1	MCS 1
802.11a/g	20MHz	None	MCS 0	MCS 0	MCS 1	MCS 2	MCS 2	MCS 2	MCS 2	MCS 3	MCS 3
802.11n	20MHz	None	MCS 0	MCS 0	MCS 0	MCS 1	MCS 1	MCS 1	MCS 1	MCS 2	MCS 2
802.11n	40MHz	None	None	None	None	MCS 0	MCS 0	MCS 0	MCS 1	MCS 1	MCS 1
802.11ac	20MHz	None	MCS 0	MCS 0	MCS 0	MCS 1	MCS 1	MCS 1	MCS 1	MCS 2	MCS 2
802.11ac	40MHz	None	None	None	None	MCS 0	MCS 0	MCS 0	MCS 1	MCS 1	MCS 1
802.11ac	80MHz	None	None	None	None	None	None	None	MCS 0	MCS 0	MCS 0
802.11ac	160MHz	None	None	None	None	None	None	None	None	None	None

	SNR in dB	11	12	13	14	15	16	17	18	19	20
802.11b	20MHz	MCS 2	MCS 2	MCS 2	MCS 2	MCS 2	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3
802.11a/g	20MHz	MCS 4	MCS 4	MCS 4	MCS 4	MCS 5	MCS 5	MCS 5	MCS 6	MCS 6	MCS 7
802.11n	20MHz	MCS 3	MCS 3	MCS 3	MCS 3	MCS 4	MCS 4	MCS 4	MCS 5	MCS 5	MCS 6
802.11n	40MHz	MCS 1	MCS 2	MCS 2	MCS 3	MCS 3	MCS 3	MCS 3	MCS 4	MCS 4	MCS 4
802.11ac	20MHz	MCS 3	MCS 3	MCS 3	MCS 3	MCS 4	MCS 4	MCS 4	MCS 5	MCS 5	MCS 6
802.11ac	40MHz	MCS 1	MCS 2	MCS 2	MCS 3	MCS 3	MCS 3	MCS 3	MCS 4	MCS 4	MCS 4
802.11ac	80MHz	MCS 1	MCS 1	MCS 1	MCS 1	MCS 2	MCS 2	MCS 3	MCS 3	MCS 3	MCS 3
802.11ac	160MHz	MCS 0	MCS 0	MCS 0	MCS 1	MCS 1	MCS 1	MCS 1	MCS 2	MCS 2	MCS 3

	SNR in dB	21	22	23	24	25	26	27	28	29	30
802.11b	20MHz	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3
802.11a/g	20MHz	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7
802.11n	20MHz	MCS 6	MCS 6	MCS 6	MCS 6	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7
802.11n	40MHz	MCS 5	MCS 5	MCS 6	MCS 6	MCS 6	MCS 6	MCS 6	MCS 7	MCS 7	MCS 7
802.11ac	20MHz	MCS 6	MCS 6	MCS 6	MCS 6	MCS 7	MCS 7	MCS 7	MCS 7	MCS 8	MCS 8
802.11ac	40MHz	MCS 5	MCS 5	MCS 6	MCS 6	MCS 6	MCS 6	MCS 6	MCS 7	MCS 7	MCS 7
802.11ac	80MHz	MCS 4	MCS 4	MCS 4	MCS 5	MCS 5	MCS 6	MCS 6	MCS 6	MCS 6	MCS 6
802.11ac	160MHz	MCS 3	MCS 3	MCS 3	MCS 4	MCS 4	MCS 4	MCS 5	MCS 5	MCS 6	MCS 6

	SNR in dB	31	32	33	34	35	36	37	38	39	40
802.11b	20MHz	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3
802.11a/g	20MHz	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7
802.11n	20MHz	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7
802.11n	40MHz	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7
802.11ac	20MHz	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9
802.11ac	40MHz	MCS 7	MCS 8	MCS 8	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9
802.11ac	80MHz	MCS 7	MCS 7	MCS 7	MCS 7	MCS 8	MCS 8	MCS 9	MCS 9	MCS 9	MCS 9
802.11ac	160MHz	MCS 6	MCS 6	MCS 6	MCS 7	MCS 7	MCS 7	MCS 7	MCS 8	MCS 8	MCS 9

	SNR in dB	41	42	43	44	45	46	47	48	49	50
802.11b	20MHz	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3	MCS 3
802.11a/g	20MHz	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7
802.11n	20MHz	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7
802.11n	40MHz	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7	MCS 7
802.11ac	20MHz	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9
802.11ac	40MHz	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9
802.11ac	80MHz	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9
802.11ac	160MHz	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9	MCS 9

Modulation Key
None = Grey
BPSK = Red
QPSK = Orange
16-QAM = Yellow
64-QAM = Blue
256-QAM = Green

802.11 Type Key
802.11b
802.11ag
802.11n
802.11ac

Figure A2 SNR and MCS mapping for Wi-Fi standards