Performance Modeling, Design and Analysis of Transport Mechanisms in Integrated Heterogeneous Wireless Networks

by

Humphrey Rutagemwa

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2007

© Humphrey Rutagemwa, 2007

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Recently, wireless access to Internet applications and services has attracted a lot of attention. However, there is no single wireless network that can meet all mobile users' requirements. Consequently, integrated heterogeneous wireless networks are introduced to meet diverse wireless Internet applications and services requirements. On the other hand, integrated heterogeneous wireless networks pose new challenges to the design and development of reliable transport mechanisms. Wireless Application Protocol version 2 (WAP 2.0) is one of the promising transport mechanisms. It uses wireless profiled TCP (WP-TCP), which is fully compatible with TCP, as one of the reliable transport protocols to cope with the wireless link impairments. For WAP 2.0 to continue providing reliable and efficient transport services in the future, one of the key issues is to thoroughly study, understand, and improve its performance in integrated heterogeneous wireless networks.

In this thesis, we develop analytical frameworks and propose a solution to respectively study and improve the performance of WP-TCP in integrated heterogeneous wireless networks. Specifically, we consider WP-TCP short- and long-lived flows over integrated wireless local area network (WLAN) and wireless wide area network (WWAN), where WLAN can be static or mobile. In order to facilitate the analysis of WP-TCP performance in integrated WLAN and WWAN, we first construct a novel WLAN link model, which captures the impact of both uncorrelated and correlated transmission errors, and derive mathematical expressions that describe packet loss probability and packet loss burst length over WWAN-WLAN link.

Then, we develop analytical frameworks for studying the performance of WP-TCP short- and long-lived flows. Differently from those reported in the literature, our analytical framework for

WP-TCP short-lived flows takes into account both correlated and uncorrelated packet losses. Furthermore, our analytical framework for long-lived flow can be used to study the short-term (during vertical handover) and long-term performances of WP-TCP and it captures the effects of vertical handover, such as excessive packet losses and sudden change in network characteristics, which are commonly experienced in integrated static WLAN and WWAN. By using the developed analytical frameworks, we extensively analyze the performance of WP-TCP flows and investigate the optimal protocol design parameters over a wide range of network conditions.

Finally, based on our analytical studies, we propose a receiver-centric loosely coupled cross-layer design along with two proactive schemes, which significantly improve the vertical handover performance. The proposed solution is easy to implement and deploy, compatible with traditional TCP, and robust in the absence of cross-layer information. Extensive simulations have been conducted to confirm the effectiveness and practicability of our schemes.

Acknowledgements

This thesis is the result of three and half years of hard work at the University of Waterloo, where many people and institutions have helped me in many ways. I am now delighted to take this opportunity to express my gratitude and appreciation to them.

First and foremost, I am deeply grateful to my supervisors, Professor Xuemin (Sherman) Shen and Professor Jon W. Mark, for their continuous guidance, kind support, and great enthusiasm for high-quality research. Throughout my graduate studies, I have benefited greatly from their invaluable advice on both, my research and career path. It has been a tremendous pleasure and inspiration to work with them. Definitely, this thesis would have been impossible without their help.

I would like to thank Professor Yuguang (Michael) Fang for serving as the external examiner and sharing his valuable ideas that have further refined this thesis. I would also like to thank other members of my Ph.D. defense committee, Professor Yulia Gel, Professor Sagar Naik, Professor Pin-Han Ho, for their insightful and helpful comments and suggestions on my thesis proposal and the earlier versions of this thesis.

It has been an honor for me to be a member of the Broadband Communications Research (BBCR) Group at the University of Waterloo. I remain grateful to Professor Weihua Zhuang and other members of the BBCR Group for their friendship and spirit of intellectual exchange. Special thanks to past members of this group, Dr. Sangheon Pack, Dr. Minghui Shi, and Dr. Lin Cai, for their fruitful collaborations, which have improved the quality of this thesis.

Financial support for my graduate studies came from the Natural Science and Engineering Research Council (NSERC) of Canada, the University of Waterloo, the QNX Software Systems, and the Bell Canada. Ontario Research & Development Challenge Fund Bell/QNX Scholarships and University of Waterloo Graduate Scholarships, Faculty of Engineering Graduate Scholarships, Conference Travel Fund Scholarships, and International Graduate Student Awards are gratefully acknowledged. Thanks to Wendy Boles, Wendy Gauthier, Karen Schooley, Lisa Hendel, Anne Jenson, and Elaine Garner, for their friendly attitude and administrative support.

I wish to express a special thanks to invaluable friends for their kind support during my graduate studies: Mahinthan Veluppillai, Zachariah Diigula, Gabriel N. Njikang, Aliko Mwakatobe, Patrick Reid, Heidi Hoernig, and Agnes Mwasumbi.

Finally, my greatest gratitude belongs to my wife Julia Neema for her understanding, love, and selfless support in the preparation of this thesis. I am forever indebted to my lovely parents, sisters, and brothers: Eustace, Ernestina, Abela, Joyce, Florence, Egbert, Hilda, and Fredolin. Their continued support, endless patience, and encouragement, in the times when it was most needed, have enabled me to get this far.

Dedication

To my late mother, Ernestina Kokutekeleza and late sister, Florence Kokubelwa

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

The number of worldwide Internet users surpassed 1 billion in 2005 and is expected to reach 2 billion in 2011 according to the Computer Industry Almanac [1]. Recently, the world has witness a large number of wireless networks, such as wireless wide area networks (WWAN) and wireless local area networks (WLAN), deployed to provide Internet access to mobile users. Cellular networks (dominant type of WWAN) have evolved to the third generation (3G) cellular networks on a large commercial scale in order to meet the demand for wireless services. WLANs have been massively deployed in buildings/open areas (e.g., convention centers, cafes, airports, shopping malls, etc.) and mobile platforms (e.g., trains, buses, ships, ferries, etc.) in order to provide convenient internet access. Despite of these dramatic changes, no single wireless network can satisfy the ever-increasing demands of wireless Internet applications for seamless coverage and high bandwidth. For example, the 3G cellular networks can cover a wide geographic area and support quality of service (QoS) with high price tag. Whereas WLANs can provide higher data rate services at much lower cost without good QoS provisioning and only over a small geographic area.

In order to support diverse wireless Internet applications and services requirements, integrated heterogeneous wireless networks such as WWANs (e.g., GPRS and UMTS cellular networks) and WLANs (e.g., IEEE 802.11x and HiperLAN/2) are introduced.

The integration of WLAN and WWAN is a novel concept to offer Internet users both high-rate data services and ubiquitous connectivity in wide area. However, the integration poses new challenges to reliable transport mechanisms. Wireless Application Protocol version 2 (WAP 2.0) is one of the promising transport mechanisms for future wireless networks. It is a de-facto world standard for the presentation and delivery of wireless information on wireless devices and it supports standard Internet protocols. WAP 2.0 uses wireless profiled transmission control protocol (TCP) to provide connection-oriented reliable transport service. Wireless profiled TCP (WP-TCP) is optimized for wireless communication networks and it can interoperate with standard TCP Internet implementations. WP-TCP is the key transport protocol that supports attractive applications such as wireless browsing, emails, and file downloads.

For WAP 2.0 to continue providing reliable and efficient transport services in the future, one of the key issues is to thoroughly study and improve its performance in integrated heterogeneous wireless networks. In this thesis, we focus on the challenges that arise in studying and improving the performance of WP-TCP in integrated WLAN and WWAN.

1.2 Problem Description

Figure 1.1 illustrates integrated heterogeneous wireless networks scenarios in urban areas, which are considered in this thesis. In the first scenario the WWAN is integrated with WLAN installed in mobile platforms (mobile WLAN) and in the second scenario the WWAN is integrated with WLAN installed in the buildings/open areas (static WLAN). In the integrated mobile WLAN and WWAN scenario, a mobile node (MN) in the mobile platform communicates with a correspondent node (CN) in the Internet over a series of wireless links: mobile WLAN and WWAN links. In this case, the MN is equipped with a single radio interface for mobile WLAN. The mobile WLAN can accommodate many MNs without excessive usage of the resource in the WWAN and the WWAN can provide long range wireless connection to the vehicle. In the integrated static WLAN and WWAN scenario, a MN communicates with a CN in the Internet over a pair of wireless links: static WLAN and WWAN links. In this case, the MN has multiple radio interfaces and can switch between WLAN and WWAN (i.e., vertical handover (VHO)). The static WLAN and WWAN respectively provide high data rate and wide coverage services to wireless Internet users. The remainder of this section describes our research problems, which arise in studying and improving the performance of WP-TCP in the integration scenarios above.

1.2.1 Performance Study

From the literature [2], the main techniques used in studying the performance are analytic modeling, simulation, and measurement. Generally, there is no single best technique for all. However, in the early stages of protocols and algorithms development, the use of analytic modeling technique in studying the performance is usually preferred. This is due to the fact that analytical models are relatively flexible, cost-effective, and can produce results with reasonable accuracy.

In this thesis, we develop frameworks for study the performance by using analytic modeling technique. To reduce the complexity and increase the flexibility, analytical frameworks for studying the performance of WP-TCP in integrated WLAN and WWAN are developed in two phases. In the first phase, the low level characteristics of wireless networks are modeled at the packet level. In the second phase, the performance of WP-TCP is analyzed by considering the packet level characteristics of wireless networks.

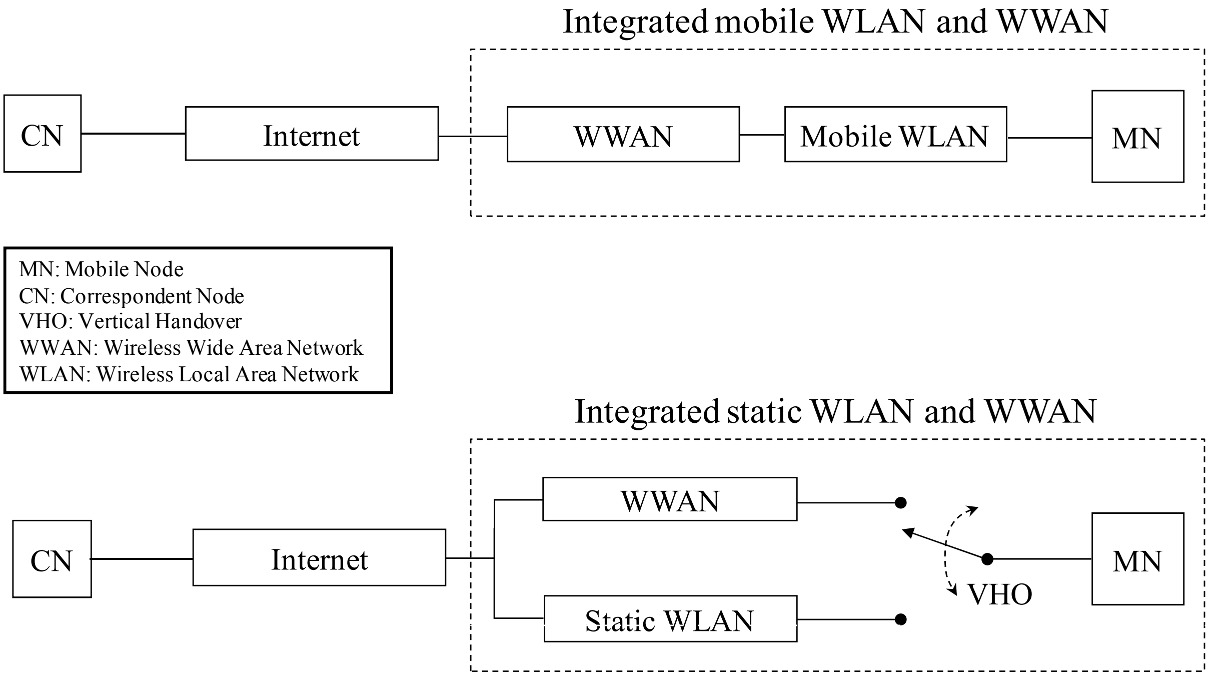

Integrated mobile WLAN and WWAN
CN
Internet
WWAN
Mobile WLAN
MN
MN: Mobile Node
CN: Correspondent Node
VHO: Vertical Handover
WWAN: Wireless Wide Area Network
WLAN: Wireless Local Area Network
Integrated static WLAN and WWAN
CN
Internet
WWAN
Static WLAN
VHO
MN

Figure 1.1. Integrated heterogeneous wireless networks scenarios

1.2.1.1 Modeling Packet Level Characteristics

In this phase, we need to establish appropriate analytical models which can describe characteristics of WWAN, WLAN, and WLAN-WWAN links (shown in Figure 1.1) at the packet level. In the literature there are several models [3] that can approximate the error process in WWAN links at the packet level and capture most of the wireless impairments, such as interference and fading. However, there are limited models suitable for WLAN and WLAN-WWAN links.

Recently, the effects of channel fading in WLANs have been reported in several studies [4], [5], [6]. It is revealed that WLANs have considerable transmission errors due to channel fading. The well-known analytical model proposed by Bianchi [7] and its variants [8], [9] are based on packet collisions. Hence, they do not capture the impact of transmission errors. To remedy this problem, several analytical models [10], [11] have been proposed to capture the impact of transmission errors. But, these models are based on an independent packet loss assumption which does not portray the wireless fading channel well.

Most of the channel models, proposed for multi-hop wireless links, are based on homogeneous wireless links. For this reason, they are not suitable for WLAN-WWAN links. Various research studies [12], [13], [14], [15], [16], [17], [18], [19], [20] have analyzed the performances over WLAN-WWAN links through simulations or measurements.

From the above survey, we therefore need to introduce new analytical models for WLAN model, which capture the effects of transmission errors due to channel fading. And then, based on the new WLAN and existing WWAN models, we need to derive mathematical expressions which present the packet loss probability and packet loss burst length over WWAN-WLAN links.

1.2.1.2 Modeling WP-TCP Performance

To the best of our knowledge, there is no analytical model in the open literature, which is specifically developed for WP-TCP protocol. Since the WP-TCP is based on the traditional TCP, previous analytical models for TCP may be used. Several analytical models have been proposed for TCP short-lived flows [21], [22], [23], [24]. Unfortunately, these models are based on an independent packet loss assumption which does not fit well in the wireless environment. Available analytical performance studies on TCP long-lived flows in [25], [26], [27], and references therein, only are develop for simple wireless or wireline links. And thus, they do not take into account the effects of vertical handovers. Most of the performance studies on TCP long-lived flows in integrated static WLAN and WWAN are conducted through simulation or experiments and live measurements [28], [29], [30].

In order to study the performance of WP-TCP short-lived flows with reasonable accuracy, packet level channel models which capture both correlated and uncorrelated transmission errors need to be considered. Also, the analytical model which describes short-term (during vertical handover) and long-term performances of WP-TCP long-lived flows need to be investigated. The analytical models should describe the behavior of WP-TCP long-lived flows under the influence of wireless channel errors, sudden change in network characteristics, and excessive packet losses due to vertical handovers.

1.2.2 Performance Improvement

WP-TCP has been adopted in WAP 2.0 to cope with the unpleasant wireless links characteristics. Howerver, when WP-TCP is deployed in integrated static WLAN and WWAN, its performance can dramatically degrade during and shortly after vertical handover due to premature timeouts or false fast retransmits.

After a vertical handover from WLAN to WWAN, round-trip time delay can suddenly and significantly increase. This can result in WP-TCP premature timeouts. The premature timeouts can degrade the WP-TCP performance by triggering unnecessary packet retransmission and congestion control response. During vertical handover from WWAN to WLAN, new data packets transmitted through WLAN can arrive at the WP-TCP receiver before the arrival of the old data packets transmitted through the WWAN (i.e., data packet reordering). Data packet reordering can generate a sufficiently large number of duplicate acknowledgement packets which may result in false fast retransmit. The false fast retransmit can degrade WP-TCP performance by triggering unnecessary packet retransmission and congestion control response.

In the literature, there are several schemes proposed to reduce the effects of premature timeouts and false fast retransmit. The schemes proposed in [31], [32], [33] detect falsely triggered fast retransmit or timeouts, and then undo the congestion control actions invoked. These schemes are proposed with the assumptions of homogeneous networks and they are reactive by nature. Recently, various proactive schemes, which consider heterogeneous wireless networks, have been proposed to specifically enhance the performance of TCP in vertical handover. However, these schemes are either effective only under mild vertical handover effects [34] or very complex to implement [35].

In order to improve the performance of WP-TCP, new solutions that thoroughly address the problem of premature timeouts and false fast retransmits due to vertical handovers need to be investigation. The solutions should be relatively simple, easy to implement, and compatible with standard TCP implementations.

1.3 Main Contributions

Our main objective in this thesis is to study and improve the performance of WP-TCP in integrated WLAN and WWAN. The models used to study the performance need to be flexible, cost-effective, and produce results with reasonable accuracy. Furthermore, the solutions proposed to improve the performance of WP-TCP need to be relatively simple, easy to implement, and compatible with standard TCP implementations.

To achieve our objective in studying the performance, we consider analytic technique. Firstly, we develop packet level channel models to facilitate the analysis of WP-TCP performance in integrated mobile WLAN and WWAN. We construct a novel analytical model which describes packet transmission over a fading channel for the IEEE 802.11-based WLAN link. The analytical model takes into account packet losses due to packet collisions and transmission errors. It captures the impact of both uncorrelated and correlated transmission errors. Furthermore, we develop an analytical model which presents the packet loss probability and packet loss burst length over WWAN-WLAN link. This model is suitable for describing the packet level characteristics of integrated mobile WLAN and WWAN.

Secondly, we develop an analytical framework for studying the performance of WP-TCP short- and long-lived flows. Contrary to other analytical frameworks, our analytical framework for WP-TCP short-lived flows is based on a two-state Markov chain and captures both correlated and uncorrelated packet losses. Our analytical framework for long-lived flows describe the short- and long-term performances of WP-TCP for given network and protocol parameters. It captures the effects of vertical handover, such as excessive packet losses and sudden change in network characteristics, experienced in integrated static WLAN and WWAN. Through the developed analytical frameworks, we extensively analyze performance of WP-TCP flows and investigate the WP-TCP design parameters.

Finally, we propose a cross-layer design and two proactive schemes called *RTT Inflation* and *RTT Equalization* at the mobile WP-TCP receiver. The *RTT Inflation* scheme suppresses the premature timeouts and the *RTT Equalization* scheme prevents the false fast retransmit. The proposed schemes significantly improve the performance during vertical handover in a wide range of practical WP-TCP parameters and network conditions. In addition, they are easy to implement and deploy, compatible with traditional TCP, and robust in the absence of perfect cross-layer information.

1.4 Organization of Thesis

The remainder of the thesis is organized as follows. Chapter 2 introduces the background material related to integrated heterogeneous wireless networks, WAP 2.0 architecture, and WP-TCP reliable transport mechanism.

In Chapter 3, we focus on establishing packet level channel models for WLAN, WWAN, and WLAN-WWAN links. We first construct two-state Markov models for WLAN and WWAN links to describe the packet transmission processes. Then, we extend the two-state Markov models to present the packet loss probability and packet loss burst length experienced over WLAN-WWAN link. We also conduct extensive simulations to verify the validity of the developed models.

In Chapter 4, we propose an analytical framework for studying the performance of WP-TCP short-lived flows based on a two-state Markov channel model. We first derive an explicit mathematical expression which yields a good estimate of WP-TCP transfer delay for a given network and protocol parameters. Then, we investigate the optimal WP-TCP design parameters for short-lived flows in various network conditions.

Chapter 5 presents an analytical framework for studying the performance of WP-TCP long-lived flows over the integrated static WLAN and WWAN. In this chapter, performance of WP-TCP long-lived flows is extensively analyzed and necessary conditions for the occurrence of premature timeout and false fast retransmits are investigated.

Based on the analysis presented in Chapter 5, a cross-layer design and proactive schemes are proposed in Chapter 6 to address the WP-TCP performance degradation due to vertical handover. Extensive simulation results are given to demonstrate the feasibility of the proposed solution and the significance of the performance gain.

Finally, we present a summary of our contributions and future research directions in Chapter 7.

1.5 Bibliographic Notes

Most of the work reported in this thesis has appeared in refereed research papers [36], [37], [38], [39], [40], [41], [42], [43], [44], [45]. The work in Chapter 2 has appeared in [36], Chapter 3 has appeared in [37], [41], [42], Chapter 4 has appeared in [38], [43], Chapter 5 has appeared in [39], [44], and Chapter 6 has appeared in [40], [45].

Chapter 2

Background

2.1 Heterogeneous Wireless Networks

Various wireless networks have been deployed to provide access to Internet applications and services as a tetherless alternative to wired networks. The deployed wireless networks include wireless personal area networks (WPANs) [46], WLANs, wireless metropolitan area networks (WMANs), and WWANs. The focus of this thesis is only on WLANs and WWANs. However, the material presented in this thesis can be easily extended to other wireless networks.

WLANs operate at the license-exempt industrial, scientific, and medical (ISM) frequency bands and can support high data rates. Generally, WLANs cover small geographic area and do not support terminal mobility over a wide area. In addition, they are mostly available within distinct areas called hotspots. The WLAN media access mechanisms are mostly based on the contention and therefore limited quality of services (QoS) are provided. Most of WLANs today are based on the IEEE 802.11x standards. The IEEE 802.11b (the most popular WLAN standard) can support data rate up to 11 Mbps and operate at 2.4 GHz band. The IEEE 802.11a and 802.11g can support data rate up to 54 Mbps and respectively operate at 5 GHz and 2.4 GHz bands. The coverage of IEEE 802.11x standards is up to few hundreds of meters around each access point (AP).

Cellular network is the dominant type of WWAN used for wireless data services. Currently, the popular cellular networks for data services include GPRS (General Packet Radio Service), UMTS (Universal Mobile Telecommunications System), and CDMA2000 systems. These networks operate in reserved frequency bands and support data rates from 10Kbps to 2Mbps. They usually cover a large geographical area (from few hundred meters to several kilometers) and support terminal mobility by using several techniques such as location updating, paging, and handover. Contrary to WLAN, cellular network can provide guaranteed QoS. However, cellular network infrastructure is more expensive than WLAN infrastructure and so the charge for using the network.

Table 2.1 summarizes the characteristics of WLAN and WWAN. Generally, there is no single wireless network with all best characteristics to support diverse wireless Internet application and services. However, the integration of WLAN and WWAN presents complementary and attractive characteristics which can support a wide range of wireless Internet applications and services. Consequently, the integrated WLAN and WWAN access networks are introduced. WWAN can be integrated with WLANs installed in buildings/open areas such as convention centers, cafes, airports, shopping malls (static WLAN) or in mobile platforms such as trains, buses, ships, ferries (mobile WLAN). The remainder of this subsection discusses typical network architectures for both integrated static WLAN and WWAN and integrated mobile WLAN and WWAN.

Table 2.1. Characteristics of WLAN and WWAN

Characteristic	WLAN	WWAN
Data rate	High	Low
Coverage area	Small	Large
Frequency bands	Unlicensed	Licensed
Mobility support	Poor	Excellent
Service charge	Cheap	Expensive
QoS support	Poor	Excellent

2.1.1 Integrated Mobile WLAN and WWAN

Figure 2.1 describes a typical network architecture for integrated mobile WLAN and WWAN. Within a moving vehicle, the WLAN is used to connect a number of mobile nodes (MNs) to an access point (AP) or gateway. The WWAN is employed for the connection between the AP and the base station (BS), which is in turn connected to the Internet through a wireline link. Packets sent from a correspondent node (CN) to an MN are first routed to the BS through the Internet, and then transmitted to the MN over the WWAN-WLAN link.

The integration of mobile WLAN and WWAN has several advantages. First, WLAN supports higher data rate than WWAN but it has a smaller service coverage than WWAN. By integrating these two technologies, WWAN provides an extended service coverage to the moving vehicle, and WLAN accommodates more users without excessive usage of the resource in WWAN. Second, the aggregated traffic at the AP is transmitted to the BS through an antenna mounted on top of the vehicle which has better communication channels to the BS as compared to a channel between the BS and MNs. Third, the AP has less energy constraint than the MNs. Using the AP to relay the data to the BS can significantly save the energy consumption in MNs.

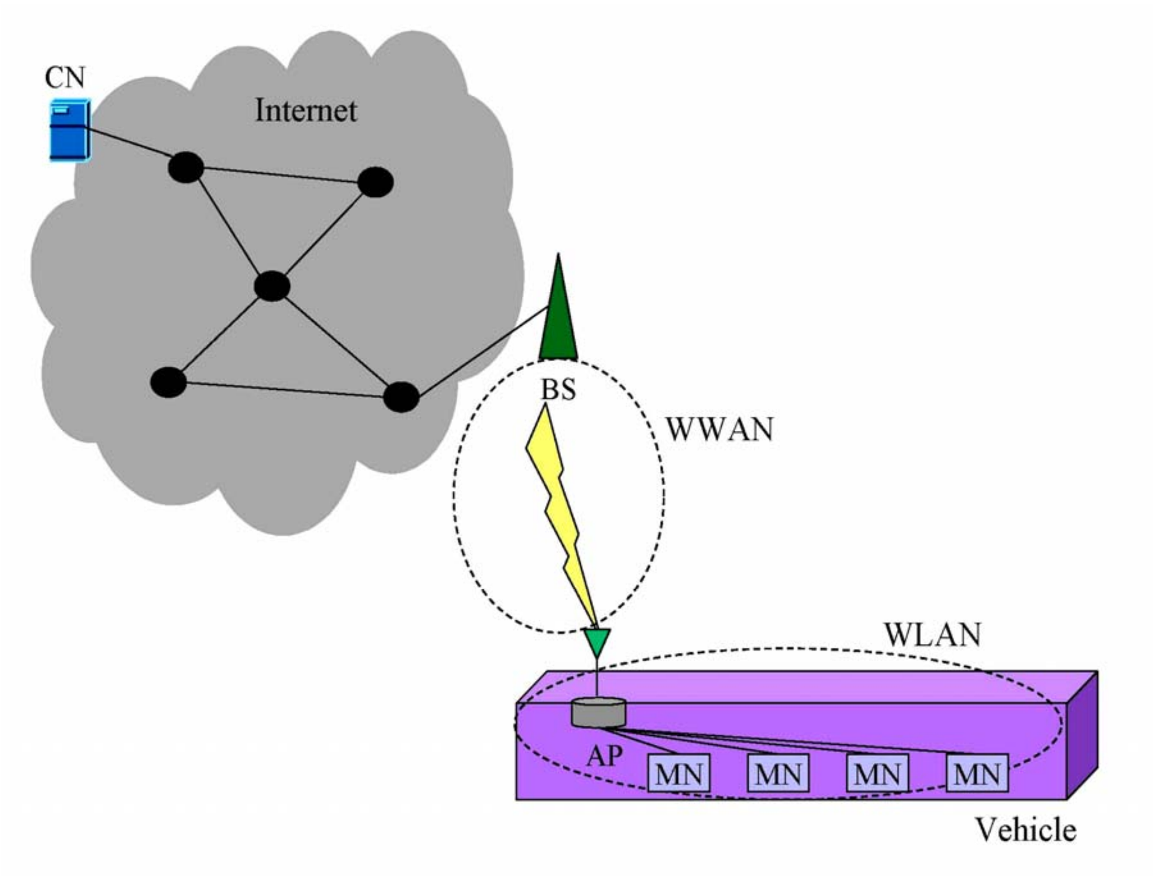


Figure 2.1. Architecture for integrated mobile WLAN and WWAN

2.1.2 Integrated Static WLAN and WWAN

The network architecture for integrated static WLAN and WWAN is depicted in Figure 2.2. A MN with multiple radio interfaces can communicate with a CN in the Internet through an AP in WLAN or a BS in WWAN. In this network environment the MN can switch between WLAN and WWAN. Note that a standard Internet protocol such as Mobile IPv6 [47] is required at the network layer in order to support mobility between WLAN and WWAN. An event where the MN switches from one network to another network is called vertical handover and average time that the MN stays in one network before switches to another network is called network residence time. The vertical handover from the WLAN to the WWAN is referred to as upward vertical handover and the vertical handover from the WWAN to the WLAN is referred to as downward vertical handover. The vertical handover can also be hard or soft. Hard vertical handover is when an old connection is broken before a new connection is made and soft vertical handover is when a new connection is made before an old connection is broken.

The integration of static WLAN and WWAN presents several advantages. The integration can provide a means for load balancing, where the WLAN can offload the traffic with elastic QoS requirements from the WWAN. Furthermore, it can offer less expensive extended service coverage with the improved data rate to wireless Internet users.

2.2 Wireless Application Protocol

Wireless Application Protocol (WAP) has been attracting a lot of attention recently, as it provides a standard environment for wireless mobile communications [48]. WAP is proposed by the WAP Forum [49] (now managed by Open Mobile Alliance) to enable mobile users with digital hand-held devices to access the Internet and advanced telephony services. It is a de-facto world standard for the presentation and delivery of wireless information services on wireless devices. The earlier versions of WAP (1.0, 1.1, 1.2, 1.2.1) are standards optimized for mobile environment where hand-held wireless devices are limited by CPU power, memory, battery lifetime and simple user interface. As wireless networks and mobile devices evolve, some of these constraints become less significant. The WAP version 2 (WAP 2.0) standard, released in July 2001, utilizes the advanced technology in wireless network and mobile devices. Unlike WAP 1.x standards, WAP 2.0 supports standard Internet protocols.

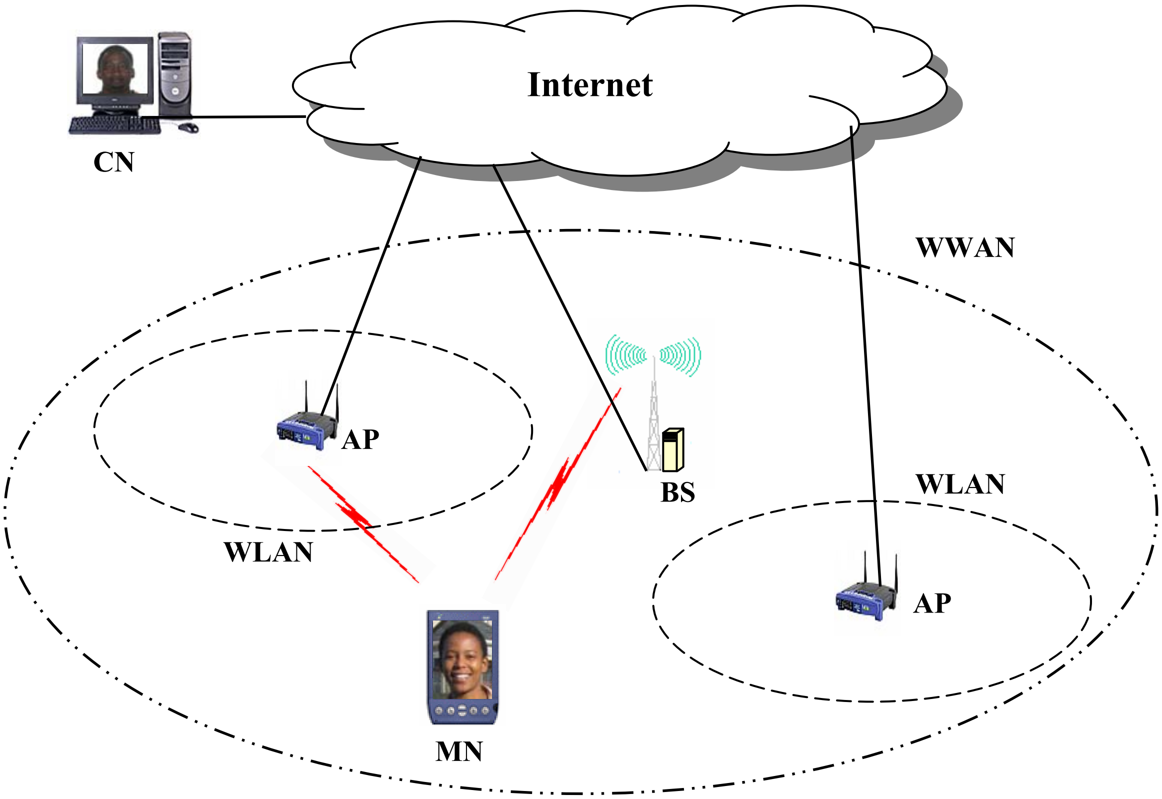


Figure 2.2. Architecture for integrated static WLAN and WWAN

Figure 2.3 shows WAP 2.0 architecture. Similar to the Open Systems Interconnection (OSI) reference model, WAP 2.0 has a layered architecture, with an application framework and a protocol framework. The application framework is built on top of the protocol framework and provides an interoperable environment. It allows the applications and services to be built on a wide variety of different wireless platforms. The protocol framework has four layers: a session services layer, a transfer services layer, a transport services layer, and a bearer networking layer. The transport services layer provides datagram service and connection service. User datagram protocol (UDP). Wireless datagram protocol (WDP) are used to provide datagram transport service and wireless profiled transmission control protocol (TCP) is used to provide connection-oriented reliable transport service. Figure 2.4 shows the WAP 2.0 wireless/Internet internetworking model with a typical protocol stack for reliable data delivery. In the wireless domain, the wireless profiled TCP (WP-TCP) is used for communication between WAP device and proxy. In the wireline domain, the standard TCP is used for communication between web server and the proxy. In this model, WAP 2.0 utilizes proxy technology to connect the wireless domain and the Internet. The remainder of this chapter focuses on WP-TCP. A complete overview of WAP 2.0 architecture can be found in [50].

2.3 Wireless Profiled TCP

WP-TCP [51] is the key reliable transport protocol that supports attractive applications such as mobile browsing, emails, and file downloads. It is optimized for wireless links and it can interoperate with standard TCP implementations. WP-TCP protocol specifications have mandatory and optional requirements. All WP-TCP implementations are mandated conform to the requirements for TCP implementation described in Request for Comments (RFC) 0793 [52] and RFC

1122 [53], TCP congestion control specified in RFC 2581 [54], and selective acknowledgement (SACK) given in RFC 2018 [55]. In addition to the mandatory requirements, WP-TCP implementations may also conform to the requirements specified for window scale option given in RFC 1323 [56], timestamps option for RTTM given in RFC 1323, large initial window given in RFC 2414 [57], path MTU discovery given in RFC 1191 [58] and RFC 1981 [59], explicit congestion notification (ECN) given in RFC 2481 [60], and large window size based on bandwidth-delay product (BDP). In the next subsections, we first present fundamentals of TCP and then summarize the WP-TCP mechanisms. Finally, we discuss the performance of WP-TCP in integrated WLAN and WWAN.

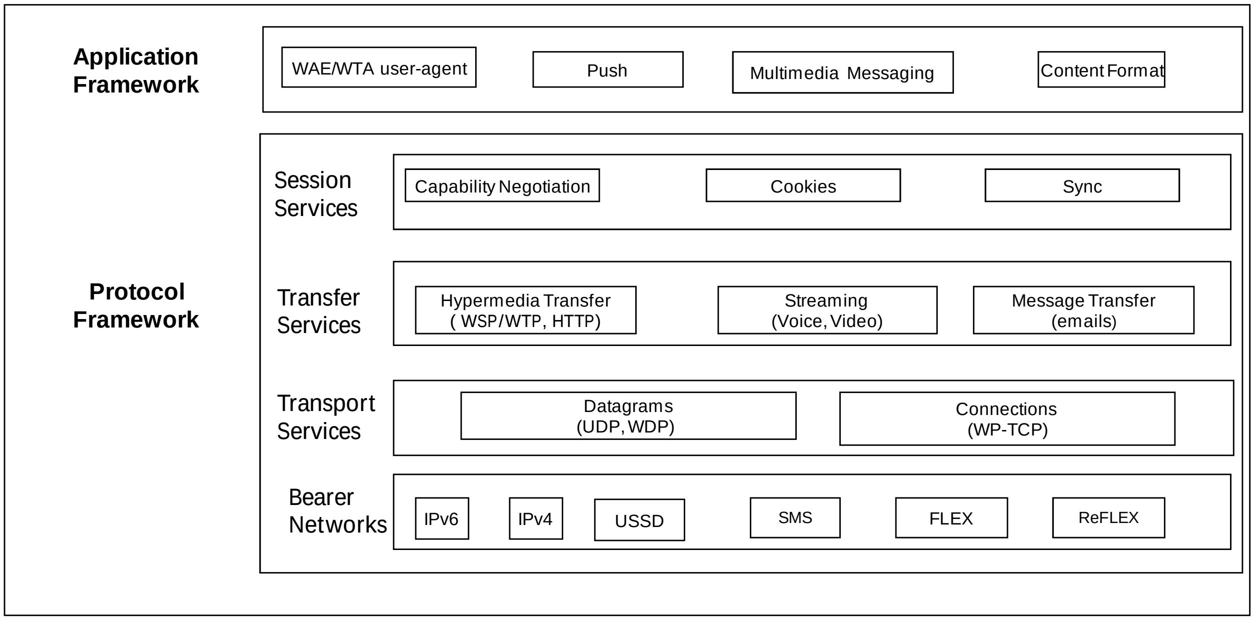


Figure 2.3. WAP 2.0 architecture

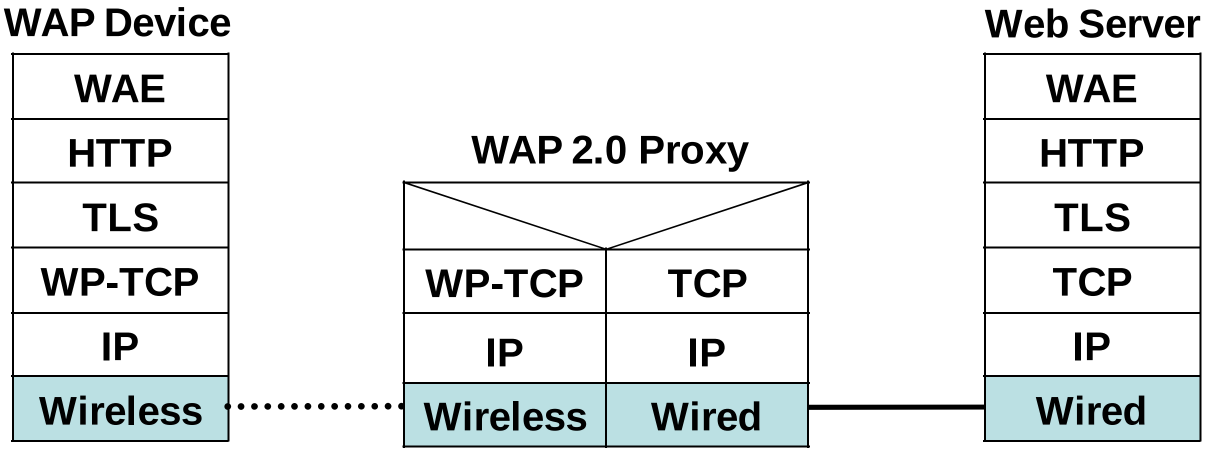


Figure 2.4. WAP 2.0 wireless/Internet interworking model

2.3.1 TCP Basics

TCP is the most widely used connection-oriented transport protocol on the Internet. It provides fully reliable, in-order, byte-stream, end-to-end data delivery services by using connection management, congestion and flow control, and loss recovery mechanisms.

Connection Management: TCP has a complex connection management mechanism. It first uses 3-way handshakes to open a data channel for data transfer. That is, the TCP sender sends a SYN packet, the TCP receiver replies with a corresponding SYN-ACK packet, and the TCP sender replies with an ACK packet. After opening the data channel, data packets can be transmitted in both directions. After transferring all data packets, the TCP sender and receiver use 2-way handshakes to close the pre-established data channel. That is, the TCP sender sends a FIN packet, the TCP receiver replies with a corresponding ACK packet. Then, immediately the TCP receiver sends a FIN packet, the TCP sender replies with a corresponding ACK packet. Note that the ACK packet sent by the TCP sender in the 3-way handshakes is usually piggy-backed in the first data packet.

Congestion and Flow control: Schemes that TCP uses to control network congestion can be categorize into two types: window-based schemes and rate-based schemes. Since WP-TCP uses the window-based scheme, the rate-based schemes are not discussed in this chapter. In window-based schemes, TCP sender maintains a congestion window that regulates number of unacknowledged data packets in the network. Each sent packet consumes a slot in the congestion window and the sender can send a packet only if a free slot is available in the congestion window. When an acknowledgement for outstanding packet(s) is received, the window is shifted and free slot(s) become available. On the other hand, TCP uses slide window (in sequence space) to implement a flow control between sender and receiver. The TCP receiver notifies the TCP sender the amount of free space available in the receiver's buffer by using an advertised window. The TCP sender performs congestion and flow control by ensuring that the transmission window does not exceed the congestion window and receiver's advertised window size.

Loss Recovery: TCP uses timer-driven and data-driven retransmissions mechanisms to discover and recover packet losses. The data-driven recovery mechanism relies on feedback from cumulative acknowledgements (ACKs). When a packet is lost, the receipt of all later packets generates duplicate ACKs to the TCP sender. The TCP sender can then detect the lost packet and retransmit it when the number of successfully received duplicate ACKs exceeds a predetermined threshold. In timer-driven recovery mechanism, the TCP sender maintains the timer. If a positive cumulative ACK for a packet is not received within a certain timeout interval, the TCP sender retransmits the missing packet and backoff exponentially after each unsuccessful retransmission. The timeout interval is normally determined by using an estimated round-trip time.

2.3.2 WP-TCP Mechanisms

WP-TCP uses basic TCP mechanisms for connection management and flow control. However, it utilizes modified TCP mechanisms for congestion control and loss recovery and enhanced procedure for measuring round-trip times (RTT) and computing retransmission timeouts (RTO).

Congestion control and loss recovery mechanisms: The WP-TCP sender maintains a congestion window, which limits the number of outstanding unacknowledged data packets in the network, and a slow start threshold, which determines the rate of adjusting the congestion window. On startup, congestion window and slow start threshold are initialized to the initial window and maximum window size, respectively. Whenever a new ACK is received, the congestion window is increased by one packet if it is below the slow start threshold (slow start phase) and by 1/(congestion window) if it is equal to or greater than slow the start threshold (congestion avoidance phase). In either phase, the upper limit of congestion window increase is the maximum window size. The WP-TCP sender assumes a packet is lost either after a timeout or after receiving a certain number of consecutive duplicate ACKs (ACK with the sequence number same as the previous ACK). This number is normally referred to as the duplicate ACK threshold. When a timeout occurs, the slow start threshold is set to the larger of {2, (congestion window)/2} and the congestion window is reset to one. The lowest unacknowledged packet is retransmitted and the WP-TCP sender enters the slow start phase. In the case of packet loss, indicated by duplicate

ACKs, fast retransmit is invoked followed by fast recovery. The fast recovery procedure ensures that the congestion avoidance phase follows after fast retransmit and not the slow start phase.

RTT Measurement and RTO computation: Whenever a data packet is sent, the WP-TCP sender places a timestamp in the Timestamp Value field (*TSVal*) in the data packet. The timestamp value is obtained from a local clock. When a new in-order data packet is received at the WP-TCP receiver, the timestamp value from *TSVal* is stored in a local variable *TS.recent*. Whenever an acknowledgement packet is sent, the WP-TCP receiver copies the current timestamp stored in *TS.recent* to the Timestamp Echo Reply field (*TSecr*) in the acknowledgement packet. When a valid acknowledgement packet arrives at the WP-TCP sender, the sample round-trip time (*RTT*) is computed as a difference between the current time of the local clock and the timestamp echoed in the *TSecr*. For every *RTT*, sampled at the WP-TCP sender, retransmission timeout (*RTO*) is computed by using the following procedural steps.

$$RTTVAR \leftarrow (1-\beta)\cdot RTTVAR + \beta \cdot \left| SRTT - RTT \right| \tag{2.1}$$

and

$$SRTT \leftarrow (1-\alpha)\cdot SRTT + \alpha \cdot RTT, \tag{2.2}$$

where *SRTT* (smoothed round-trip time) and *RTTVAR* (round-trip time variation) are state variables, and α = 1/8 and β = 1/4. The WP-TCP sender updates the *RTO* as

$$RTO \leftarrow SRTT + \max\left\{G, k\cdot RTTVAR\right\}, \tag{2.3}$$

where *G* is a clock granularity and *k* = 4.

2.3.3 WP-TCP Performance

Standard TCP tends to perform poorly over wireless links [61]. Therefore, WP-TCP has been adopted in WAP 2.0 to cope with the unpleasant wireless links characteristics. However, when WP-TCP is deployed in an integrated WLAN and WWAN, its performance can dramatically degrade due to vertical handover, which can result in bandwidth-delay product mismatch, premature timeouts, inrush packet transmission, false fast retransmit, and burst packet losses.

During hard upward or downward vertical handover, all or a large number of in-flight data and acknowledgement packets can get lost. This can result in burst losses of data packets which can only be recovered by a costly timeout mechanism. Furthermore, the burst losses of acknowledgment packets can lead to unnecessary stalling of the transmission process.

WP-TCP uses a congestion window to control the number of in-flight packets, which is ideally equivalent to the bandwidth-delay product of the network path. The congestion window is limited to the TCP maximum window and adjusted according to the additive increase and multiplicative decrease (AIMD) congestion control mechanism [62]. Since the congestion window is usually adjusted slowly, any abrupt change in bandwidth-delay product after vertical handover can result in congestion of the bottleneck link or underutilization of the new network path.

During soft upward vertical handover, the WP-TCP sender continues to transmit data packets through WWAN at the same (or higher) rate of receiving acknowledgement packets from the WLAN. Since the WLAN has relatively higher bandwidth as compared to the WWAN, the inrush data packets transmission in WWAN can lead to temporary congestion and packet losses which in turn degrade the WP-TCP performance. After soft upward vertical handover the round-trip time delay can suddenly and significantly increase, This can result in WP-TCP premature timeouts. The premature timeouts can degrade the WP-TCP performance by triggering unnecessary packet retransmission and congestion control response.

When soft downward vertical handover occurs, new data packets transmitted through WLAN can arrive at the WP-TCP receiver before the old data packets transmitted through the WWAN (i.e., data packet reordering). The data packet reordering can generate a sufficiently large number of duplicate acknowledgement packets which may result in false fast retransmit. The false fast retransmit can degrade the WP-TCP performance by triggering unnecessary packet retransmission and congestion control response.

Chapter 3

Channel Model for Integrated WLAN and WWAN

3.1 Introduction

One of the main issues in studying the performance of transport mechanisms in the scenarios presented in Section 1.2 is the availability of appropriate packet level models for wireless links. Consequently, in this chapter, we focus on establishing packet level models for WWAN, WLAN, and WLAN-WWAN links.

The effects of channel fading in WLAN have been reported in several studies [4], [5], [6]. It is revealed that WLAN has considerable transmission errors due to channel fading. In the literature, there is a limited packet level WLAN models which capture the effects of transmission errors due to channel fading. The well-known analytical model proposed by Bianchi [7] and its variants [8], [9] do not capture the impact of transmission errors. To remedy this problem, several analytical models [10], [11] have been proposed to capture the impact of transmission errors. However, these models are based on an independent packet loss assumption which does not portray the wireless fading channel well. For WWAN, there are several models [3] which can be used to describe the error process at the packet level. These models can capture most of the wireless impairments, such as interference and fading. To the best of our knowledge there is no reasonable packet level channel model for WWAN-WLAN link in the open literature. Various research studies have analyzed the performances of integrated mobile WLAN and WWAN [12], [13], [14], [15], [16], [17], [18], [19], [20], but most of the studies are based on simulations or measurements. Also, available packet level channel models for multi-hop wireless links are mostly based on homogeneous multi-hop wireless links. Therefore, they cannot be used to present characteristics of the WLAN-WWAN links at the packet level.

In order to facilitate the performance analysis of the transport mechanisms in integrated WLAN and WWAN, we construct a new analytical model for WLAN link, which captures the impact of both uncorrelated and correlated transmission errors. Then, based on the new WLAN link and existing WWAN link models, we develop an analytical model for WWAN-WLAN link, which describes the packet loss probability and packet loss burst length. Finally, we conduct extensive analysis, which shed some light on the design of transport mechanisms over integrated mobile WLAN and WWAN.

The remainder of this chapter is organized as follows. Section 3.2 develops packet level models for WLAN, WWAN, and WLAN-WWAN links. Extensive simulation and analytical results are presented in Section 3.3 and the summary of this chapter is given in Section 3.4.

3.2 Packet Level Wireless Links Models

In this section, we construct packet level models for WLAN, WWAN, and WLAN-WWAN links. For WLAN, N saturated MNs (i.e., each MN always has a packet to send) are assumed to share the WLAN channel and transmission failures are due to either packet collision or channel fading. The IEEE 802.11 distributed coordination function (DCF) [63] is used for media access control. For WWAN, we consider a dedicated fading channel. Therefore, transmission errors are only due to channel fading of the received signal strength. In addition, a truncated automatic repeat request (ARQ) scheme is used to recover transmission failure. In order to approximate the correlated transmission error behavior in the WLAN, WWAN, and WLAN-WWAN links, we use a discrete-time two-state Markov channel model [64], which has a good state and a bad state. In the good state a packet is transmitted successfully and in the bad state a packet is transmitted unsuccessfully. The details on a two-state Markov channel model constructed for particular link are presented in the next subsections. For easy reference, the notions used frequently in this chapter are listed in Table 3.1.

3.2.1 WWAN Link Model

Our study is based on urban areas (Section 1.2), where there are practically no paths of direct transmission between MNs and a BS in the WWAN. We consider that slowly varying fading due to the distance or shadowing is compensated by the power-controlled transmission and packet transmission rate (in packets/seconds) much higher than the maximum Doppler frequency (Hz). Consequently, at the data link layer, it is reasonable to consider a non-line-of-sight (NLOS) frequency-nonselective (flat) slow fading wireless channel. This type of fading channel can be modeled as a multiplicative complex Gaussian random process $c(t)$ with zero mean and covariance function $K(\tau) = J_0(2\pi f_d\,|\tau|)$ [65], where $J_0(\cdot)$ is the zeroth order Bessel function of the first kind, $f_d = vf_c/v_c$ is the maximum Doppler frequency, v is the velocity of a vehicle, f_c is the carrier frequency, and v_c is the speed of light. The corresponding power spectrum for the Gaussian process is given by

$$S_X(f) = \begin{cases} S_X(0)\left[1-\left(\dfrac{f}{f_d}\right)^2\right]^{-1/2}, & |\,f\,|< f_d \\ 0, & \mathit{otherwise.} \end{cases} \tag{3.1}$$

The envelope $\alpha(t)$ defined as $|c(t)|$ is Rayleigh-distributed at any time t. This model has been widely accepted as it has been proven to achieve good predictions of measured signal statistics [65]. Given a modulation scheme, the dynamics of the fading channel can be characterized at the packet level. However, the performance analysis of high-level protocols becomes quite complex. As an alternative to this problem, a widely adopted two-state Markov channel model [3], with good state g and bad state b, is used to approximate the error process at the packet level. The time slot duration for the Markov chain equals the packet transmission time in WWAN. Let Γ be the fading margin. That is, the maximum fading attenuation that the system can tolerate. Then, the average transmission error probability is given as [3]

$$\pi_b^{WWAN} = 1 - e^{-1/\Gamma}. \tag{3.2}$$

Let $\mathbf{P} = \begin{pmatrix} p_{bb} & p_{bg} \\ q_{gb} & p_{gg} \end{pmatrix}$ be the WWAN link state transition matrix, where p_{xy} is the transition probability from state $x \in \{b, g\}$ to state $y \in \{b, g\}$. The state transition probabilities can be found as

Table 3.1. List of local notions for Chapter 3

Notation	Definition (unit)
f_c	carrier frequency (Hz)
v	velocity of a vehicle (m/sec)
v_c	speed of light (299 792 458 m/sec)
Γ	fading margin: maximum fading attenuation that the system can tolerate (dB)
l_B	average number of consecutive WLAN time slots in the bad state
π_b^X	wireless link $X \in \{WLAN, WWAN\}$ packet loss probability, below data link layer
π_F^X	wireless link $X \in \{WLAN, WWAN\}$ packet loss probability, above data link layer
l	retransmission retry limit of the truncated ARQ scheme for the WWAN
m	retransmission retry limit of the IEEE 802.11 MAC scheme for the WLAN
P_l	probability of packet loss over WWAN-WLAN link
L_b	average number of consecutive packet losses over WWAN-WLAN link
N	number of mobile nodes in WLAN.
$\mathbf{P} = [p_{XY}]$	transition matrix of the two-state Markov chain for WWAN link, where an element p_{XY} denotes the transition probability from state X to state Y
$\mathbf{Q} = [q_{XY}]$	transition matrix of the two-state Markov chain for WLAN link, where an element q_{XY} denotes the transition probability from state X to state Y
$\mathbf{R} = [r_{XY}]$	transition matrix of the two-state Markov chain for WLAN-WWAN link, where an element r_{XY} denotes the transition probability from state X to state Y

$$p_{bb} = 1 - \frac{Q(\theta, \rho\theta) - Q(\rho\theta, \theta)}{e^{1/\Gamma} - 1} \tag{3.3}$$

and

$$p_{gg} = \frac{1 - \pi_b^{WWAN}(2 - p_{bb})}{1 - \pi_b^{WWAN}}, \tag{3.4}$$

where $\theta = \sqrt{\frac{2/\Gamma}{1-\rho^2}}$ and $\rho = J_0(2\pi f_d D)$ is the Gaussian correlation coefficient of two samples of the complex amplitude in a fading channel, which are sampled D seconds apart. $Q(\cdot,\cdot)$ is the Marcum Q function. Note that $p_{gb} = 1 - p_{gg}$ and $p_{bg} = 1 - p_{bb}$.

By deploying the truncated ARQ scheme, a sender retransmits a packet until the packet is successfully delivered, or dropped the packet if the retry limit *l* (including the first transmission) is reached [66]. Consequently, the probability that a packet is lost in the WWAN link can be found as

$$\pi_F^{WWAN} = \pi_b^{WWAN} p_{bb}^{l-1}. \tag{3.5}$$

3.2.2 WLAN Link Model

A time slot in the Markov chain, *E*[*slot*], is defined as the time interval between two consecutive backoff counter decrements [7], and it can be found in Appendix A. In general, the upper bound of *E*[*slot*] equals the transmission time of a packet. The transmission time of a 1000-byte packet over the WLAN with 11 Mbps channel is less than 1 ms. On the other hand, a relative movement speed of the communicating devices or other objects in in-vehicle WLANs is usually very low. In practice, the channel coherence time can be approximated as 42.3% of multiplicative inverse of the maximum Doppler frequency (a more conservative approximation) [67]. The maximum Doppler frequency of the fading channel, with user velocity of 5 m/s (18 km/hour) and carrier frequency of 2.4 GHz, is about 40 Hz. In that case, the channel coherence time is more than 10 ms. From this observation and other research [68], $E[slot]$ is generally much less than the WLAN channel coherence time. Therefore, it is reasonable to consider the state transitions of the Markov channel model after every time slot. In the bad state, a packet transmission fails while a packet transmission is successful in the good state if there is no collision. The transition probabilities of these states are given by the matrix

$$\mathbf{Q} = \begin{pmatrix} q_{bb} & q_{bg} \\ q_{gb} & q_{gg} \end{pmatrix}, \tag{3.6}$$

where q_{xy} is the transition probability from state $x \in \{b, g\}$ to state $y \in \{b, g\}$. Given an average error rate (π_b^{WLAN}), defined as average transmission error probability (the ratio of the number of erroneous packets to the total number of packets sent), and a burst length (l_B), defined as the average number of consecutive time slots in the bad state, the transition probabilities can be computed as

$$q_{bg} = \frac{1}{l_B} = 1 - q_{bb} \tag{3.7}$$

and

$$q_{gb} = \frac{\pi_b^{WLAN} q_{bg}}{1 - \pi_b^{WLAN}} = 1 - q_{gg}. \tag{3.8}$$

The fading parameters l_B and π_b^{WLAN} can be obtained from measurements or from wireless channel profiles by considering the physical characteristics of the channel and the physical layer modulation/coding schemes used [6].

Figure 3.1 illustrates the transmission success/failure behaviors in the IEEE 802.11 basic access mode[1]. For tactical analysis, it is assumed that a packet transmission is successful when channel condition is good (*g*) and no collisions (*coll*) occur at the transmission time slot. A maximum of *m* retransmissions are attempted before the packet is aborted (i.e., after *m*+1 transmission failures). For every transmission failure, the sender defers its transmission for a random duration, i.e., backoff interval. In Figure 3.1, τ_{i+1} denotes the time slot where a packet transmission occurs at the *i*th backoff stage $(0 \le i \le m)$, where *m* is the transmission retry limit. Since the backoff stage length (in numbers of *E*[*slot*]) is uniformly chosen in the range $[0, W_i - 1]$, where $W_i = 2^i CW_{\min}$ and $CW_{\min}$ is the minimum contention window size, the average length of the *i*th backoff stage (n_i) can be computed as $n_i = (W_i - 1)/2$.

The transition probability from state $x \in \{b, g\}$ at time slot τ_i to state $y \in \{b, g\}$ at time slot τ_{i+1}, denoted by $q_{xy}^{(n_i)}$, can be obtained from elements in the n_i-step transition probability matrix,

$$\mathbf{Q}^{(n_i)} = \begin{pmatrix} q_{bb}^{(n_i)} & q_{bg}^{(n_i)} \\ q_{gb}^{(n_i)} & q_{gg}^{(n_i)} \end{pmatrix}. \tag{3.9}$$

As indicated before, a packet transmission fails when the channel is bad after winning channel contention or when a collision occurs in the good state. Since a packet loss occurs after *m*+1 consecutive transmission failures, there are 2^{m+1} possible unique sequences (traces) of the WLAN channel states for a packet loss event. Let $T_i \in \{(s_0, s_1, ..., s_m)\}$ denote the *i*th trace of all possible

[1] Even though another access mode, request-to-send (RTS)/clear-to-send (CTS), is efficient to overcome the hidden-node problem, it is disabled in most products available in the current market [25]. Therefore, we assume the basic access mode.

traces, where $s_k \in \{b, g\}$ is the WLAN channel state in the kth transmission attempt (see Figure 3.1). Then, the packet loss probability in the WLAN link is given by

$$\pi_F^{WLAN} = \sum_{i=1}^{2^{m+1}} \Theta(T_i) \cdot p^{N(T_i)}, \tag{3.10}$$

where p is the probability that a collision occurs when a packet is transmitted, $N(T_i)$ is the number of transmission attempts in the good channel state (or the number of packet collisions) in the trace T_i, and $\Theta(T_i)$ is the occurrence probability of trace $T_i = (s_0, s_1, ..., s_m)$ which can be found as

$$\Theta(T_i) = \pi_b^{WLAN} q_{bs_0}^{(0)} q_{s_0 s_1}^{(1)} ... q_{s_{m-1}s_m}^{(m)} + \pi_g^{WLAN} q_{gs_0}^{(0)} q_{s_0 s_1}^{(1)} ... q_{s_{m-1}s_m}^{(m)}. \tag{3.11}$$

$q_{xy}^{(i)}$ (for $x, y \in \{b, g\}$) represents $q_{xy}^{(n_i)}$ in (3.9), and $\pi_b^{WLAN} = q_{gb} \big/ (q_{gb} + q_{bg})$ and $\pi_g^{WLAN} = q_{bg} \big/ (q_{gb} + q_{bg})$ are the steady state probabilities that the WLAN link conditions are bad and good, respectively.

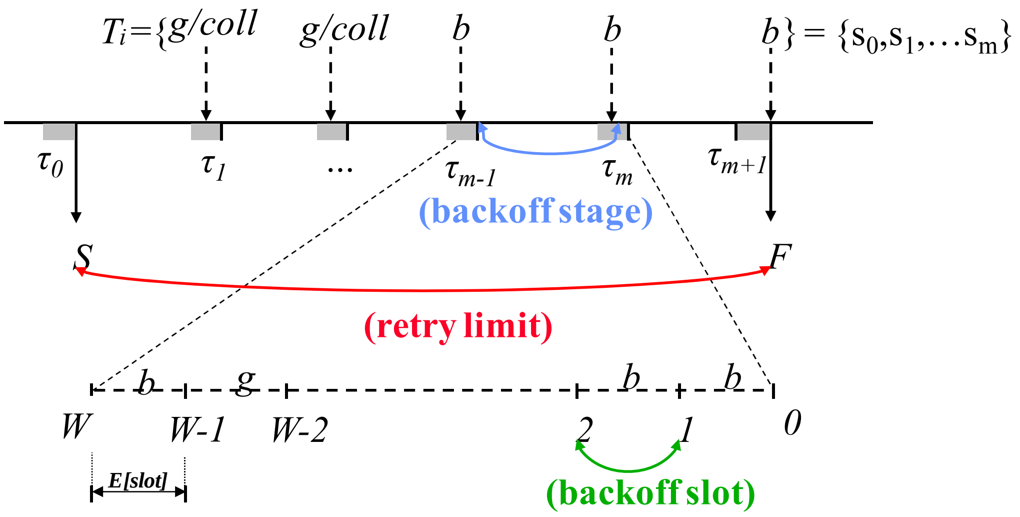


Figure 3.1. Transmission success/failure behaviors in the IEEE 802.11 WLAN

Let $bw(t)$ and $bs(t)$ respectively denote the stochastic processes representing the backoff window size and the backoff stage for a given node at time t [8]. The packet transmission behavior can be described by a two-dimensional process as $\{bs(t), bw(t)\}$. Let $b_{i,k}$ be the stationary distribution of the Markov chain, i.e., $b_{i,k} = \lim_{t\to\infty} P\{bs(t) = i, bw(t) = k\}$. τ is the probability that a packet is transmitted in a randomly chosen slot and it is given by [8]

$$\tau = \sum_{i=0}^{m} b_{i,0} = \frac{1 - p_f^{m+1}}{1 - p_f} b_{0,0}, \tag{3.12}$$

where p_f is the packet transmission failure probability in a given slot. $b_{0,0}$ is given by

$$b_{0,0} = \begin{cases} \dfrac{2(1-2p_f)(1-p_f)}{W(1-(2p_f)^{m+1})(1-p_f) + (1-2p_f)(1-p_f^{m+1})}, & m \le m' \\ \dfrac{2(1-2p_f)(1-p_f)}{W(1-(2p_f)^{m'+1})(1-p_f) + (1-2p_f)(1-p_f^{m+1}) + W 2^{m'} p_f^{m'+1}(1-2p_f)(1-p_f^{m-m'})}, & m > m', \end{cases} \tag{3.13}$$

and m' is the number of contention window sizes (i.e., the maximum contention window size is $2^{m'}$) and W is the minimum contention window size.

To compute the packet transmission failure probability p_f, we use the same assumptions and notations as in [7], [8]. Even though we consider burst transmission errors in the derivation of π_F^{WLAN}, it is quite complicated to consider burst transmission errors in the derivation of p_f. Therefore, we assume that p_f is affected by independent transmission errors. This approximation can be acceptable since p_f mainly captures the average transmission failure rate. For instance, if we randomly sample an error sequence (i.e., τ_i in Figure 3.1), the average error rate of the sampled sequence remains the same as the original error rate and it is independent of the self-correlated coefficient of the original sequence. Furthermore, the transmission probability in the

n_i-step transition matrix approaches the steady state probability (i.e., $q_{xy}^{(n_i)} \approx \pi_y^{WLAN}$) when n_i is sufficiently large. Note that the contention window size increases exponentially and CW_{min} is 32 in the IEEE 802.11b specification, which is sufficiently large for justifying the approximation. In Section 3.3, we will validate this approximation through simulation. Given N saturated nodes, the collision probability p is given by $p = 1-(1-\tau)^{N-1}$. Then, the packet transmission failure probability, p_f, in a given slot can be approximated as

$$p_f \approx \pi_b^{WLAN} + (1-\pi_b^{WLAN})p. \tag{3.14}$$

Using (3.12), (3.13), and (3.14), p_f and p can be computed numerically. Once we obtain the packet transmission failure probability p_f and the collision probability p, we can use (3.10) to calculate the packet loss probability in WLAN, considering correlated channel fading.

3.2.3 WLAN-WWAN Link Model

Since downlink traffic dominate uplink traffic for many Internet applications (e.g., web browsing and file downloads), we focus on the downlink transmission. That is, a packet is first transmitted over the WWAN link and then over the WLAN link. In this Subsection, we consider the models for WWAN and WLAN, respectively presented in Subsection 3.2.1 in Subsection 3.2.2, and develop a two-state Markov model for WLAN-WWAN link.

Let us consider a two-state Markov chain with failure (F) and success (S) states. A packet is successfully transmitted in state S and dropped in state F. Let r_{xy} be the transition probability from state $x \in \{F,S\}$ to state $y \in \{F,S\}$. Then, the transition matrix of the Markov chain is given by

$$\mathbf{R} = \begin{pmatrix} r_{FF} & r_{FS} \\ r_{SF} & r_{SS} \end{pmatrix}, \tag{3.15}$$

where $r_{FS} = 1 - r_{FF}$ and $r_{SF} = 1 - r_{SS}$. The packet loss burst length (L_b), defined as the average number of consecutive packet losses in the WWAN-WLAN link, can be found from (3.15) as

$$L_b = \sum_{i=1}^{\infty} i \cdot r_{FF}^{i-1}(1 - r_{FF}) = \frac{1}{1 - r_{FF}}. \tag{3.16}$$

The packet loss probability (P_l), defined as the probability that a packet is lost over the WWAN-WLAN link, can be computed from (3.15) as

$$P_l = \frac{1 - r_{SS}}{2 - r_{FF} - r_{SS}}. \tag{3.17}$$

In order to compute L_b and P_l, we need to find expressions for transition probabilities r_{SS} and r_{FF}. From the WWAN link model (Subsection 3.2.1) and WLAN link model (Subsection 3.2.2), the packet loss probability over the WWAN-WLAN link, can also be computed as

$$P_l = \pi_F^{WWAN} + (1 - \pi_F^{WWAN}) \cdot \pi_F^{WLAN}. \tag{3.18}$$

By combining (3.17) and (3.18), r_{SS} can be expressed in terms of r_{FF} and other known variables. Therefore, we only need to find the expression for r_{FF} in order to define all transition probabilities in matrix **R,** that characterizes the two-state Markov chain.

Let X and Y denote two consecutive packet transmission events over the WWAN-WLAN link. By definition, r_{FF} means $\Pr(Y = F \mid X = F)$ and it can be written as

$$\Pr(Y = F \mid X = F) = \frac{\Pr(X = F, Y = F)}{\Pr(X = F)} \tag{3.19}$$

where $\Pr(Y = F, Y = F)$ is the joint probability that two consecutive packets are lost. $\Pr(X = F)$ is simply given by $\pi_F^{WWAN} + (1 - \pi_F^{WWAN}) \cdot \pi_F^{WLAN}$. To calculate $\Pr(X = F, Y = F)$, let A and B be the events that a packet is lost over the WWAN and WLAN links, respectively. Then, $\Pr(X = F, Y = F)$ is given by

$$\begin{aligned} \Pr(X = F, Y = F) = \Pr(X = A, Y = A) + \Pr(X = A, Y = B) \\ + \ \Pr(X = B, Y = A) + \Pr(X = B, Y = B). \end{aligned} \tag{3.20}$$

The remainder of this subsection presents the expressions for $\Pr(X = A, Y = A)$, $\Pr(X = A, Y = B)$, $\Pr(X = B, Y = A)$, and $\Pr(X = B, Y = B)$.

3.2.3.1 Expression for $\Pr(X = A,\ Y = A)$

The event ($X = A$, $Y = A$) represents that the first packet is lost over the WWAN link and the second packet is also lost over the WWAN link. The probability of the first packet loss is π_F^{WWAN}. Since the first packet is discarded at the WWAN link, the WWAN link condition during the last transmission of the first packet is bad. Therefore, the probability that the first transmission of the second packet fails is given by p_{bb}. Also, if all $l-1$ retransmissions fail, the second packet is discarded. Consequently, $\Pr(X = A,\ Y = A)$ is given by

$$\Pr(X = A, Y = A) = \pi_F^{WWAN} \cdot p_{bb} p_{bb}^{l-1}. \tag{3.21}$$

3.2.3.2 Expression for Pr($X = A$, $Y = B$)

The event ($X = A$, $Y = B$) refers to the case where the first packet is lost over the WWAN link and the second packet is lost over the WLAN link. Since the first packet is lost at the WWAN link, the WWAN link state at the *l*th transmission of the first packet is bad. To successfully transmit the second packet over the WWAN link, the WWAN link should be in the good state. Let ϕ_b be the probability that the second packet transmission over the WWAN link is successful given that the link state at the last trial of the first packet is bad. Then, ϕ_b is derived as

$$\phi_b = p_{bg} + p_{bb}p_{bg} + p_{bb}^2 p_{bg} + \cdots + p_{bb}^{l-1} p_{bg} = p_{bg} \frac{1 - p_{bb}^l}{1 - p_{bb}} = 1 - p_{bb}^l, \tag{3.22}$$

where $p_{bb}^{i-1} p_{bg}$ represents the probability that the WWAN link state becomes good at the *i*th attempt ($1 \le i \le l$). On the other hand, the probability that the second packet is lost over the WLAN link is π_F^{WLAN}. Consequently, Pr($X = A$, $Y = B$) is derived as

$$\Pr(X = A, Y = B) = \pi_F^{WWAN} \phi_b \pi_F^{WLAN}. \tag{3.23}$$

3.2.3.3 Expression for Pr($X = B$, $Y = A$)

In the event (X = B, Y = A), the first packet is lost over the WLAN link and its probability is $(1 - \pi_F^{WWAN})\pi_F^{WLAN}$. In this case, the WWAN link state at the last attempt of the first packet is good. However, the second packet is dropped at the WWAN link due to bad link condition. The probability for this event is $p_{gb} p_{bb}^{l-1}$. Consequently, Pr(X = B, Y = A) is given by

$$\Pr(X = B, Y = A) = (1 - \pi_F^{WWAN})\pi_F^{WLAN} p_{gb} p_{bb}^{l-1}. \tag{3.24}$$

3.2.3.4 Expression for Pr(X = B, Y = B)

For the event (X = B, Y = B), two consecutive packets are lost over the WLAN link. This implies that two consecutive packets are successfully transmitted over the WWAN link. Let ϕ_g be the probability that the second packet transmission over the WWAN link is successful given that the WWAN link state on the last transmission attempt of the first packet is good. Also let θ_F be the probability that the second packet is lost over the WLAN link when the first packet is lost over the WLAN link. Then, Pr(X = B, Y = B) is given by

$$\Pr(X = B, Y = B) = (1 - \pi_F^{WWAN})\pi_F^{WLAN}\phi_g\theta_F. \tag{3.25}$$

Similar to ϕ_b in (3.22), ϕ_g is obtained from

$$\phi_g = p_{gg} + p_{gb}p_{bg} + p_{gb}p_{bb}p_{bg} + p_{gb}p_{bb}^2p_{bg} + \cdots + p_{gb}p_{bb}^{l-2}p_{bg} = 1 - p_{gb}p_{bb}^{l-1}, \tag{3.26}$$

where p_{gg} is the probability that the WWAN link condition is good at the first transmission attempt and $p_{gb}p_{bb}^{i-2}p_{bg}$ is the probability of the event when the WWAN link becomes good at the ith transmission attempt ($2 \le i \le l$).

In order to derive the expression θ_F, let us consider the WLAN model developed in Subsection 3.2.2, where a transmission failure over the WLAN link is caused by a collision or transmission error. Let $T_i \mid s_0 = x$ be a trace with $s_0 = x \in \{b, g\}$, where s_j is the WLAN link state at the jth transmission attempt. Similarly, $T_i \mid s_m = x$ represents a trace with $s_m = x \in \{b, g\}$. Then, θ_F is represented by

$$\begin{aligned}\theta_F = & \frac{\omega_b}{\omega_b+\omega_g}\Big(\sum_i \Theta_b(T_i \mid s_0 = b)p^{N(T_i|s_0=b)} + \sum_i \Theta_b(T_i \mid s_0 = g)p^{N(T_i|s_0=g)}\Big) \\ & + \frac{\omega_g}{\omega_b+\omega_g}\Big(\sum_i \Theta_g(T_i \mid s_0 = b)p^{N(T_i|s_0=b)} + \sum_i \Theta_g(T_i \mid s_0 = g)p^{N(T_i|s_0=g)}\Big)\end{aligned} \tag{3.27}$$

where $\omega_b = \sum_i \Theta(T_i \mid s_m = b)p^{N(T_i|s_m=b)}$ and $\omega_g = \sum_i \Theta(T_i \mid s_m = g)p^{N(T_i|s_m=g)}$. That is, ω_b and ω_g are the probabilities that the last transmission failure of the first packet in the WLAN link is due to transmission error and collision, respectively. In (3.27), the first term refers to the case that the second packet is lost over the WLAN link when the last transmission of the first packet fails due to bad link condition. The second term represents that the second packet is lost over the WLAN link when the first packet fails at the last transmission attempt due to collision. $\Theta_b(\cdot)$ and $\Theta_g(\cdot)$ are the occurrence probabilities when the WLAN link states at the last transmission of the previous packet (i.e., τ_0 in Figure 3.1) are known as bad and good, respectively. For a given $T_i = (s_0, s_1, ..., s_m)$, they are defined as $\Theta_b(T_i) = q_{bs_0}^{(0)}q_{s_0s_1}^{(1)}...q_{s_{m-1}s_m}^{(m)}$ and $\Theta_g(T_i) = q_{gs_0}^{(0)}q_{s_0s_1}^{(1)}...q_{s_{m-1}s_m}^{(m)}$.

3.3 Model Verification and Analysis

We verify the correctness of the analytical model by conducting extensive simulations using the ns-2 simulator [69] and analyze the influence of velocity and load of the mobile WLAN, WLAN and WWAN retransmit retry limits, and WLAN and WWAN channel conditions on WLAN-WWAN link packet loss probability and packet loss burst length. For realistic simulations, we use continuous-time two-state error models for the WWAN and WLAN links. That is, the durations of the good and bad states are drawn from exponential distributions where the average durations correspond to those in the discrete-time two-state error model used in the analytical model.

The following parameters are used for our simulation and analytical results, unless otherwise explicitly stated. The carrier frequency of the WWAN link is 900 MHz. The WWAN bandwidth is assumed to be 400 Kbps and the payload size is fixed to 250 bytes. Hence, the transmission time of a packet over the WWAN link is 5 msec. The default values (i.e., the values used if they are not explicitly stated) for velocity is 10 m/s and fading margin is 10 dB. Regarding the WLAN link, the transmission error rate and burst error length are 0.01 and 1000 slots, respectively. The default values of retransmission limits for the WWAN and WLAN links are 4. The number of MNs within a vehicle is varied from 1 to 20 and its default value is 10. The parameters for WLAN follow those of the IEEE 802.11b specification in [8] and the data rate is 11 Mbps. To obtain more accurate results, the simulation is conducted for 100 packets and repeated 100 times with different random seeds.

From Figures 3.2-3.8, it can be seen that despite of approximations considered in Section 3.2, the analytical results are in close agreement with simulation results. This verifies the reasonableness of our assumptions and correctness of the analysis.

3.3.1 Mobile WLAN Velocity and Load

Figure 3.2 shows the effect of velocity (v) of the mobile WLAN and the numbers of MNs in WLAN (N) on packet loss probability (P_l) of the WWAN-WLAN link. It can be seen that P_l decreases as v increases. This observation can be explained as follows. When v increases, the Doppler frequency increases (i.e., the WWAN link's coherence time decreases) and that reduces the burstiness of the transmission errors in the WWAN link. Since there is a finite number of retransmission attempts in the WWAN link, P_l decreases as v increases and then P_l remains fairly constant when v exceeds a certain value. Figure 3.2 also indicates that P_l increases as N (load of the mobile WLAN) increases. This is due to the fact that transmission errors in the WLAN link are fairly steady; therefore, the number of competing nodes is an important factor affecting the packet loss probability. In other words, as the number of nodes serviced by a single AP in a mobile WLAN increases, the packet losses due to frequent collisions also increase.

Note that even though analytical and simulation results in Figure 3.2 are mostly consistent, there is some discrepancy between them when N is small and v is low. This discrepancy is partially due to the fact that the channel coherence time increases as v decreases. Therefore, the independent transmission errors assumption made for p_f and p becomes less valid for lower values of v.

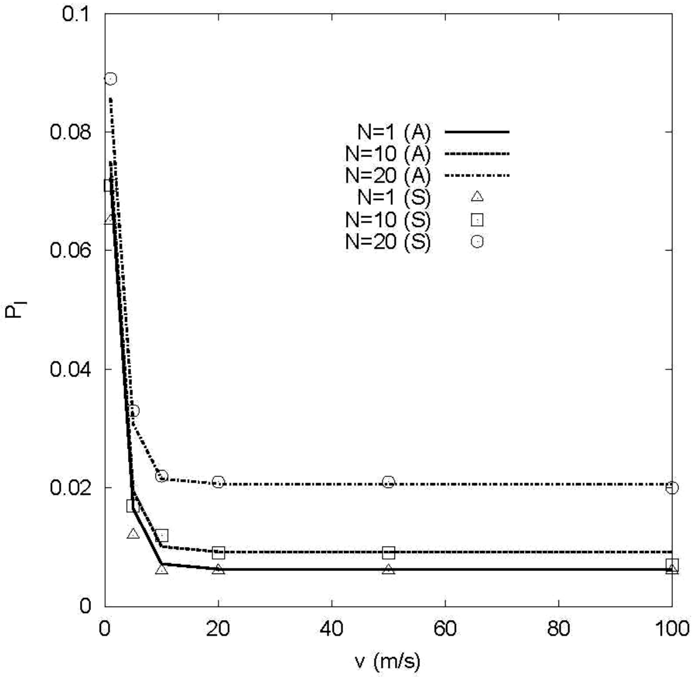


Figure 3.2. Effects of *v* and *N* on P_l (A: Analysis, S: Simulation)

Figure 3.3 plots the packet loss burst length (L_b) over the WWAN-WLAN link as a function of v. It can be seen that when v < 5 m/s, L_b is drastically reduced as v increases. However, L_b increases with v when 5 m/s < v < 20 m/s and L_b remains fairly constant for v > 20m/s. This interesting observation can be explained as follows. From (3.16) and (3.19), L_b can be written as a function of π_F and Pr(X = F, Y = F), i.e., $L_b = \pi_F / (\pi_F - \Pr(X = F, Y = F))$. Since π_F is more sensitive to the channel coherence time than Pr(X = F, Y = F), π_F is more drastically reduced than $\pi_F - \Pr(X = F, Y = F)$ with v when v < 5 m/s. Accordingly, L_b decreases as v increases. On the other hand, when v exceeds 5 m/s, Pr(X = F, Y = F) becomes almost constant. Therefore, L_b can be represented in the form $f(x) = x/(x-a) \ = 1 + a/(x-a)$, where $f(x)$, x and a correspond to L_b, π_F, and Pr(X = F, Y = F), respectively, and a is a constant. As mentioned before, π_F is reduced as v increases. Consequently, L_b increases as v increases because $f(x)$ is a decreasing function of $x > a$. However, if v exceeds 20m/s, the reduction of π_F is also negligible. Therefore, L_b does not further increase.

From Figure 3.3, it is also shown that L_b is reduced as N increases. Since Pr(X = F, Y = F) is mainly affected by the fading process in the WWAN and WLAN links, it is rarely affected by N. Therefore, Pr(X = F, Y = F) can be considered as a constant with respect to N and thus L_b can be represented as a decreasing function of π_F, similar to $f(x)$. Since π_F increases with N, L_b is reduced as N increases.

3.3.2 Retransmission Retry Limits

The link layer retransmission mechanisms are essentially employed in order to improve transmission reliability. Let $l-1$ and m be retransmission retry limit set in the WWAN and WLAN links, respectively. The effects of l and m on P_l are respectively shown in Figure 3.4 and Figure 3.5. In Figure 3.4, it can be seen that P_l is reduced when a large l is employed. However, the effect of l becomes insignificant at a high value of v. As v increases, the coherence time in the WWAN link apparently decreases. Therefore, burst transmission errors in the WWAN link are infrequent when v is high, and thus the reduction of P_l by employing a large l is not noticeable. Similarly, P_l can be reduced by increasing m as shown in Figure 3.5. In the WLAN link, more packets are lost due to collisions when N is large. Therefore, the effectiveness of a large m on P_l becomes apparent as N increases.

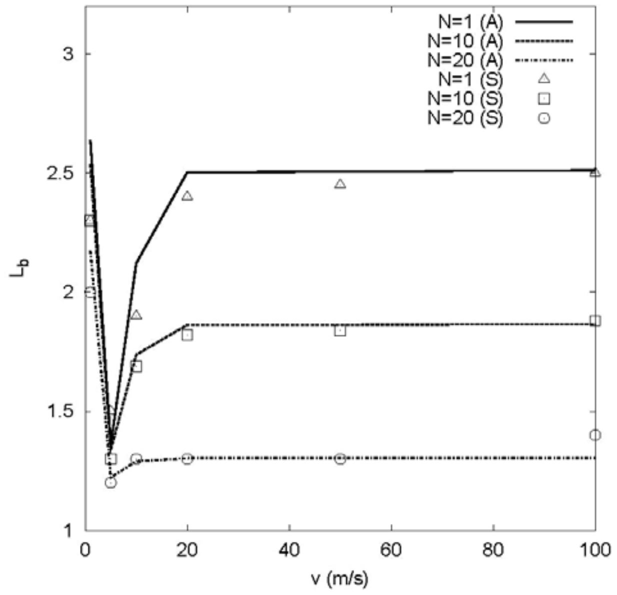


Figure 3.3. Effects of *v* and *N* on L_b (A: Analysis, S: Simulation)

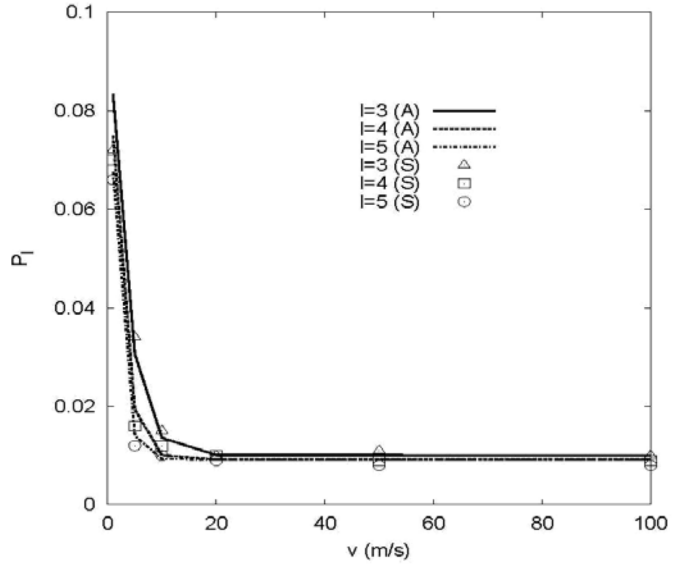


Figure 3.4. Effect of *l* on P_l (A: Analysis, S: Simulation)

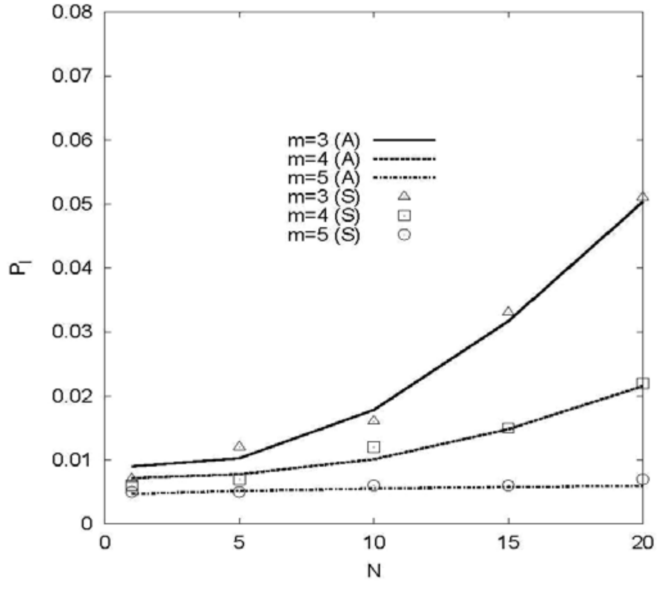


Figure 3.5. Effect of *m* on P_l (A: Analysis, S: Simulation)

3.3.3 Wireless Link Conditions

The fading margin Γ is the maximum fading attenuation that the system can tolerate. Therefore, the WWAN link condition (transmission error probability in WWAN) can be varied by changing the fading margin Γ. The WLAN link condition (transmission error probability in WLAN) can be varied by changing the average error rate π_b^{WLAN}.

Figure 3.6 shows the effect of the WWAN link conditions on P_l. As we expected, P_l decreases as Γ increases. However, this trend is only visible when v is sufficiently low. This is due to the fact that when v is high, the burstiness of WWAN transmission errors is negligible. Since most non-bursty WWAN transmission errors can be recovered using ARQ, the effect of the WWAN transmission error probability is not significant for a high value of v. Figure 3.7 indicates that the effects of π_b^{WLAN} on P_l becomes clearer when l_B is high. The approximate increasing rates in P_l are 0.01 and 0.68 when l_B is 100 and 1000, respectively. Figure 3.8 demonstrates the effects of π_b^{WLAN} and l_B on the packet loss burst length (L_b). When l_B is 100, the packet loss burstiness is negligible, i.e., L_b is almost 1.0. However, when l_B is 500 or 1000, an apparent L_b larger than 1.0 is observed.

3.4 Summary

In this chapter, we have proposed channel models based on two-state Markov chains in order to facilitate the performance analysis of the transport mechanisms in integrated WLAN and WWAN. First, we have developed a novel analytical model which describes packet transmission over a fading channel for the IEEE 802.11-based WLAN link. Then, based on the developed WLAN model and existing WWAN model, we have constructed a WLAN-WWAN model and derived the expressions for packet loss probability and packet loss burst length. Finally, we have verified the correctness of the developed model and analyzed its behavior at the packet level.

The main contributions in this chapter are three-fold: 1) we have constructed a novel analytical model for WLAN link. Unlike the previous WLAN models, our analytical model captures the impact of both uncorrelated and correlated transmission errors; 2) we have developed analytical model which can describe the packet loss probability and packet loss burst length of the WWAN-WLAN link. This model is suitable for describing the packet level characteristics of integrated mobile WLAN and WWAN; and 3) we have conducted extensive analysis, which sheds some lights to designing heterogeneous wireless networks such as integrated mobile WLAN and WWAN.

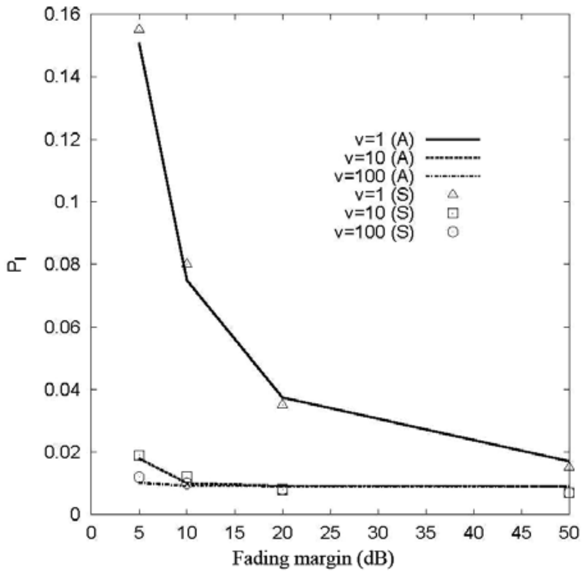


Figure 3.6. Effect of WWAN link condition on P_l (A: Analysis, S: Simulation)

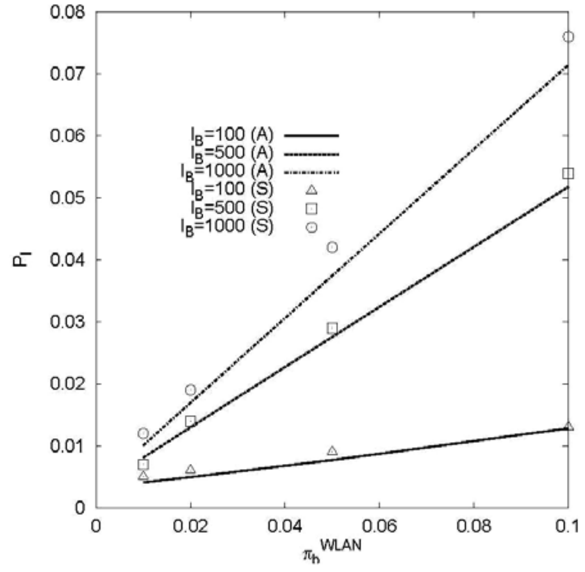


Figure 3.7. Effect of WLAN link condition on *P_l* (A: Analysis, S: Simulation)

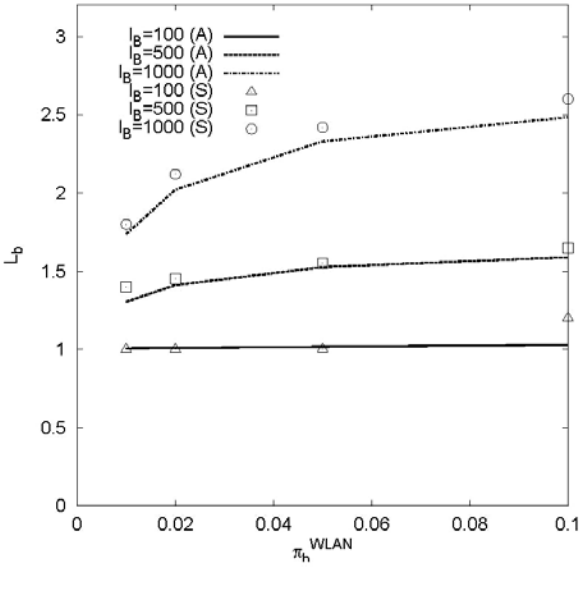


Figure 3.8. Effect of WLAN link condition on *L_b* (A: Analysis, S: Simulation)

Chapter 4

Analytical Framework for Short-lived WP-TCP Flows

4.1 Introduction

The connection life time of WP-TCP short-lived flows is considerably short compared to the network residence times. Consequently, the effects of vertical handover on the performance of WP-TCP short-lived over integrated static WLAN and WWAN are negligible. Therefore, the performance of WP-TCP short-lived flows in the network scenarios of our interest (summarized in Section 1.2) can be analyzed by using generic packet level channel models that can capture the effects of transmission errors in WWAN, WLAN, and WWAN-WLAN links.

A limited amount of analytical frameworks for studying the performance of WP-TCP short-lived flows have been reported in the literature. Since WP-TCP is one of the TCP variants, previous analytical framework for standard TCP short-lived flows may be used. There are several analytical models proposed for short-lived TCP transfers [21], [22], [23], [24]. Unfortunately, these models are based on an independent packet loss assumption which does not fit well in the wireless environment. In order to study the performance of WP-TCP short-lived flows with reasonable accuracy, the packet level channel models, which capture both correlated and uncorrelated transmission errors, need to be developed.

In Chapter 3, it has been shown that two-state Markov channel model can capture both correlated and independent packet losses that can occur in WWAN, WLAN, and WWAN-WLAN links. In this chapter, we develop an analytical framework for WP-TCP short-lived flows based on the two-state Markov channel model and study the performance. We first derive an explicit mathematical expression that estimate the WP-TCP transfer delay and then analyze the transfer delay performance. Simulation results are also provided to verify the accuracy of analytical results.

The remainder of this chapter is organized as follows. The analytical framework for studying the performance of WP-TCP short-lived flows is developed in Section 4.2. Section 4.3 discusses the simulation and analytical results, followed by summary of this chapter in Section 4.4.

4.2 Analytical Framework

In this section, we develop an analytical framework for studying the performance of WP-TCP short-lived flows where a single pair WP-TCP sender-receiver is assumed to run over the wireless channel whose average error rate (e) and packet burst length (BL) is approximated by the two-state Markov channel model. Note that e is a ratio of the number of packets corrupted to the number of packets transmitted over wireless channel and BL is average length of consecutive packet losses (measured in packets) over wireless channel. WP-TCP is considered to be implemented with all mandatory requirements and some important optional requirements (Large initial window, selective acknowledgement (SACK), timestamps option for round-trip time measurement). Our performance metric is file transfer time (transfer delay), which is defined as the time used to download a given file size.

From Section 2.3.1, WP-TCP has three transfer stages: connection establishment, data transfer, and connection tearing down. Since we are only interested in the completion time of transferring the actual data packets, the connection tearing down stage is not considered. In case there will be a need to consider this stage, the modeling process of the connection tearing down stage is similar to that for the connection establishment stage. Therefore, the average file transfer time (T) can be found as: $T = T_{\mathrm{cs}} + T_{\mathrm{dt}}$, where T_{cs} is the average connection setup time and T_{dt} is the average data transfer time. For easy reference, the notions used frequently in this chapter are listed in Table 4.1.

4.2.1 Connection Establishment

The connection establishment stage is a "three-way handshake" as described in Section 2.3.1. In this stage, the packet losses due to wireless can occur in both the downlink direction and the uplink direction. Since the timeout for SYN packet is relatively large and doubles after every transmission retry, the correlation of SYN packet losses or SYN-ACK packet losses is very weak. Therefore, uncorrelated packet losses are considered in connection establishment phase. By assuming the same SYN timeout (T_{s}) and average error rate ($e < 0.5$) in the downlink and uplink directions, and considering an infinite number of retries for connection establishment, the average connection setup time can be calculated as [21]

$$T_{\mathrm{cs}} = 2l + T_{\mathrm{s}}\left(\frac{2e}{1\text{-}2e}\right) \tag{4.1}$$

where l is the channel delay defined as the time taken for a packet traveling from the WP-TCP sender to the WP-TCP receiver or vice-versa. The channel delay is a result of propagation delay and all processing delays encountered at the end nodes. Note that the time to transmit SYN or SYN-ACK is assumed negligible and thus the round-trip time is approximated by $2l$. This assumption is reasonable since the size of SYN and SYN-ACK packets are very small compared to the size of data packets.

Table 4.1. List of local notions for Chapter 4

Notation	Definition (Unit)
BL	burst length: average length of consecutive packet losses over wireless channel (packets)
e	average error rate: ratio of the number of packets corrupted and number of packets transmitted over wireless channel
l	channel delay: the time taken for a packet traveling from the WP-TCP sender to the WP-TCP receiver or vice-versa (sec)
W_{iw}	WP-TCP initial window (packets)
$W_{\max}$	WP-TCP maximum window size (packets)
w	WP-TCP congestion window (packets)
w_{th}	WP-TCP slow start threshold (packets)
Z	WP-TCP duplicate acknowledgement threshold (packets)
ρ	packet length (bytes)
F	file size (bytes)
$N = \lceil F/\rho \rceil$	file size (packets)
$\mathbf{\Phi} = [\phi_{AB}]$	transition matrix of the embedded Markov chain, where an element ϕ_{AB} denotes transition probability from state A to state B
$\mathbf{D} = [d_{AB}]$	matrix corresponding to the embedded Markov chain, where an element d_{AB} transition time from state A to state B in seconds

4.2.2 Data Transfer

The data transfer stage begins right after the connection establishment stage and ends when the sender receives an acknowledgement for the last byte of the transferred data. Figure 4.1 shows the approximate model for the actual WP-TCP data transfer stage. The sender begins transmission in the slow start phase until either data transfer is completed (transition a), or the channel changes to the bad state (transition b), or the congestion window reaches the slow start threshold (transition c). Similarly, the sender remains in the congestion avoidance phase until either data transfer is completed (transition d), or the channel changes to the bad state (transition e). In the loss recovery phase, the WP-TCP sender uses duplicate acknowledgement (ACK) or timeout mechanisms to detect and recover lost packets. If timeout is used to detect packet loss, the sender enters the slow start phase (transition f) and restarts retransmission. If the duplicate ACK mechanism is used to detect and recover lost packet(s), the sender enters the congestion avoidance state (transition g), and continues with transmission or finishes if the recovered packet completes the transmission (transition h).

To track the data transfer process, we consider a random process $\mathrm{S}(t) = (c(t\text{-}1), w(\mathrm{t}), w_{\mathrm{th}}(\mathrm{t}), k(t))$, where $c(t) \in \{\mathrm{b,g}\}$ is the channel state, $w(t) \in \{1,2,\ldots,W_{\max}\}$ is the congestion window, $w_{\mathrm{th}}(t) \in \{2,3,\ldots,\lceil W_{\max}/2 \rceil, W_{\max}\}$ is the slow start threshold, $k(t) \in \{0,1,\ldots,N\}$ is the number of packets transmitted successfully, t is the time measured in slots[2], $W_{\max}$ is the maximum window size, and N is the file size measured in packets. Given a packet length ρ in bytes and the size of the transfer file F in bytes, N can be computed as $N = \lceil F/\rho \rceil$. Note that according to WP-TCP

[2] A slot is equal to the time used to transmit one WP-TCP data packet

congestion control the $w_{\mathrm{th}}(t)$ is initialized to $W_{\max}$ on startup and can be reset to max$\{2, \lceil W_{\max}/2 \rceil\}$ whenever required. Since $w(t)$ cannot exceed $W_{\max}$, the possible values for $w_{\mathrm{th}}(t)$ are 2, 3,…, $\lceil W_{\max}/2 \rceil$, and $W_{\max}$. By sampling the random process $S(t)$ at instants t_n just after transition shown in Figure 4.1 occurs, the new sampled process $S(t_n)$ is a semi-Markov process [70] with embedded Markov chain, in the state space $W_s = \{(c,w,w_{\mathrm{th}},k)$: $c \in \{\{b,g\}$, $w \in \{1,2,\ldots,W_{\max}\}$, $w_{\mathrm{th}} \in \{2,3,\ldots, \lceil W_{\max}/2 \rceil, W_{\max}\}$, $k \in \{0,1,\ldots,N\}\}$, defined by the transition probability matrix $\mathbf{\Phi} = [\phi_{AB}]$. ϕ_{AB} denotes the transition probability from state $A \in W_s$ to state $B \in W_s$.

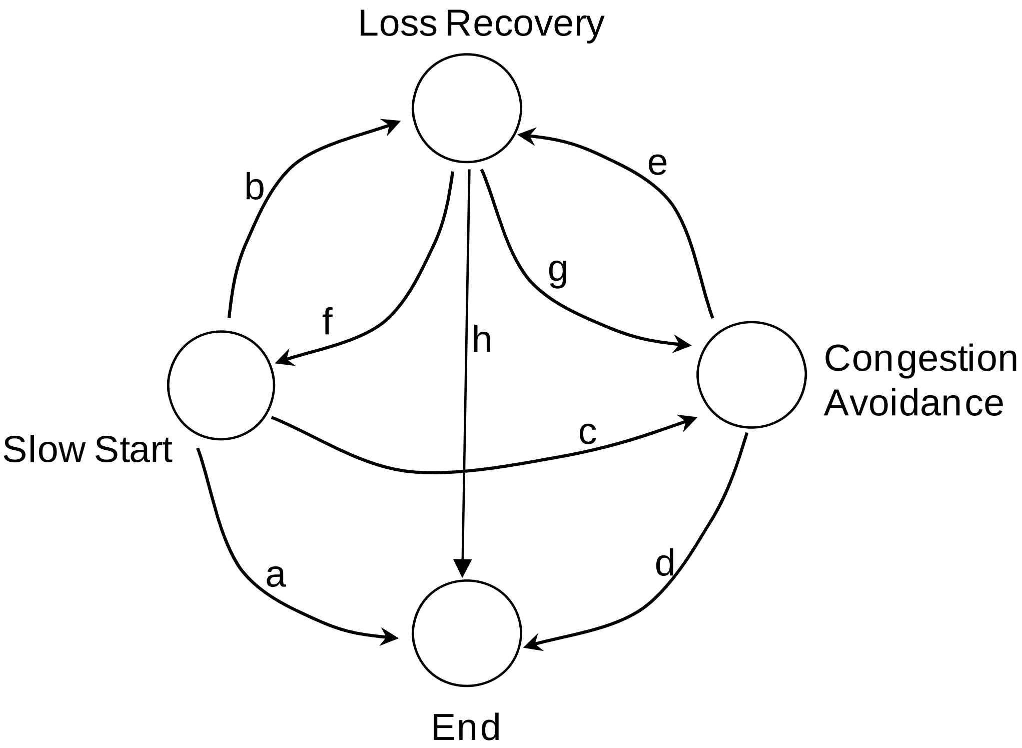


Figure 4.1. Approximate model for data transfer stage

To compute the average data transfer time, the transitions of embedded Markov chain $S(t_n)$ are labeled with corresponding elements of the matrix $\mathbf{D} = [d_{AB}]$, where element d_{AB} is the transition time from state $A \in W_s$ to state $B \in W_s$. To reduce the state space and computational complexity, all possible states that the data transfer stage can end with are grouped to form a super state denoted as (X,X,X,N). Since all states forming the super state (X,X,X,N) are trapping states (i.e., they have no self-transitions), the vector $\mathbf{T} = [T_A]$ whose elements represent the expected delay given that the transmission process starts in the initial state $A \in W_s$ and terminates at the super state (X,X,X,N) can be computed as [70]

$$\mathbf{T} = [\mathbf{I} - \mathbf{\Phi}'']^{-1} \overline{\mathbf{D}}, \tag{4.2}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{\Phi}''$ is obtained by deleting a row and a column corresponding to the super state (X,X,X,N) from $\mathbf{\Phi}$, and $\overline{\mathbf{D}} = diag(\mathbf{\Phi}'[\mathbf{D}']^t)$, where the superscript t denotes matrix transpose. $\mathbf{\Phi}'$ and $\mathbf{D}'$ are obtained by only deleting a row corresponding to the super state (X,X,X,N) from $\mathbf{\Phi}$ and $\mathbf{D}$, respectively. By considering the WP-TCP initialization procedure introduced in Section 2.3.2, the possible data transfer initial states are $(\mathrm{g},W_{\mathrm{iw}},W_{\mathrm{max}},0)$ and $(\mathrm{b},W_{\mathrm{iw}},W_{\mathrm{max}},0)$. Consequently, the average data transfer time can be computed as

$$T_{\mathrm{dt}} = \pi_g T_{(g,W_{iw},W_{\max},0)} + \pi_b T_{(b,W_{iw},W_{\max},0)}, \tag{4.3}$$

where π_{g} and π_{b} are the steady state probabilities of the channel being in state g and b, respectively. π_{g} and π_{b} can be computed from the transition probability matrix $\mathbf{P}$ as shown in [71]. To compute the elements of the matrices $\mathbf{\Phi}$ and $\mathbf{D}$, the transmission process in the data transfer stage is analyzed, as depicted in Figure 4.1, in three phases: slow start, congestion avoidance, and loss recovery. In each phase, the initial states and possible transitions to the final states are explored by conditioning the number of packets transmitted successfully. Then, for every possible transition the corresponding transition probability and transition time are computed. Note that this model is not expected to produce exact value for data transfer time but rather a reasonable approximation.

4.2.2.1 Slow Start

The transmission process enters the slow start phase either at the beginning of data transfer or when a packet loss is recovered by using the timeout mechanism. Therefore, a set of initial states is A = {(g,W_{iw},$W_{\max}$,0) $\cup$ [(g,1,w_{th},k): $w_{\mathrm{th}} \in \{2,3,\ldots,\lceil W_{\max}/2 \rceil\}$, $k \in \{1,2,\ldots,N\text{-}1\}$]}, where $\cup$ is the union operation. Let $s \in \{1,2,\ldots, N\text{-}k\}$ be the number of packets transmitted successfully before a transition occurs (transfer ends or enters congestion avoidance or packet loss). Under perfect channel condition, the latency and dynamic of congestion window in the slow start phase can accurately be modeled [62]. Given the initial state (g,w,w_{th},k), the possible final states can be deterministically computed as

$$B = \begin{cases} (X, X, X, N), & s \le 2w_{th} - w, s = N - k \\ (b, W_B^{SS}(s), w_{th}, k + s), & s < 2w_{th} - w, s < N - k \\ (g, W_B^{SS}(s), w_{th}, k + s), & s = 2w_{th} - w, s < N - k, \end{cases} \tag{4.4}$$

where $W_B^{\mathrm{SS}}(s) = \lfloor (w+s)/2 \rfloor$ is the congestion window after sending s packets. Note that the first case, second case, and third case in equation (4.4) correspond to transitions a, b, and c presented in Figure 4.1, respectively. The associated transition time can be obtained by

$$d_{AB} = \begin{cases} s, & Q < 1 \\ \\ s + (2l+1)Q - (2^{Q} - 1)w, & \text{otherwise}, \end{cases} \tag{4.5}$$

where $Q = \lfloor \min\{\ 1 + \log_2(2l/w + 1/w), \log_2(1 + s/w - 1/w)\} \rfloor$ and l is the channel delay measured in slots. The transition probability can be found as

$$\phi_{AB} = \begin{cases} p_{\mathrm{gg}}^{s-1}, & \mathrm{s} = \min\{N-k, 2w_{\mathrm{th}} - w\} \\ p_{\mathrm{gg}}^{s-1} p_{\mathrm{gb}}, & s < \min\{N-k, 2w_{\mathrm{th}} - w\} \\ 0, & \text{otherwise.} \end{cases} \tag{4.6}$$

4.2.2.2 Congestion Avoidance

Since the transmission process enters the congestion avoidance phase when a packet loss is recovered by using the duplicate ACK mechanism or when the congestion window reaches the slow start threshold, a set of initial states is therefore given as A={(g,w_{th},w_{th},k): $w_{\mathrm{th}} \in$ {2,3,…, $\lceil W_{\max}/2 \rceil$, $W_{\max}$}, $k \in$ {1,2,…,N-1}}. Let $s \in$ {1,2,…, N-k} be the number of packets transmitted successfully before transition occurs (transfer ends or packet loss occurs). Under perfect channel condition, the latency and dynamic of congestion window in the congestion avoidance phase can be modeled similar to that in [62]. Given the initial state (g,w_{th},w_{th},k), the possible final states can be deterministically obtained as

$$B = \begin{cases} (X, X, X, N)\ , & s = N - k \\ (\mathrm{b}, W_{A}^{\mathrm{CA}}(s), w_{\mathrm{th}}, k+s)\ , & s < N - k, \end{cases} \tag{4.7}$$

where $W_B^{\mathrm{CA}}(s) = \min\{-1/2 + \sqrt{(w_{\mathrm{th}} - 1/2)^2 + 2s}, W_{\max}\}$. Note that the first case and second case in equation (4.7) correspond to transitions d and e presented in Figure 4.1, respectively. The associated transition time can be found as

$$d_{AB} = s + (2l + 2 - w_{\mathrm{th}})Q - (Q+1)Q/2 \tag{4.8}$$

where $Q = \min\{\ 2l - w_{\mathrm{th}} + 2, 1/2 - w_{\mathrm{th}} + \sqrt{(w_{\mathrm{th}} - 1/2)^2 + 2s}\ \}$. The transition probability can be written as

$$\phi_{AB} = \begin{cases} p_{\mathrm{gg}}^{s-1}, & s = N - k \\ p_{\mathrm{gg}}^{s-1} p_{\mathrm{gb}}, & s < N - k \\ 0, & \text{otherwise.} \end{cases} \tag{4.9}$$

4.2.2.3 Loss Recovery

All transitions to the loss recovery phase occur when the channel becomes bad during transmission and the first packet loss occurs in the first time slot of the loss recovery phase. Therefore, a set of initial states is given as A = {(b, w, w_{th}, k): $w \in \{1,2,\ldots,W_{\max}\}$, $w_{\mathrm{th}} \in \{2,3,\ldots,\lceil W_{\max}/2 \rceil, W_{\max}\}$, $k \in \{0,1,\ldots, N\text{-}1\}$}. After the first packet loss, all subsequent packets that are transmitted successfully will generate duplicate ACKs. Let Z be the duplicate ACK threshold. If the total number of duplicate ACKs is less than Z, the sender will wait for a timeout interval (T_0) and then enter the slow start phase. When the number of duplicate ACKs reaches Z, fast transmit followed by fast recovery will be triggered. During fast recovery, duplicate ACKs are ignored until half of the window is acknowledged, and after that, the WP-TCP sender will send a new packet for every received duplicate ACK. To model the recovery process, two cases are considered: when a lost packet is recovered with the timeout mechanism and when it is recovered without using the timeout mechanism.

Let $\alpha_c(s,w)$ be defined as the probability of having $s \in \{1,2,\ldots, w\text{-}1\}$ successfully transmitted packets out of w transmitted packets given that the channel was in the bad state at the beginning of transmission and the channel is in state $c \in \{\mathrm{g},\mathrm{b}\}$ at the end of transmission. Similar to [72], it follows that

$$\alpha_{\mathrm{b}}(s,w) = \begin{cases} p_{\mathrm{bb}}^{w}, & s = 0 \\ \sum_{i=1}^{\min\{s,w-s\}} \binom{w-s}{i}\binom{s-1}{i-1} p_{\mathrm{bb}}^{w-s-i}\, p_{\mathrm{bg}}^{i}\, p_{\mathrm{gb}}^{i}\, p_{\mathrm{gg}}^{s-i}, & s = 1,\ldots,w-1 \\ 0, & s \geq w \end{cases} \tag{4.10}$$

and

$$\alpha_{\mathrm{g}}(s,w) = \begin{cases} \sum_{i=1}^{\min\{s+1,w-s\}} \binom{w-s-1}{i-1}\binom{s}{i-1} p_{\mathrm{bb}}^{w-s-i}\, p_{\mathrm{bg}}^{i}\, p_{\mathrm{gb}}^{i-1}\, p_{\mathrm{gg}}^{s-i+1}, & s = 0,\ldots,w-1 \\ 0, & s \geq w. \end{cases} \tag{4.11}$$

4.2.2.3.1 Loss recovery with timeout mechanism

We further consider two scenarios that will result in timeout. The first scenario is when s is less than Z. The second scenario is when s is greater than or equal to Z but less than half of the congestion window and packets retransmitted using fast transmit get lost. From empirical studies on TCP [73], it is found that most of the TCP flows only suffer a single timeout. With this observation, a timeout with no exponential backoff is assumed. Since WP-TCP is implemented with the timestamps option for round-trip time measurement, the WP-TCP sender can estimate round trip time fairly accurately. Consequently, in this analysis T_0 is set to twice the round trip time (i.e., $T_0 = 2(2l+1)$). Given the initial state (b, w, w_{th}, k), possible final states can be written as

$$B = (c, 1, \lceil w/2 \rceil, k+s) \tag{4.12}$$

for $c \in \{\mathrm{g, b}\}$. The transition time and transition probability can be written as

$$d_{AB} = \begin{cases} T_0, & s < Z \\ \\ T_0, & Z \leq s, s < w/2 \end{cases} \tag{4.13}$$

and

$$\phi_{AB} = \begin{cases} [\alpha_{\mathrm{g}}(s,w) + \alpha_{\mathrm{b}}(s,w)] p_{\mathrm{b}c}(T_0 - 1), & s < Z \\ [\alpha_{\mathrm{g}}(s,w) + \alpha_{\mathrm{b}}(s,w)] p_{\mathrm{bb}}(w - s + Z + 2l) p_{\mathrm{bc}}(T_0 - 1), & Z \leq s, s < \dfrac{w}{2} \\ 0, & \text{otherwise,} \end{cases} \tag{4.14}$$

where $p_{xy}(n)$ is the n-step transition probability from channel state $x \in \{\mathrm{b,g}\}$ to channel state $y \in \{\mathrm{b,g}\}$.

4.2.2.3.2 Loss recovery without timeout mechanism

When the loss recovery process is completed without using the timeout mechanism, the sender continues with transmission of new packets if there are more data to send (i.e., if $w + k < N$); otherwise transmission ends (i.e., if $w + k = N$). Given the initial state (b, w, w_{th}, k), possible final states can be written as

$$B = \begin{cases} (X, X, X, N), & w + k = N \\ \\ (c, w/2, \lceil w/2 \rceil, w + k), & w + k < N, \end{cases} \tag{4.15}$$

for $c \in \{\mathrm{g},\mathrm{b}\}$. During the loss recovery phase it is possible for retransmitted packets to get lost again. However, if the SACK option is enabled and the number of successfully transmitted packets is larger than *Z*, the impact of retransmitted packet losses becomes less significant. Since, in this case, the number of successfully transmitted packets is greater than *Z*, perfect retransmissions are assumed (i.e., lost packets are always retransmitted successfully).

To compute the transition time and transition probability, two scenarios are further considered. The first scenario is when *s* is greater than or equal to *Z* and also greater than or equal to half of the congestion window. With the perfect retransmission assumptions the number of packets recovered per round trip will equal the number of duplicate ACKs exceeding the congestion window (i.e., *s* – *w*/2). Therefore, the time spent in recovering lost packets can be approximated as $(1+2l)\lceil (w-s)/(1+(s-w/2)) \rceil$, where (*w* – *s*) is the number of lost packets and (1+2*l*) is the round trip time measured in time slots. The second scenario is when *s* is greater than or equal to *Z* but less than half of the congestion window. Again, with perfect retransmission assumptions only one packet can be recovered per round trip. Therefore the time spent in recovering lost packets can be approximated as (1+2*l*)(*w* – *s*). Let $v(s, w)$ denote the time spent to complete the loss recovery phase given that at the beginning of the recovery phase *s* packets out of *w* transmitted packets were successful. We have

$$v(s,w)=\begin{cases}(w+2l)+(1+2l)\left\lceil\dfrac{(w-s)}{1+(s-w/2)}\right\rceil, & \dfrac{w}{2}\le s\\ (w+2l)+(1+2l)(w-s), & s<\dfrac{w}{2}.\end{cases}\tag{4.16}$$

Let $\beta(s,w,k)$ denote the probability of completing the loss recovery phase without using a time-out given that *s* packets out of *w* packets were initially transmitted successfully. Then the probability can be written as

$$\beta(s,w,k)=\begin{cases}[\alpha_{\mathrm{g}}(s,w)+\alpha_{\mathrm{b}}(s,w)]p_{\mathrm{b}c}(v(s,w)), & \dfrac{w}{2}\le s, w+k<N\\ \alpha_{\mathrm{g}}(s,w)+\alpha_{\mathrm{b}}(s,w), & \dfrac{w}{2}\le s, w+k=N\\ [\alpha_{\mathrm{g}}(s,w)+\alpha_{\mathrm{b}}(s,w)]p_{\mathrm{bg}}(w-s+Z+2l)p_{\mathrm{b}c}(v(s,w)), & s<\dfrac{w}{2}, w+k<N\\ [\alpha_{\mathrm{g}}(s,w)+\alpha_{\mathrm{b}}(s,w)]p_{\mathrm{bg}}(w-s+Z+2l), & s<\dfrac{w}{2}, w+k=N\\ 0, & \text{otherwise.}\end{cases}\tag{4.17}$$

From equations (4.16) and (4.17), the transition time and transition probability can be computed by considering *s* over the interval [*Z*, *w*-1] as

$$d_{AB}=\frac{\sum_{s=Z}^{w-1}v(s,w)\beta(s,w,k)}{\sum_{s=Z}^{w-1}\beta(s,w,k)}\tag{4.18}$$

and

$$\phi_{AB}=\sum_{s=Z}^{w-1}\beta(s,w,k).\tag{4.19}$$

4.3 Numerical Results and Discussion

In this section, the proposed analytical framework is validated by simulations using the ns-2 simulator [69]. Simulation topology is shown in Figure 4.2. A single pair of WP-TCP sender-receiver is configured to run over the wireless link with the packet error process modeled by a two-state Markov chain. FTP is set as an application that transfers a file with a specified size. WP-TCP is simulated according to the real protocol as described in Section 2.3. The underlying wireless channel is simulated to capture the fundamental effects of transmission errors. The results from simulations may be slightly inflated compared to those obtained from the real wireless networks. However, the trends related to transmission errors in wireless networks are similar.

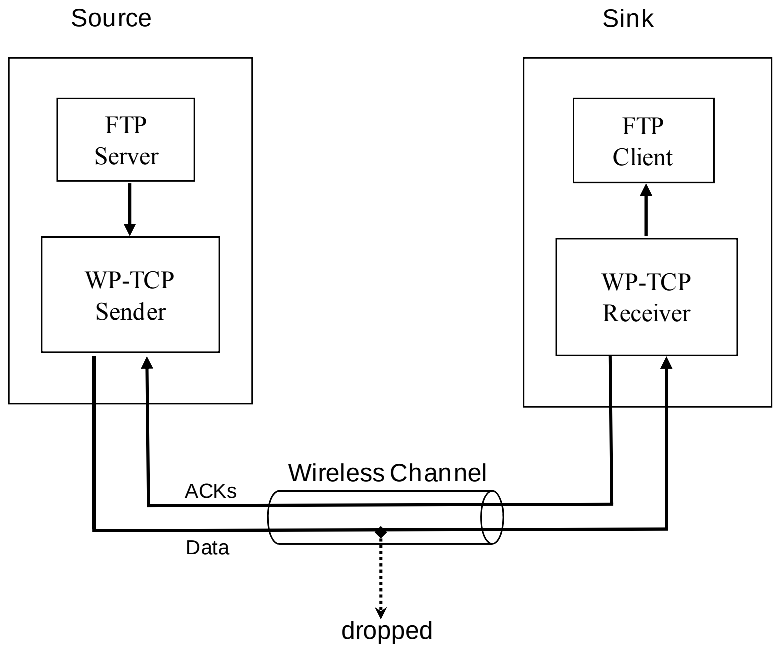


Figure 4.2. Simulation topology

To obtain more accurate results, each simulation scenario is repeated 100 times with different random seeds to arrive at the average results. The analytical and simulation results are obtained by setting the bandwidth (*B*) at 128 Kbps, the duplicate ACK threshold (*Z*) at 3, the packet length (ρ) at 1KB, while the channel delay (*l*) , the maximum window size (W_{max}), the burst length (*BL*), the file size (*F*), the average error rate (*e*), and the initial window (W_{iw}) take on different values.

From Figure 4.3-Figure 4.8 and Table 4.2-Table 4.3, it can be seen that despite of approximations introduced in Section 4.2, the analytical results are in agreement with simulation results. This verifies the reasonableness of our assumptions and correctness of the analysis.

4.3.1 Effect of the Average Error Rate and the Burst Length

BL reflects the correlation of the packet errors. If the *BL* is relatively large, packet errors are highly correlated (burst errors). On the other hand, if the *BL* is relatively small, the packet errors occur independently (random errors). In this chapter, the transmission is considered to be under a random error environment if *BL* = 1 and bursty error environment if *BL* ≥ 2.

For Wiw = 1, l = 125ms, W_{max} = 8 packets, and F = 6KB and 30KB, the impact of average error rate in the random error environment (*BL* = 1) and bursty error environment (*BL* = 3) are shown in Figure 4.3 and Figure 4.4, respectively. It can be seen that the transfer time increases with average error rate. This is because of the burden added due to packet retransmission. For small file size (6KB), the impact of average error rate in the bursty error and random error environments appears to be almost the same. However, for the large file size (30KB) the impact of error rate is more significant in the bursty error environment (Figure 4.4) than in the random error environment (Figure 4.3). For instance, when the average error rate is 0.01, the difference in transfer time when *BL* = 1 and *BL* = 3 is nearly half of a second. But when the average error rate increases to 0.15, the transfer time for *BL* = 3 is two seconds more than that for *BL* = 1. The reason for these phenomena can be explained as follows. In the case of small files, the congestion window tends to be small and therefore any packet loss will most likely cause a timeout to trigger. Hence bursty losses and random losses exert almost the same effect to transfer time. In the case of large files, congestion window tends to grow relatively large such that packet losses can be detected by duplicate acknowledgements. Since in the bursty error environment most of the corrupted packets are from the same sender window, the advantage of having relatively large congestion window becomes more beneficial in the random error environment than in the bursty error environment.

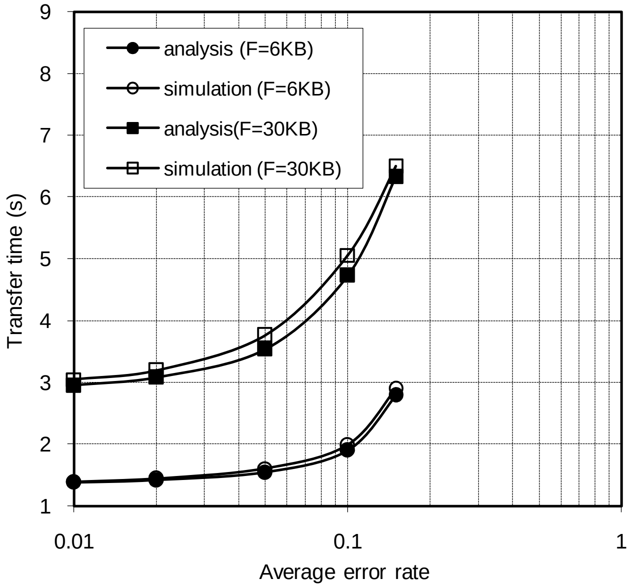


Figure 4.3. Transfer time vs. *e* for *BL* = 1 and *F* = 6KB and 30KB

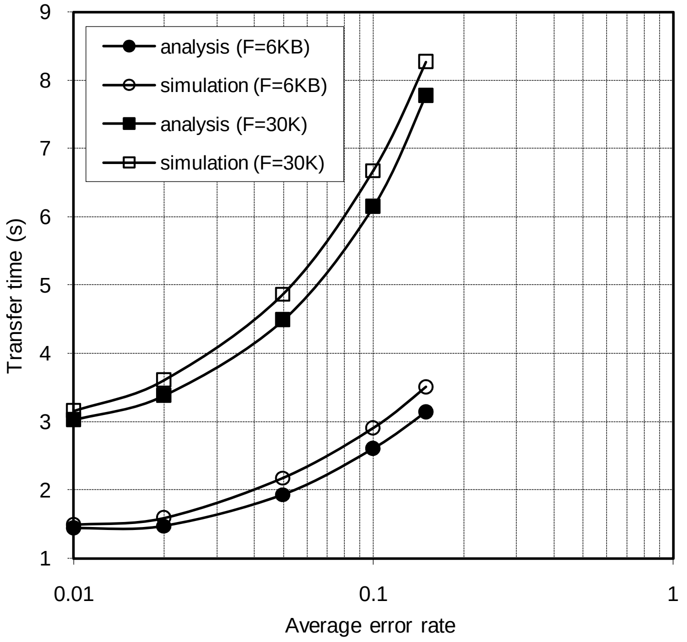


Figure 4.4. Transfer time vs. *e* for *BL* = 3 and *F* = 6KB and 30KB

4.3.2 Effect of the Initial Window

Table 4.2 and Table 4.3 present analytical and simulation transfer times for $W_{\max}$ = 8 packets, l = 125ms, BL = 3, and various values of F, e, and W_{iw}. From Table 4.2 it can be seen that, in a low error rate environment (e = 0.01), WP-TCP performs better when W_{iw} = 4 packets than when W_{iw} = 1 packet. This is because the increase of initial window reduces the number of unnecessary round trip times. From Table 4.3, it can be seen that in a high error rate environment (e = 0.10) WP-TCP performs better when W_{iw} = 1 packet than when W_{iw} = 4 packets. To explain this we observe the number of packet losses and timeouts. It is found that the number of timeouts remains almost the same in both cases (W_{iw} = 4 packets and W_{iw} = 1 packet) but the number of packet losses is higher when W_{iw} = 4 packets than when W_{iw} = 1 packet. Since one or few packets can only be recovered per round trip time, more time will be needed to transfer a given file when W_{iw} = 4 packets than when W_{iw} =1 packet.

Table 4.2. Transfer times (in seconds) for *e* = 0.01

F (KB)	W_{iw} = 1 packet		W_{iw} = 4 packets	
	analysis	simulation	analysis	simulation
6	1.39	1.54	1.02	1.08
12	1.85	2.00	1.42	1.43
18	2.25	2.36	1.81	1.80
24	2.64	2.77	2.20	2.21
30	3.03	3.16	2.59	2.60

Table 4.3. Transfer times (in seconds) for *e* = 0.1

F (KB)	W_{iw} = 1 packet		W_{iw} = 4 packets	
	analysis	simulation	analysis	simulation
6	2.71	2.95	2.92	3.14
12	4.00	4.26	3.98	4.29
18	4.99	5.24	5.28	5.38
24	5.50	5.82	5.61	6.01
30	6.35	6.68	7.18	7.46

4.3.3 Effect of the Maximum Window Size

The variations of transfer time with file size (F) are observed at $W_{\max} = 4$ and 8 packets in a low error rate environment ($e = 0.01$) and high error environment ($e = 0.15$). In each case we set $l = 250$ ms, $BL = 1$, and $W_{\text{iw}} = 1$ packet. It is found that as the size of the transferred file increases in a low error environment (Figure 4.5), the difference between transfer times when $W_{\max} = 4$ packets and when $W_{\max} = 8$ packets increases significantly. However, in a high error environment (Figure 4.6), the increase of the difference between transfer times become less significant. This is due to the fact that with a sufficiently large bandwidth-delay product (BDP) in the low error environment, congestion window tends to grow to larger values as the size of the transfer file increases. Therefore, $W_{\max}$ becomes a limiting factor for congestion window.

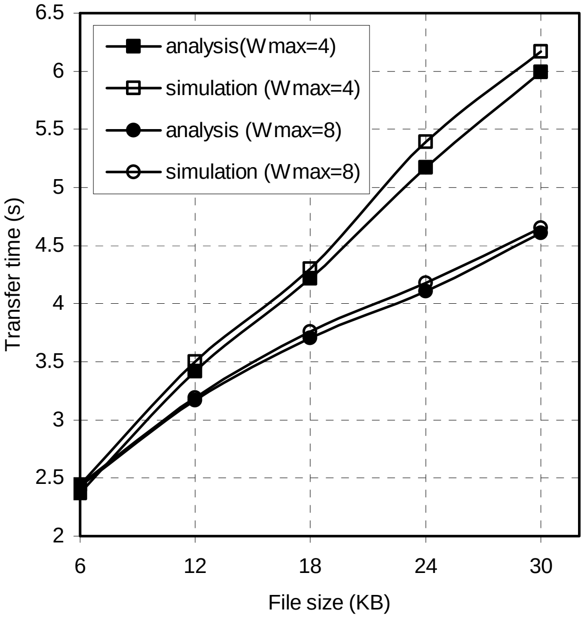


Figure 4.5. Transfer time vs. *F* for *e* = 0.01, $W_{\max}$ = 4 and 8 packets

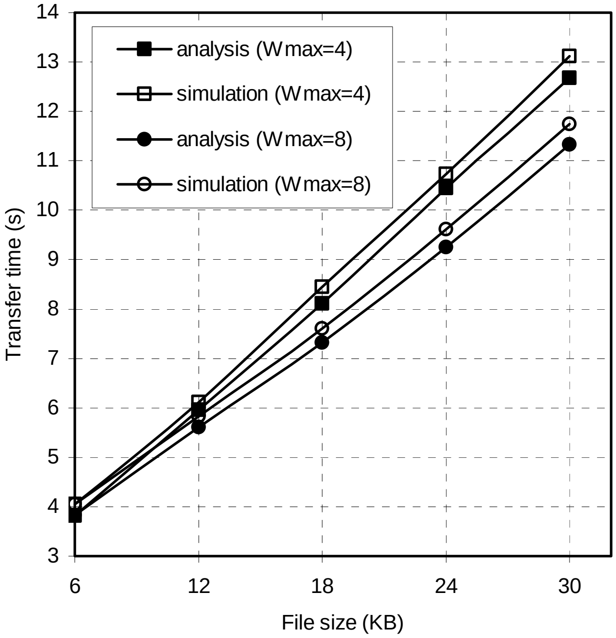


Figure 4.6. Transfer time vs. *F* for *e* = 0.15, $W_{\max}$ = 4 and 8 packets

In the case of high error environment, congestion window is mostly limited to low values by the WP-TCP congestion control response. Therefore $W_{\max}$ becomes less significant in dictating the size of congestion window. The transfer time is further studied for different values of *l* at $W_{\max}$ = 4 and 8 packets, in a low error rate environment (*e* = 0.01) and a high error environment (*e* = 0.15). In each case we set *F* = 30KB, *BL* = 1, and W_{iw} = 1 packet. As expected from previous explanations, the difference in transfer times when $W_{\max}$ = 4 packets and when $W_{\max}$ = 8 packets increases significantly in the low error environment (Figure 4.7) and slightly in the high error environment (Figure 4.8) as *l* increases.

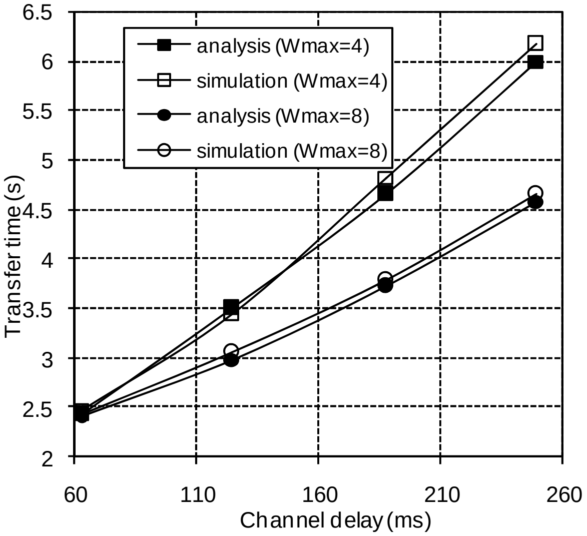


Figure 4.7. Transfer time vs. *l* for *e* = 0.01, $W_{\max}$ = 4 and 8 packets

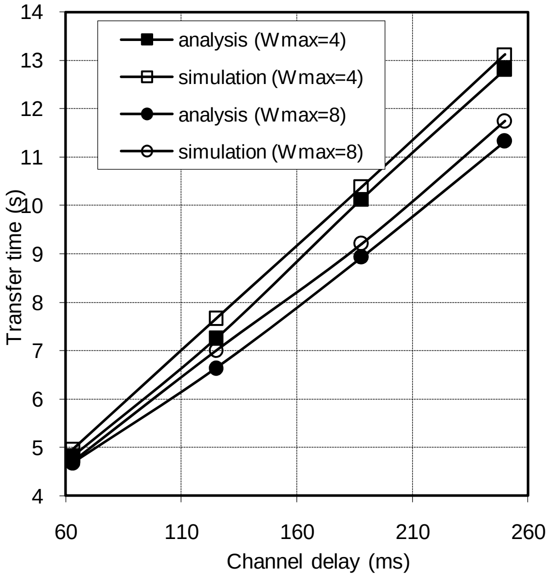


Figure 4.8. Transfer time vs. *l* for *e* = 0.15, $W_{\max}$ = 4 and 8 packets

4.4 Summary

In this chapter, an analytical framework for studying the transfer delay performance of WP-TCP short-lived flows is proposed. Computer simulation results demonstrate that the proposed analytical framework produces a good prediction of the WP-TCP transfer delay. These results show that for large file sizes (> 30KB), WP-TCP is more sensitive to bursty packet losses than random packet losses. Furthermore, the transfer delay of WP-TCP can be improved by increasing the size of the initial window in a low rate bursty error environment. However, in a high rate bursty error environment, a larger initial window degrades the transfer delay performance.

Chapter 5

Analytical Framework for Long-lived WP-TCP Flows

5.1 Introduction

Various studies related to WP-TCP long-lived flows have been reported in the literature. The work in [27], presents an overview of the different issues to be considered when modeling TCP over the Internet. In [25] and [26], analytical frameworks for studying the bulk throughput performance of several TCP variants have been developed based on a two-state Markov model. Since the two-state Markov model can capture the effects of wireless transmission errors in a great extent, the analytical frameworks in the literature can be used to study the performance of WP-TCP long-lived flows in integrated mobile WLAN and WWAN.

The connection life time of WP-TCP long-lived flows is significantly longer than the time between two successful vertical handovers. Therefore, contrary to the case of short-lived flows presented in Chapter 4, the effects of vertical handover on performance of WP-TCP long-lived flows in integrated static WLAN and WWAN can not be ignored. Recently, several studies on performance of TCP over integrated static WLAN and WWAN have been reported [28], [29], [30]. In these studies, the performance of TCP has been evaluated using simulation and/or experimentation techniques. To the best of our knowledge, there is no analytical study in the open literature that specifically focuses on the performance of WP-TCP over integrated static WLAN and WWAN.

This chapter complements the simulations and measurements studies by introducing an analytical framework for studying the performance of WP-TCP in integrated static WLAN and WWAN. For given network and protocol parameters, explicit mathematical expressions which describe the short- and long-term performance of WP-TCP under the influence of excessive packet losses and sudden change in network characteristics are derived. Extensive simulations are conducted to verify the reasonableness of the assumptions and the validity of the analytical expressions.

The remainder of this chapter is organized as follows. The analytical framework is developed in Section 5.2. Section 5.3 presents analytical and simulation performance results. Summary of this chapter is given in Section 5.4.

5.2 Analytical Framework

From Section 2.1, WWANs have wide coverage area. WLANs have small coverage area and they are mostly available within distinct areas called hotspots. Therefore, it is reasonable to assume that a single large cell of a WWAN is overlaid on disjoint WLAN hotspots. We further consider that the WLAN and WWAN network residence times are exponentially distributed random variables and the underlying WLAN and WWAN have sufficiently large buffers and fixed but arbitrary channel delays, bandwidths, and error rates.

From Section 2.3, WP-TCP has three transfer stages: connection establishment, data transfer, and connection tearing down. Since we are only interested in the performance of WP-TCP during vertical handover and the long-run average performance of long-lived WP-TCP flows, the connection establishment and tearing down stages are not considered. Figure 5.1 shows the transition model representing the WP-TCP transmission process. The WP-TCP sender remains in the transient congestion window phase (which is slow start or congestion avoidance phase) until the congestion window reaches the maximum window size (transition a), or the packet gets lost due to transmission error (transition b), or a vertical handover occurs (transition c). While in the steady congestion window phase (which is the maximum window size phase), the WP-TCP sender re-enters the steady congestion window phase if all packets in the same congestion window are successfully transmitted (transition d); otherwise the WP-TCP sender enters the packet recovery phase if there is a transmission error (transition e) or vertical handover (transition f). After the recovery due to transmission errors, the WP-TCP sender enters the transient congestion window phase (transition g) and continues with transmission. After the recovery due to handover, the WP-TCP sender enters the transient congestion window phase (transition h) if the congestion window is smaller than the maximum window size; otherwise, it enters the steady congestion window phase (transition i).

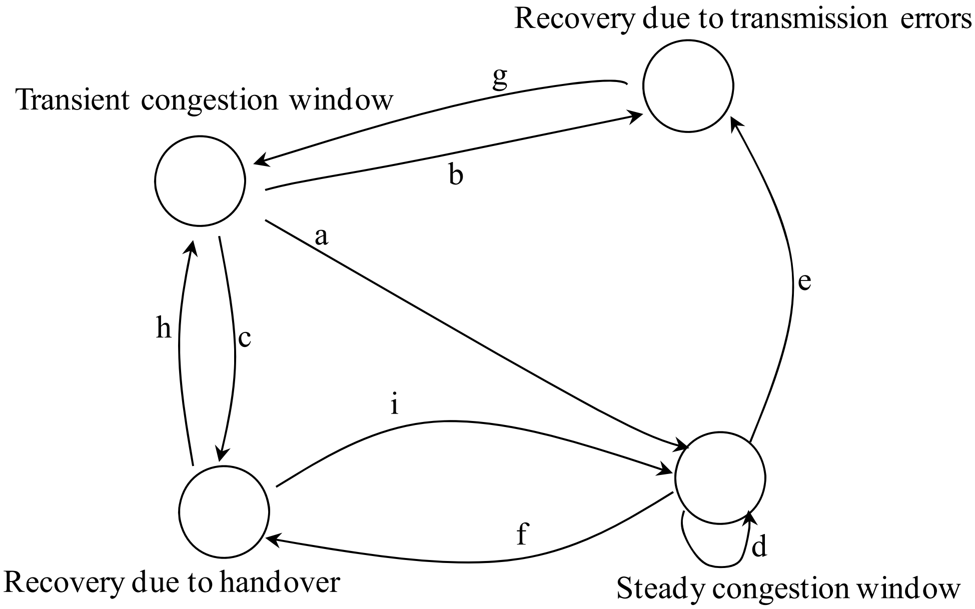

Recovery due to transmission errors
Transient congestion window
g
b
a
e
h
c
i
f
d
Recovery due to handover
Steady congestion window

Figure 5.1. Transition model for WP-TCP transmission process

For easy reference, the notions frequently used in this chapter are listed in Table 5.1. To track the packet transmission process (Figure 5.1), we consider a random process S(t) = ($i(t)$,$c(t)$,$w(t)$,$w_{th}(t)$), where $i(t) \in \{0,1\}$ is the state of communication channel (0 means WLAN and 1 means WWAN), $c(t) \in \{\text{g,b,h}\}$ is the status of the transmitted packet over the channel (g means no packet loss, b means packet loss due to transmission error, and h means packet loss due to handover), $w(t) \in \{1,2,\ldots, w_{\max}\}$ is the WP-TCP sender congestion window, $w_{\text{th}}(t) \in \{2,3,\ldots, \lceil w_{\max}/2 \rceil\}$ is the WP-TCP sender slow start threshold, t is the time measured in slots, and $w_{\max}$ is the WP-TCP sender maximum window size. By sampling the random process S(t) at the beginning of each phase, the sampled process is a semi-Markov process [70] with embedded Markov chain in the state space $W_s = (i, c, w, w_{th}),\ \forall i, c, w, w_{th}$. The embedded Markov chain is defined by the transition matrix $\mathbf{\Phi} = [\phi_{AB}]$, where ϕ_{AB} denotes the transition probability from state $A \in W_s$ to state $B \in W_s$. To compute the long-term performance, the transitions of embedded Markov chain are labeled with the corresponding elements of the matrices $\mathbf{S} = [s_{AB}]$, $\mathbf{D} = [d_{AB}]$, and $\mathbf{M} = [m_{AB}]$. The elements s_{AB}, d_{AB}, and m_{AB} are respectively the number of packets transmitted successfully, the transition time, and the number of packets transmitted from state $A \in W_s$ to state $B \in W_s$.

The long-term performance metrics are throughput rate and efficiency. Throughput rate is defined as the number of packets transmitted successfully per unit time, whereas the efficiency is defined as the ratio of the number of packets transmitted successfully to the number of packets transmitted. The long-run average throughput rate (Ω) and efficiency (Λ) can be computed as

$$\Omega = \lim_{t \to \infty} S(t)/t = \sum_{A \in W_s} \pi_A \sum_{B \in W_s} \phi_{AB} s_{AB} \Big/ \sum_{A \in W_s} \pi_A \sum_{B \in W_s} \phi_{AB} d_{AB} \tag{5.1}$$

and

$$\Lambda = \lim_{t \to \infty} S(t)/M(t) = \sum_{A \in W_s} \pi_A \sum_{B \in W_s} \phi_{AB} s_{AB} \Big/ \sum_{A \in W_s} \pi_A \sum_{B \in W_s} \phi_{AB} m_{AB}\,, \tag{5.2}$$

where $S(t)$ and $M(t)$ denote the number of packets transmitted successfully and the number of packets transmitted by time t, respectively. π_A is the steady state probability of being in state A and can be found by manipulating the matrix $\mathbf{\Phi}$ as in [70]. In each phase (Figure 5.1) the possible initial states and transitions to the final states are explored. For every possible transition, the corresponding transition time, the number of packets transmitted successfully, and the number of packets transmitted are computed. In the rest of this section, the functions $T(.,.,.)$, $S(.,.,.)$, and $W(.,.,.)$, which are defined in Table 5.1, are used to facilitate the analysis. These functions are deterministic and they can be easily tabulated or derived as in Chapter 4.

Table 5.1. List of local notions for Chapter 5

Notation	Definition (Unit)
λ_i	mean residence times for network *i* (sec)
b_i	link bandwidth for network *i* (bits/sec)
l_i	one-way channel delay for network *i* (sec)
p_i	transmission error rate for network *i*
w	WP-TCP congestion window (packets)
w_{th}	WP-TCP slow start threshold (packets)
$w_{\max}$	WP-TCP maximum window size (packets)
Z	WP-TCP duplicate acknowledgement threshold (packets)
$W(s, w_i, w_{\mathrm{th}})$	returns the final congestion window measured in packets given that the number of packet transmitted successful over the ideal channel is s, the initial congestion window is w_i, and the slow start threshold is w_{th}.
$T(s, w_i, w_{\mathrm{th}})$	returns the time taken measured in seconds given that the number of packet transmitted successful over the ideal channel is s, the initial congestion window is w_i, and the slow start threshold is w_{th}.
$N(w_i, w_{\mathrm{th}}, w_f)$	returns the number of packet transmitted successful over the ideal channel given that the initial congestion window is w_i, the slow start threshold is w_{th}, and the final congestion window is s.
$\mathbf{\Phi} = [\phi_{AB}]$	transition matrix of the embedded Markov chain, where an element ϕ_{AB} denotes transition probability from state *A* to state *B*
$\mathbf{S} = [s_{AB}]$	matrix corresponding to the embedded Markov chain, where an element s_{AB} number of packets transmitted successfully from state *A* to state *B*
$\mathbf{D} = [d_{AB}]$	matrix corresponding to the embedded Markov chain, where an element d_{AB} transition time from state *A* to state *B* in seconds
$\mathbf{M} = [m_{AB}]$	matrix corresponding to the embedded Markov chain, where an element m_{AB} number of packets transmitted from state *A* to state *B*

5.2.1 Transient Congestion Window

The transmission process enters the transient congestion phase when a packet loss is recovered by using the timeout mechanism or the duplicate acknowledgment mechanism. Therefore, a set of possible initial states is $A = (i, g, w, w_{\mathrm{th}}), \forall i, w = \{1, w_{\mathrm{th}}\}, w_{\mathrm{th}}$. Let t and $s \in \{0,1,..., N(w, w_{\mathrm{th}}, w_{\max})\}$ be the network residence time and the number of packets transmitted successfully before the transition occurs, respectively. Given the initial state *A* and the number of successfully transmitted packets *s*, the final state can be found as

$$B = \begin{cases} (i, h, W(s, w, w_{th}), w_{th}), & T(s+1, w, w_{th}) \geq t > T(s, w, w_{th}), s < N(w, w_{th}, w_{\max}) \\ (i, b, W(s, w, w_{th}), w_{th}), & t > T(s+1, w_{\max}, w_{\max}), s < N(w, w_{th}, w_{\max}) \\ (i, g, W(s, w, w_{th}), w_{th}), & t > T(s, w_{\max}, w_{\max}), s = N(w, w_{th}, w_{\max}). \end{cases} \tag{5.3}$$

The first, second, and third equations in (5.3) correspond to transitions c, b, and a shown in Figure 5.1, respectively. Let $\beta(s)$ denote the probability of successfully transmitting *s* packets given that the initial and final states are *A* and *B*, respectively. It follows that

$$\beta(s) = \begin{cases} 1 - e^{-\lambda_i T(1, w, w_{th})}, & t \geq T(1, w, w_{th}), s = 0 \\ e^{-\lambda_i T(1, w, w_{th})} \cdot p_i, & t > T(1, w, w_{th}), s = 0 \\ \left(e^{-\lambda_i T(s, w, w_{th})} - e^{-\lambda_i T(s+1, w, w_{th})}\right) \cdot (1 - p_i)^s, & T(s+1, w, w_{th}) \geq t > T(s, w, w_{th}), \\ & 0 < s < N(w, w_{th}, w_{\max}) \\ e^{-\lambda_i T(s+1, w, w_{th})} \cdot (1 - p_i)^s \cdot p_i, & t > T(s+1, w, w_{th}), 0 < s < N(w, w_{th}, w_{\max}) \\ e^{-\lambda_i T(s, w, w_{th})} \cdot (1 - p_i)^s, & t > T(s, w, w_{th}), s = N(w, w_{th}, w_{\max}). \end{cases} \tag{5.4}$$

From (5.3) and (5.4) there are several values of *s* that can result in the same final state *B* given that the initial state is *A*. Let U_{AB} denote a set of possible values of *s* such that the initial state is *A* and the final state is *B*. The transition probability, the transition time, the number of packets transmitted successfully and the number of packets transmitted can respectively be represented as

$$\phi_{AB} = \sum_{s \in U_{AB}} \beta(s), \tag{5.5}$$

$$d_{AB} = \frac{\sum_{s \in U_{AB}} \beta(s) \cdot T(s)}{\sum_{s \in U_{AB}} \beta(s)}, \tag{5.6}$$

and

$$s_{AB} = m_{AB} = \frac{\sum_{s \in U_{AB}} \beta(s) \cdot s}{\sum_{s \in U_{AB}} \beta(s)}. \tag{5.7}$$

5.2.2 Steady Congestion Window

The transmission process enters the maximum window phase when the congestion window reaches the maximum window size. Therefore, a set of possible initial states is $A = (i, g, w_{\max}, w_{th}), \forall i, w_{th}$. Similar to Subsection 5.2.1, for $s \in \{0,1,\ldots, w_{\max}\}$, the final state can be found as

$$B = \begin{cases} (i, h, w_{\max}, w_{th}), & T(s+1, w_{\max}, w_{\max}) \geq t > T(s, w_{\max}, w_{\max}), s < w_{\max} \\ (i, b, w_{\max}, w_{th}), & t > T(s+1, w_{\max}, w_{\max}), s < w_{\max} \\ (i, g, w_{\max}, w_{th}), & t > T(s, w_{\max}, w_{\max}), s = w_{\max}. \end{cases} \tag{5.8}$$

The first, second, and third equations in (5.8) correspond to transitions f, e, and d shown in Figure 5.1, respectively. The corresponding transition probability, the transition time, the number of packets transmitted successfully and the number of packets transmitted can be similarly computed as in Subsection 5.2.1.

5.2.3 Recovery Due to Transmission Errors

All transitions to the loss recovery due to transmission error happen just before the first packet loss occurs. Therefore, a set of initial states is given as $A = (i, \mathrm{b}, w, w_{\mathrm{th}}), \forall i, w, w_{th}$. Let $s \in \{0,1,\ldots,$ w-1$\}$ be the number of packets successfully transmitted out of w initially transmitted packets after the first packet loss and $x \in \{0,1,\ldots,$ w-$s\}$ be the number of packets successfully retransmitted in the fast recovery phase. Then, given the initial state A, the final state can be found as

$$B = \begin{cases} (i, \mathrm{g}, 1, \max(2, \lfloor (w-Z)/2 \rfloor)), & s < Z \text{ or } s \geq Z, x < w-s \\ (i, \mathrm{g}, w/2, \max(2, \lfloor (w-Z)/2 \rfloor)), & s \geq Z, x = w-s. \end{cases} \tag{5.9}$$

The first equation in (5.9) corresponds to the case when the timeout mechanism is used (i.e., when the total number of duplicate ACKs is less the Z, or any of the retransmitted lost packets get lost again). The second equation in (5.9) corresponds to the case when the timeout mechanism is not used (i.e., when there is enough duplicate ACKs and all retransmitted packets are successful). Let $\alpha(s, w)$ be the probability of having successfully transmitted s packets out of w transmitted packets given that the packet transmission error rate is p_i. It follows that

$$\alpha(w, s) = \binom{w}{s} p_i^{\,w-s} (1 - p_i)^s \tag{5.10}$$

From (5.9) and (5.10) the expression for probability that $s \geq Z$ and $x = w - s$ can be found as

$$\sum_{s=z}^{w-1} \alpha(w-1,s)(1-p_i)^{w-s}. \tag{5.11}$$

Given the initial state is *A* and the final state is *B*, the transition probability can be computed as

$$\phi_{AB} = \begin{cases} 1 - \sum_{s=z}^{w-1} \alpha(w-1,s)(1-p_i)^{w-s}, & s < Z \text{ or } s \geq Z, x < w-s \\ \sum_{s=z}^{w-1} \alpha(w-1,s)(1-p_i)^{w-s}, & s \geq Z, x = w-s. \end{cases} \tag{5.12}$$

The transition time in this loss recovery phase is considered to be an interval between a time slot when the ACK for the lost packet is received (entering the loss recovery phase) and the time slot when the timeout is expired or when a full ACK of retransmitted packet is received (exiting the loss recovery phase). When $s < Z,$ the timeout mechanism is used and the transition time is simply *RTO*. To compute the transition time for $s \geq Z$, we further consider two time intervals: 1) the time interval between entering the loss recovery phase and starting fast retransmit, and 2) the time interval between starting fast retransmit and exiting the loss recovery phase. These time intervals can be computed by conditioning the order and the number of packets successfully transmitted before fast retransmit and successfully retransmitted in fast retransmit and fast recovery. However, with this approach, the analysis can become extremely tedious and complex. In the following, we use the close bounds of both time intervals. We assume that lost packets from the window *w* are retransmitted and if all are successful the fast recovery ends in a round-trip time *RTT*; otherwise, if any of them gets lost the retransmission timeout timer expires after the time interval *RTO*. Note that the approximation underestimates the actual recovery time. However, the deviation is expected to be reasonable due to the fact that by enabling the SACK option all lost packets in the same sender's window are reported to the WP-TCP sender within one round-trip time. The transition time can therefore be estimated as

$$d_{AB} = \begin{cases} \sum_{s=0}^{z-1} RTO \cdot \alpha(w-1,s) + \sum_{s=z}^{w-1} (RTT + RTO) \cdot \alpha(w-1,s), & s < Z \text{ or } s \geq Z, x < w-s \\ 2RTT, & s \geq Z, x = w-s, \end{cases} \tag{5.13}$$

where $RTO = \max\left\{RTO_{\min}, wMSS / b_i, 2l_i + MSS / b_i\right\}$ and $RTT = \max\left\{wMSS / b_i, 2l_i + MSS / b_i\right\}$. The number of packets transmitted successfully and the number of packets transmitted can respectively be found as

$$s_{AB} = \begin{cases} \sum_{s=0}^{z-1} s \cdot \alpha(w-1,s) + \sum_{s=z}^{w-1} \left(s + \frac{\sum_{k=0}^{w-s-1} k \cdot \alpha(w-s,k)}{\sum_{k=0}^{w-s-1} \alpha(w-s,k)} \right) \alpha(w-1,s), & s < Z \text{ or } s \geq Z, x < w-s \\ w, & s \geq Z, x = w-s \end{cases} \tag{5.14}$$

and

$$m_{AB} = \begin{cases} \sum_{s=0}^{z-1} w \cdot \alpha(w-1,s) + \sum_{s=z}^{w-1} (w + (w-s)) \cdot \alpha(w-1,s), & s < Z \text{ or } s \geq Z, x < w-s \\ \frac{\sum_{s=z}^{w-1} (w + (w-s)) \cdot \alpha(w-1,s)}{\sum_{s=z}^{w-1} \alpha(w-1,s)}, & s \geq Z, x = w-s. \end{cases} \tag{5.15}$$

5.2.4 Recovery Due to Handover

5.2.4.1 Soft Downward Vertical Handover

After observing the transmission process during soft downward vertical handover, we find that fast transmit cannot occur if at least one of the following conditions is true.

1) If the number of burst data packet injected in the new link, which can potentially overtake data packets in the old link, is less than duplicate ACK threshold (*Z*).
2) If the number of data packets that can be transmitted from WLAN access point in the interval of two consecutive data packet transmitted from the WWAN base station is less than *Z*.
3) If the transmission of the last data packets in the old link is completed before the number duplicate ACKs generated by reorder data packet reaches duplicate ACK threshold.

In order to compute the minimum duplicate ACK threshold $(Z_{\min})$ over which fast transmit cannot occur, we consider the above conditions in the following cases.

Case I: When $wMSS/b_1 > 2l_1$. Let Z_i, $i = 1, 2, 3,$ denote the minimum *Z* which satisfies the condition *i*. Since the burst data packet can only occur if ACK sent through upward WLAN overtakes the ones in WWAN, the condition 1 is satisfied when $Z_1 MSS/b_1 + l_2 > l_1$. For condition 2, the transmission time of Z_2 reordered packets in WLAN must be greater than the transmission time of a packet in WWAN. Therefore, condition 2 is satisfied if $Z_2 MSS/b_2 > MSS/b_1$. After initiating the handover, the time elapsed before the completion of transmission of the last data packet in the WWAN is given as $(wMSS/b_1 - l_1)$. And the time needed for duplicate ACKs, generated by reordering data packets, to reach Z_3 is given as $(2l_2 + Z_3 MSS/b_2)$. Therefore, condition 3 is satisfied if $(2l_2 + Z_3 MSS/b_2)$ >

$(wMSS/b_1 - l_1)$. The minimum duplicate ACK threshold is computed as $Z_{\min} = \min\{Z_1, Z_2, Z_3, w\}$.

Case II: When $wMSS/b_1 \le 2l_1$. In this case, the occurrence of fast transmit further depends on the instance that the handover occurs in a round trip time. Therefore, we consider a reasonable upper-bound. Since when Z is equal to or greater than *w* fast transmit cannot occur, the minimum duplicate ACK threshold is approximated as $Z_{\min} = w$.

Given the set of initial states before handover as $A = (1, \mathrm{h}, w, w_{\mathrm{th}}), \forall w, w_{th}$, the final state after handover can be found as

$$B = \begin{cases} (0, \mathrm{g}, W(w, w, w_{th}), w_{th}), & Z \ge Z_{\min} \\ (0, \mathrm{g}, (w-Z)/2, \max(2, \lfloor (w-Z)/2 \rfloor)), & \text{otherwise,} \end{cases} \tag{5.16}$$

The first equation in (5.16) corresponds to the case in which duplicate ACK threshold is equal or greater than $Z_{\min}$ and therefore fast transmit is not triggered. The second equation in (5.16) corresponds to the case in which duplicate ACK threshold is less than $Z_{\min}$, and therefore fast transmit is triggered. Note that when fast transmit is triggered, the slow start threshold and congestion window are set to half of the *pipe* (number of unacknowledged packets $(w-Z)$). The transition probability and the number of packets transmitted successfully can respectively be found as $\phi_{AB} = 1$ and $s_{AB} = w$.

The recovery process for soft downward vertical handover will start right after the handover initiation and end after receiving an ACK of the last data packet sent before the handover initiation. Therefore, the transition time is approximately equal to a round trip time experienced by a data packet transmitted through the WWAN and ACK packet returned through the WLAN, which is given as $d_{AB} = \max\{wMSS/b_1 - l_1 + l_0, MSS/b_1 + l_1 + l_0\}$. Note that if the WLAN round trip time is extremely small compared to the WWAN round trip time, fast transmit followed by fast recovery can end before the ACK packets, which correspond to data packet transmitted through the WWAN, arrive at the WP-TCP sender. In this case, the proposed model slightly overestimates the transition time. The number of packets transmitted includes all in-flight data packets just before handover (which is equal to *w*) plus unnecessary retransmitted packets, if fast transmit is triggered. Because of go-back-N retransmission behavior of WP-TCP, number of unnecessary retransmissions is close to half of the *pipe* just before the fast transmit (i.e., $(w-Z)/2$). Therefore, the number of packets transmitted can be found as

$$m_{AB} = \begin{cases} w, & wMSS/b_1 \le 2l_1 \\ w+(w-Z)/2, & \text{otherwise.} \end{cases} \tag{5.17}$$

5.2.4.2 Soft Upward Vertical Handover

Let RTO and Y denote retransmission timeout set at the WP-TCP sender just before the initiation of the handover and the time difference between the arrival of the last ACK from WLAN and the first ACK from WWAN at the WP-TCP sender, respectively. Note that the step increase in round trip time suddenly increases the magnitude of Y. During time interval Y, the retransmission timeout timer at the WP-TCP sender counts down the clock. Therefore, false timeouts can occur if $Y > RTO\sum_{k=1}^{n} 2^{k-1}$, where n is the number of consecutive premature timeouts and it can be found as $n = \lfloor \log_2(1+Y/RTO) \rfloor$. Retransmission timeout can be approximated as $RTO = \max\{RTO_{\min}, wMSS/b_0, 2l_0 + MSS/b_0\}$. In order to compute Y, we consider the following cases.

Case I: When $2l_0b_0 / MSS < w$. In this case, the WP-TCP receiver is always busy receiving data packet and sending ACK. Therefore, Y is approximately $(l_1 - l_0)$.

Case II: When $l_0b_0 / MSS < w \leq 2l_0b_0 / MSS$. If handover occurs when the WP-TCP receiver is busy on receiving data packet and sending ACK packets, then Y is approximately equal to $(l_1 - l_0)$. Otherwise, if handover occurs when the WP-TCP receiver is idle waiting for in-flight data packets to arrive from the WP-TCP sender, Y is approximately $(l_0 - wMSS / b_0 + l_1)$.

Case III: When $w \leq l_0b_0 / MSS$. If handover occurs when the WP-TCP receiver is busy on receiving data packet and sending ACK packets, then Y is approximately equal to $(l_1 - l_0)$. Or, if handover occurs when the WP-TCP receiver is idle waiting for in-flight data packets to arrive from the WP-TCP sender, then Y is approximately equal to $(l_0 - wMSS / b_0 + l_1)$. Otherwise, if handover occurs when the WP-TCP receiver is idle waiting for in-flight ACK packets to arrive to the WP-TCP sender, then Y is approximately $(2l_1 - wMSS / b_0)$.

From the above three cases, the value of Y corresponding to the worst, average, or best timeouts performance can be computed. For the sake of brevity, only the worst timeout performance (i.e., consider the maximum value of Y for each case) is considered in this paper. Therefore, the maximum time difference between the arrival of the last ACK from the WLAN and the first ACK from the WWAN at the WP-TCP sender is given as

$$Y = \begin{cases} l_1 - l_0, & 2l_0b_0 / MSS < w \\ l_0 - wMSS / b_0 + l_1, & l_0b_0 / MSS < w \leq 2l_0b_0 / MSS \\ 2l_1 - wMSS / b_0, & w \leq l_0b_0 / MSS. \end{cases} \tag{5.18}$$

Given the set of initial states before handover as $A = (1, \mathrm{h}, w, w_{\mathrm{th}}), \forall w, w_{th}$, the final state after handover can be found as

$$B = \begin{cases} (1, \mathrm{g}, W(w, w, w_{th}), w_{th}), & Y \le RTO \\ (1, \mathrm{g}, W(w, 1, \max(2, \lfloor w/2 \rfloor)), \max(2, \lfloor w/2 \rfloor)), & RTO < Y \le 3RTO \\ (1, \mathrm{g}, W(w, 1, 2), 2), & 3RTO < Y, \end{cases} \tag{5.19}$$

The first equation in (5.19) corresponds to the case where the timeout does not occur. The second equation in (5.19) corresponds to the case where a single timeout occurs. The third equation in (5.19) corresponds to the case where two or more timeouts occur. Note that after a single timeout, the slow start threshold is set to half of the current congestion window and the congestion window is set to one. After two or more timeouts slow start threshold is set to two (the minimum value) and the congestion window is set to one. The transition probability can be found as $\phi_{AB} = 1$ and the number of packets transmitted successfully can be computed as $s_{AB} = w.$ The transition time can be written as

$$d_{AB} = \begin{cases} \max\{wMSS/b_0 - l_0 + l_1, MSS/b_0 + l_0 + l_1\}, & Y \le RTO \\ (MSS/b_1 + 2l_1) + (wMSS/b_1 + 2l_1), & \text{otherwise.} \end{cases} \tag{5.20}$$

Equation (5.20) can be explained as follows. If there is no timeout during handover (i.e., when $Y \le RTO$), the transition time is equal to a round trip time, where the data packet is sent through the WLAN and the corresponding ACK packet is returned through the WWAN. Otherwise, if there is a single or multiple timeouts during handover (i.e., when $Y > RTO$) the transition time is found by summing two round trip times: a round trip time for a data packet sent through the WWAN after handover ($MSS/b_1 + 2l_1$) and the round trip time of the new packet sent through the WWAN after retransmitting the entire window $(wMSS/b_1 + 2l_1)$. Note that if there is a single or multiple timeouts, the transition time is considered to start right after the handover initiation and end after receiving an ACK of the new packet sent after the timeout. The packets transmitted include all in-flight data packets just before handover (which is equal to *w*) and unnecessarily retransmitted packets due to single or multiple timeouts. Because of the go-back-N retransmission behavior of WP-TCP, the number of unnecessary retransmissions is approximately $(n+w-1)$, where *n* is the number of consecutive premature timeouts. Therefore, the number of packets transmitted can be found as

$$m_{AB} = \begin{cases} w, & Y \le RTO \\ w+(n+w-1), & \text{otherwise.} \end{cases} \tag{5.21}$$

5.2.4.3 Hard Downward and Upward Vertical Handovers

After hard downward or upward vertical handover all in-flight packets get lost. Under this conjecture, a similar model for both hard downward and upward vertical handovers is developed. Let $\overline{i}$ denote the network other than *i* (for example, if *i* = 1(WWAN) then $\overline{i}$ = 0 (WLAN)). Therefore, the round trip time experienced by the first packet retransmitted through network $\overline{i}$ (the new network) is $RTT = (2l_{\overline{i}} + MSS/b_{\overline{i}})$ and the WP-TCP sender retransmission timeout is $RTO = \max\{RTO_{\min}, wMSS/b_i, 2l_i + MSS/b_i\}$. Given the set of initial states as $A = (i, \mathrm{h}, w, w_{\mathrm{th}}), \forall i, w, w_{th}$, the final state can be deterministically found as

$$B = \begin{cases} (\overline{i}, \mathrm{g}, 1, \max(2, \lfloor w/2 \rfloor)), & RTT \le 2RTO \\ (\overline{i}, \mathrm{g}, 1, 2), & \text{otherwise,} \end{cases} \tag{5.22}$$

The transition probability, the transition time, and the number of successfully transmitted packets can be found as $\phi_{AB} = 1$, $d_{AB} = RTO$, and $s_{AB} = 0$, respectively. Since the timeout mechanism is only used to detect and recover lost packets after handover, at least one timeout must occur. After the first timeout, k additional timeouts can occur if $RTT > RTO\sum_{j=1}^{k} 2^j$. The number of additional timeouts can be found as $k = \lfloor \log_2(1+0.5RTT/RTO) \rfloor$. The number of packets transmitted is the sum of all in-flight data packets just before handover (which equals to congestion window) and the number of retransmitted packets due to timeouts. Therefore, the number of packets transmitted can be found as $m_{AB} = w + 1 + \lfloor \log_2(1+0.5RTT/RTO) \rfloor$.

5.3 Numerical Results and Discussion

The proposed analytical framework is verified by simulation using ns-2 simulator [69]. The simulation topology is shown in Figure 5.2. A single pair of wireless profiled TCP (WP-TCP) sender-receiver is configured to run over the integrated WLANs and WWAN. FTP is used as an application for unidirectional downlink transfer. WP-TCP is simulated according to the real protocol as described in Section 2.3. The underlying network is simulated to capture the fundamental effects of the vertical handover. Therefore, the results from simulations may be slightly inflated compared to those obtained from the real network. However, the trends related to vertical handover are similar. The simulation and analytical results are obtained by using the parameter values and units given in Table 5.2. Note that the notation for ordered pair (WWAN, WLAN) is consistently used to present network parameters.

In the next subsections, we evaluate the short- and long-term performances of WP-TCP under the influence of vertical handover. For short-term performance, we first analyze the occurrence and the impact of falsely triggered fast retransmits in 60 seconds after the soft upward vertical handover. We analyze the occurrence and impact of premature timeouts in 60 seconds after the soft downward vertical handover. For long-term performance, we analyze the long-run efficiency and throughput under the influence of the hard vertical handovers. In this case, each simulation is run for sufficiently long time to obtain the average performance. Note that if not stated otherwise, BDP denotes the bandwidth-delay product of the network in use before the vertical handover. It is measured in packets and computed as BDP = $2lb/MSS + 1$.

From Figures 5.3–5.9, it can be seen that despite of approximations introduced in Section 5.2, the analytical results are in agreement with the simulation results. This verifies the reasonableness of our assumptions and correctness of the analysis.

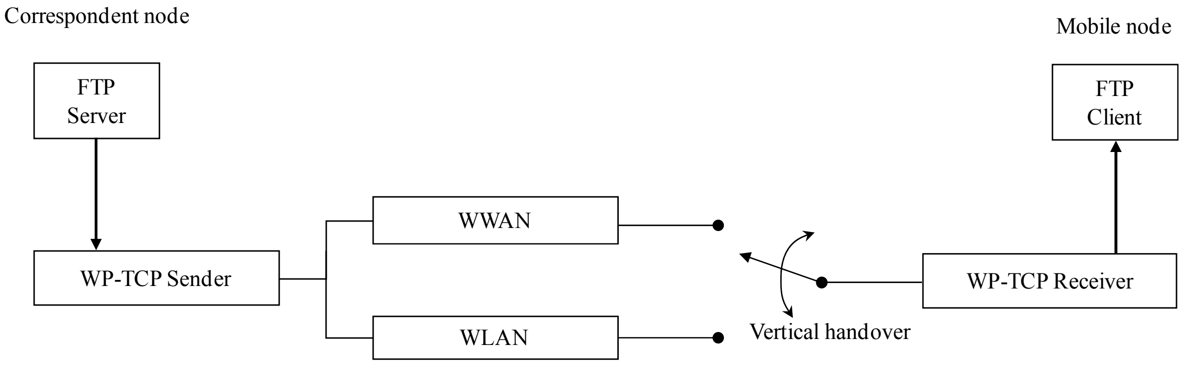


Figure 5.2. Simulation topology

Table 5.2. Simulation parameters

† Mean residence times (λ_1, λ_0)	(200,200) sec
† Link bandwidth (b_1, b_0)	(144K,2M) bps
† One-way channel delay (l_1, l_0)	(0.01–1,0.01– 0.1) sec
† Transmission error rate (p_1, p_0)	(0.00–0.15,0.00–0.02)
† Network buffer size (B_1, B_0)	$(2w_{\max}, 2w_{\max})$
WP-TCP maximum window size $(w_{\max})$	5–65 packets
WP-TCP duplicate acknowledgement threshold (Z)	1–15 packets
Minimum retransmission timeout $(RTO_{\min})$	0.2sec
WP-TCP maximum segment (packet) size (*MSS*)	1KB
Number of repeated runs with different random seeds per simulation	200

† Ordered pair notation (WWAN, WLAN) is used to present the network parameters.

5.3.1 Short-term Fast Retransmit Performance

The percentage of false fast retransmits and the average number of packets retransmitted after the soft downward handover are investigated at various values of duplicate ACK threshold (*Z*) and congestion window (*w*). In each case, the WWAN and WLAN one-way channel delay are set to 0.5 sec and 0.1 sec, respectively. In Figure 5.3a, the handover is initiated when *w* = 6, 12, and 24 packets. It can be seen that when the congestion window (*w* = 24 packets) is greater than the BDP, the occurrence of false fast retransmits changes from 100% to zero at *Z* = 8. On the other hand, when the congestion window (*w* = 12, 6 packets) is less than BDP false fast transmits occur for *Z* < *w*. To get more insight on this trend, the relationship between the percentage of false fast retransmits and the congestion window is analyzed for *Z* = 7 and 8 (Figure 5.3b). It can

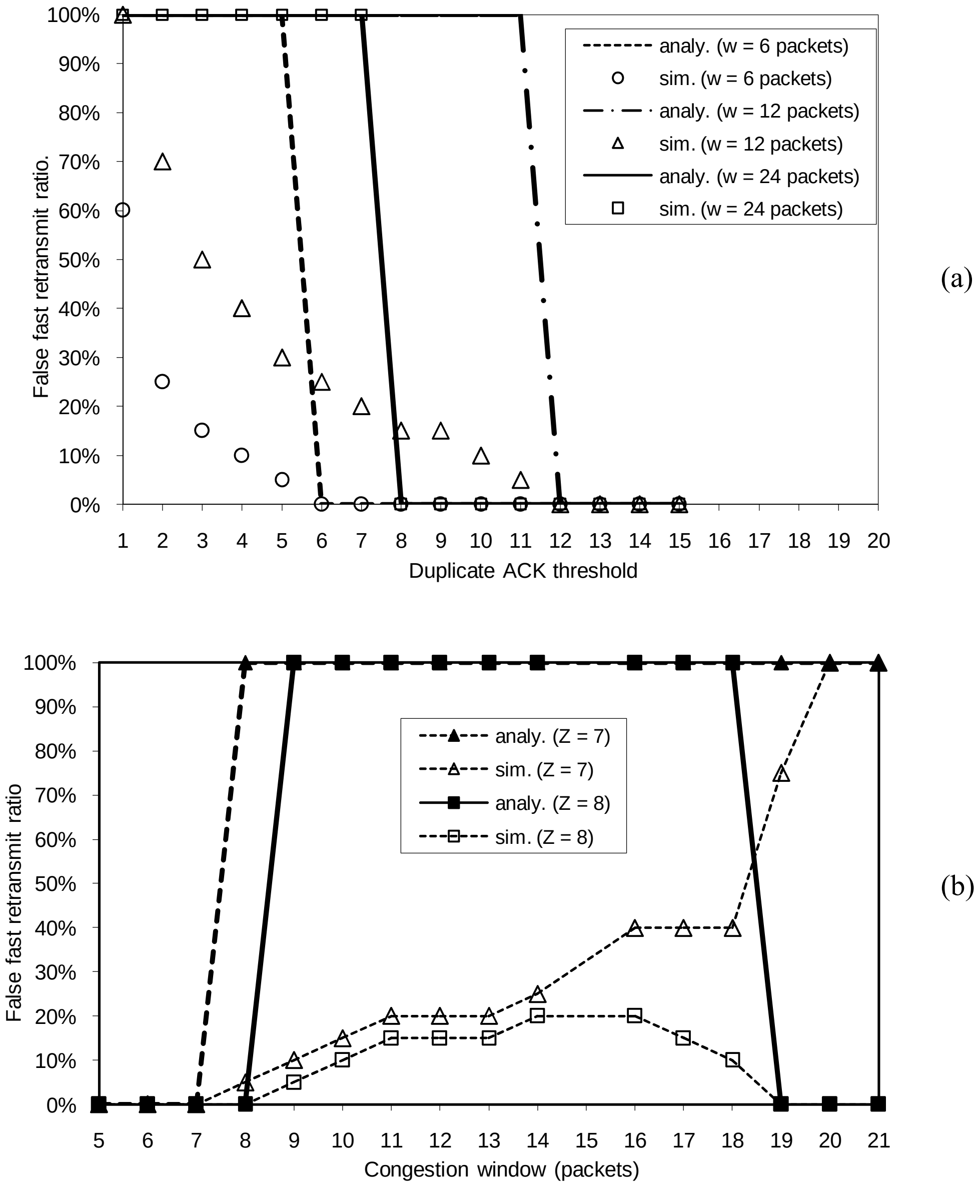


Figure 5.3. False fast retransmit ratio

be seen that when $Z < 8$, the percentage of false fast retransmits increases to 100% as w approaches the BDP. However, when $8 \leq Z$, the percentage of false fast retransmits increases and then decreases to zero as w reaches the BDP. These results reveal unexpected trend of the minimum Z which prevents the occurrence of false fast transmits as w increases. Figure 5.4 presents the average number of packets unnecessarily retransmitted (if fast transmit occurs) as a function of duplicate ACK threshold. It is shown that the number of retransmitted packets increases as the congestion window increases and decreases as Z decreases. This trend is explained by the go-back-N retransmission behavior of WP-TCP, where the number of retransmitted packets is proportional to the number of outstanding unacknowledged packets $(w - Z)$.

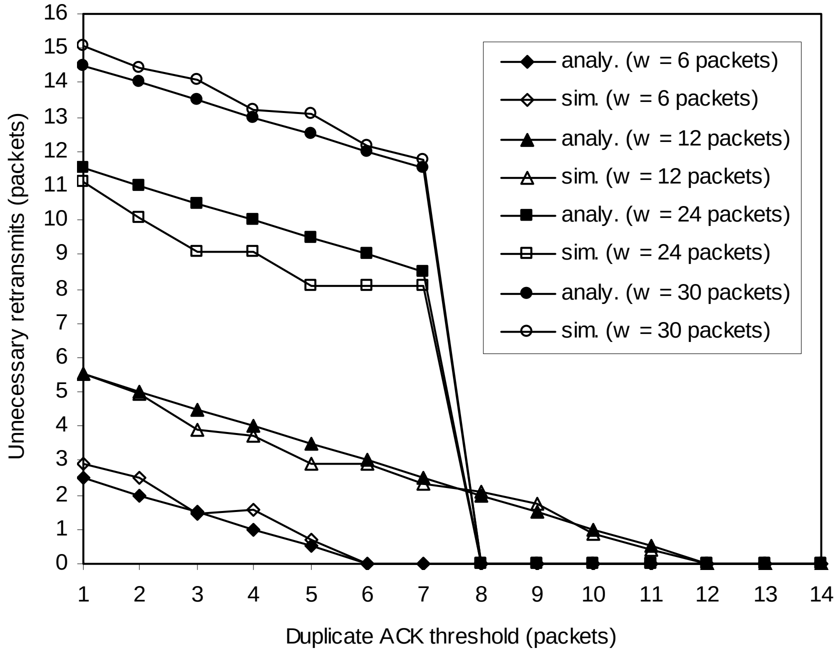


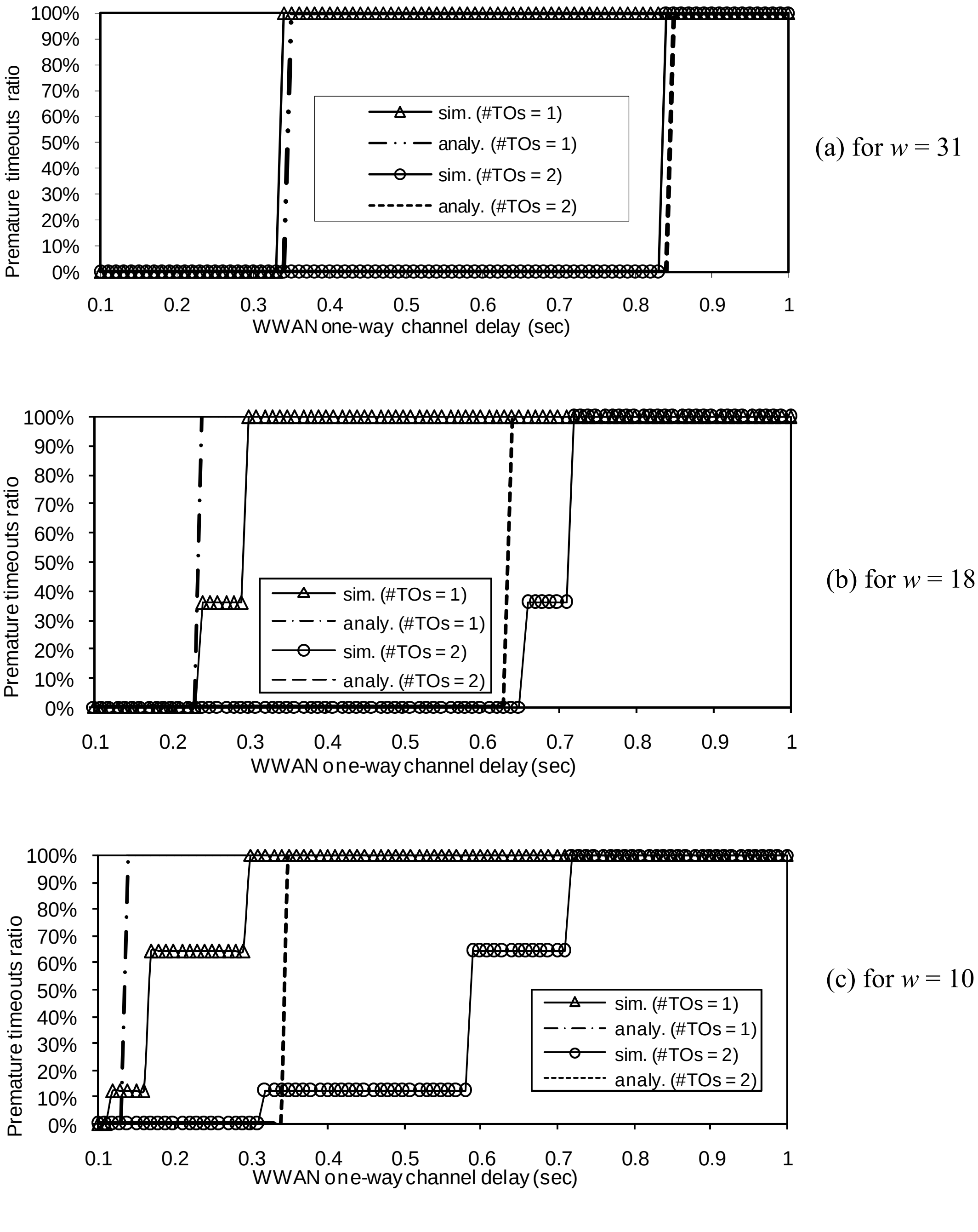
Figure 5.4. Average number of packets unnecessarily retransmitted

5.3.2 Short-term Retransmission Timeout Performance

Premature timeouts that occur during the soft upward handover is studied. The duplicate ACK threshold is set to 3 and the WLAN one-way channel delay is set to 0.1 sec. The handover is initiated when the size of congestion window is 10, 18, and 30. Figure 5.5 presents the percentage of single (#TO = 1) and two consecutive (#TO = 2) premature timeouts as a function of the WWAN one-way channel delay. It can be seen that the lowest WWAN one-way channel delay that can result in single or two consecutive premature timeouts decreases as the congestion window decreases. The observed trend can be explained as follows. The timeouts can occur when the retransmission timeout is less than the maximum time between the arrival of the last ACK from the WLAN and the first ACK from the WWAN. When the congestion window is greater than the BDP, decrease in congestion window decreases the retransmission time but does not affect the maximum time between the arrivals of the two ACKs. And when the congestion window is less than the BDP, decrease in congestion window does not affect the retransmission time but increases the maximum time between the arrivals of the two ACKs.

5.3.3 Long-term Efficiency Performance

The effect of maximum window size ($w_{\max}$) on efficiency is studied in network environments with high and low transmission error rates (p), and small and large BDP. To obtain various values of the BDP, the link bandwidths are fixed and the channel delays are varied. The efficiency as a function of $w_{\max}$ for high error rate p = (0.15, 0.02) and for low error rate p = (0.0, 0.0) are compared in both large BDP l = (0.3s, 0.1s) and small BDP l = (0.01s, 0.01s) environments, as



(a) for *w* = 31

(b) for *w* = 18

(c) for *w* = 10

Figure 5.5. Premature timeout ratio

shown in Figure 5.6 and Figure 5.7, respectively. For high transmission error rate, it is shown that the efficiency remains almost unchanged for small BDP (Figure 5.7), whereas for large BDP (Figure 5.6), the efficiency increases and converges to the limiting point. To understand this trend, we further observe the packet losses due to vertical handover and transmission for high transmission error rate in small BDP and large BDP environments. It is found that as $w_{\max}$ increases, the proportion of total packets transmitted in WWAN to the packets transmitted in WLAN decreases for large BDP and remains fairly constant for small BDP. Since the transmission error rate is higher in WWAN than in WLAN, the total number of packet losses due to transmission error decreases for large BDP and remains almost unchanged for small BDP. For low transmission error rate, the efficiency seems to decrease as $w_{\max}$ increases for both large BDP (Figure 5.6) and small BDP (Figure 5.7) environments. The number of packet losses in the low transmission error rate environment is observed. It is found that the number of packet losses due to vertical handover significantly increases when $w_{\max}$ increases. This is due to the fact that the congestion window tends to grow to $w_{\max}$ in low transmission error rate environment and hence it closely depends on $w_{\max}$. Since the number of packet losses after the vertical handover is equal to the size of the congestion window, the increase of $w_{\max}$ increases the number of packet losses due to vertical handover and therefore decreases the efficiency.

5.3.4 Long-term Throughput Performance

The impact of maximum window size ($w_{\max}$) on the throughput is studied in the network environment with high and low transmission error rates and small and large BDP. The variations of throughput with $w_{\max}$ for small BDP l = (0.01s, 0.01s) and large BDP l = (0.3s, 0.1s) are compared in low transmission error rate p = (0.0, 0.0) (Figure 5.8) and high transmission error rate p

= (0.15, 0.02) (Figure 5.9) environments, respectively. It can be seen that, when p = (0.0, 0.0), the throughputs for large BDP converges to the throughput for low BDP (Figure 5.8), whereas when p = (0.15, 0.02), the throughput for large BDP converges to the point which is far below the throughput for low BDP (Figure 5.9). The reason is as follows. For low transmission error rates, a sufficiently large $w_{\max}$ allows the WP-TCP sender congestion window to grow to a large value and that enables the WP-TCP sender to fill in the "data pipe". This eliminates the unnecessary idle waiting time due to the channel delay. For high transmission error rate, the WP-TCP congestion control response limits the WP-TCP sender congestion window to a relatively small value regardless of the sized large $w_{\max}$. Consequently, the idle waiting time when l = (0.3s, 0.1s) is significantly larger than that when l = (0.01s, 0.01s).

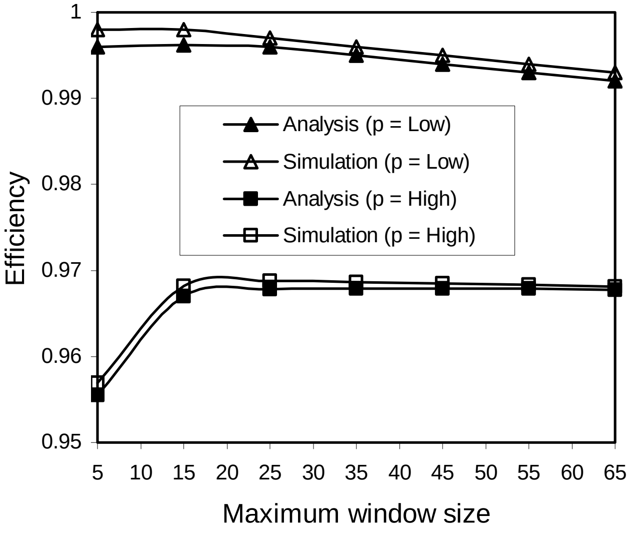


Figure 5.6. Efficiency for *l* = (0.3, 0.1) sec and *p* = (0.0, 0.0) and (0.15, 0.02)

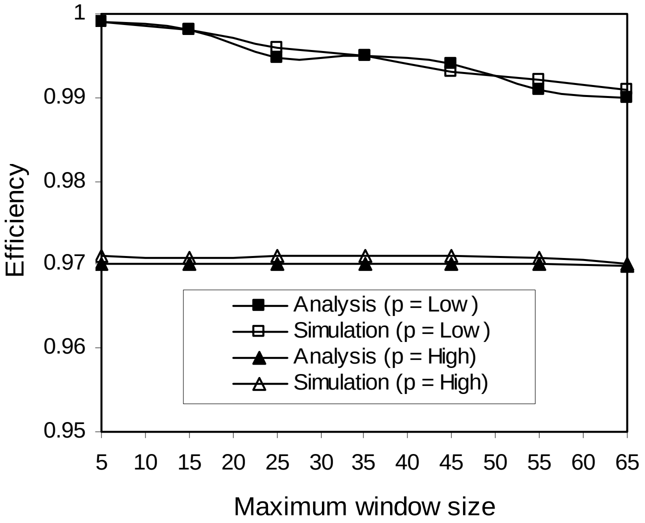


Figure 5.7. Efficiency for *l* = (0.01, 0.01) sec and *p* = (0.0, 0.0) and (0.15, 0.02)

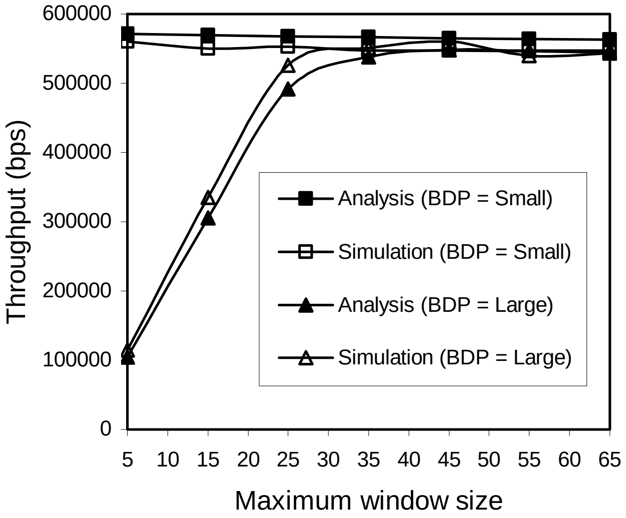


Figure 5.8. Throughput for *p* = (0.0, 0.0) and *l* = (0.01, 0.01) sec and (0.3, 0.1) sec

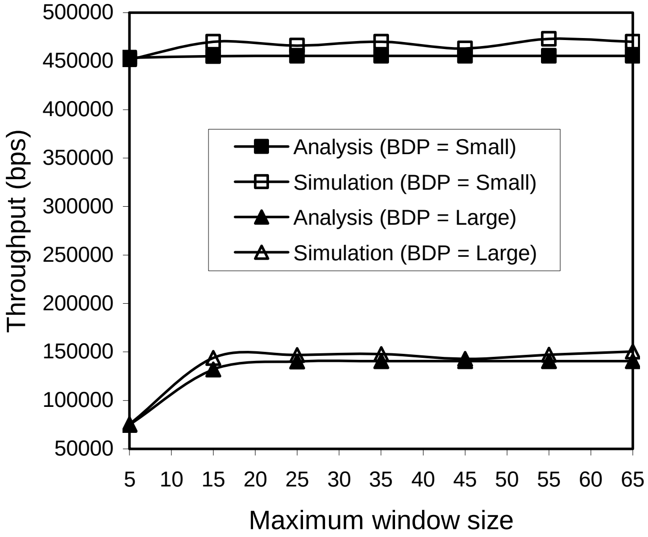


Figure 5.9. Throughput for *p* = (0.15, 0.02) and *l* = (0.01, 0.01) sec and (0.3, 0.1) sec

5.4 Summary

In this chapter, a novel analytical framework, which describes the short-term performance of WP-TCP during vertical handover and long-term performance of WP-TCP in integrated static WLAN and WWAN, is proposed. The framework captures the effects of vertical handover, such as excessive packet losses and sudden change in network characteristics, experienced in integrated static WLAN and WWAN. Simulation results have been given to demonstrate the accuracy of analytical results. It is shown that the approximations introduced in the framework have negligible effects on the accuracy of performance analysis.

Through the developed analytical framework, we have extensively analyzed the performance of WP-TCP long-lived flows. The following are the main findings. Firstly, when the network is subjected to hard handovers, increasing the maximum window size improves the efficiency in a high transmission error environment, but degrades the efficiency in a low transmission error environment. Secondly, increasing the congestion window reduces the chances of premature timeouts during soft upward vertical handover. Finally, depending on the value of the duplicate ACK threshold, increasing the congestion window can increase or reduce the chances of false fast retransmit during soft upward vertical handover

.

Chapter 6

Cross-layer Design and Analysis

6.1 Introduction

WAP 2.0 uses WP-TCP protocol as one of the reliable transport protocols to cope with the wireless network characteristics. However, when WP-TCP is deployed in integrated static WLAN and WWAN, its performance can degrade due to vertical handover, which can result in bandwidth-delay product mismatch, premature timeouts, inrush packet transmission, false fast retransmit, and burst packet losses as discussed in Subsection 2.3.3. In the literature, there are several schemes proposed to reduce the effects of burst packet losses due to network disconnections during hard vertical handover [74], [75], network congestion or underutilizations due to bandwidth-delay product mismatch [30], [28], [76] and inrush packet transmission [76], and premature timeouts and false fast retransmit due to temporary [31], [32], [33] or permanent [34], [35] step increase of round-trip time.

The schemes proposed in [31], [32], [33] detect falsely triggered fast retransmit or timeouts, and then undo the congestion control actions invoked. These schemes are proposed with the assumptions of homogeneous networks and they are reactive by nature. Recently, various proactive schemes that consider heterogeneous wireless networks have been proposed to specifically enhance the performance of TCP in soft vertical handover. In [34], the authors propose three network layer schemes (fast response, slow response, and ACK delay) to decrease the chance of premature timeouts in soft upward vertical handover. This is achieved by breaking a one-step increase of round-trip time into a two-step increase during vertical handover. The proposed schemes work well if the difference between round-trip times before and after the vertical handover is small. As the difference between round-trip times increases these schemes become ineffective. Nodupack TCP scheme has been proposed in [35] to improve the behavior of TCP during soft downward vertical handover by suppressing false fast retransmit. In this scheme, the TCP receiver needs to have knowledge of the TCP sender duplicate acknowledgement threshold parameter. Unfortunately, the implementation of this mechanism is not given and may need to be defined in the TCP option.

The fundamental cause of the performance degradation during vertical handover is primarily due to misinterpretation of apparent information such as packet losses, packet reordering, round-trip time increase at the WP-TCP sender. Therefore, we argue that an effective approach to completely address the performance degradation problem is to make information from some layers available to other layers and allow coordinated smart decisions to take place at each layer. This argument necessitates the use of the cross-layer coupling in improving the performance during vertical handover. But, on the other hand, a tight cross-layer coupling can lead to long-term design and performance problems. Therefore, it is important to carefully design and minimize the degree of dependence and exchanged information between non adjacent layers.

In this chapter, we consider integrated static WLAN and WWAN and address two WP-TCP performance barriers in vertical handover: premature timeout due to step increase of round-trip time and false fast retransmit due to packet reordering. First, we develop a mobile receiver centric loosely coupled cross-layer design, which is easy to implement and deploy, backward compatible with the standard TCP protocol, and robust in the absence of cross-layer information. Second, based on the developed design, we propose two proactive schemes which prevent false fast retransmit by equalizing the round-trip delay experienced by all packets and suppress premature timeouts by carefully inflating retransmission timeout time. Finally, we carry out extensive simulations and analyze the performance of the proposed schemes in downward and upward vertical handovers. It is demonstrated that our proposed schemes significantly improve the performance and outperform other schemes in a wide range of network conditions.

The rest of this chapter is organized as follows. Section 6.2 presents the developed cross-layer design and the proposed proactive schemes for vertical handover. Section 6.3 analyzes the proposed proactive schemes in ideal network environment. Simulation study to demonstrate the performance of the proposed schemes in non-ideal network environment is presented in Section 6.4. The salient aspects of cross-layer design presented in this chapter are summarized in Section 6.5.

6.2 Cross-layer Design

Our analytical study on WP-TCP long-lived flows in Chapter 5 suggest that the false fast retransmits induced by packet reordering and premature timeouts due round-trip time increase can be suppressed by jointly considering some parameters in the data link, network, and transport layers. Consequently, we use a cross-layer coupling [77] approach in improving the vertical handover performance. But, on the other hand, a tight cross-layer coupling can lead to long-term design and performance problems [78]. Therefore, we consider a loosely coupled cross-layer design where non-adjacent layers can only interact indirectly through the enhancement module. Since it is easy to collect the information about the vertical handover and the status of the wireless channels at the mobile node, we further consider a mobile receiver centric design to simplify the implementation. In the next subsections, we present the protocol architecture of the proposed mobile receiver centric loosely coupled cross-layer design and develop proactive schemes for vertical handover.

6.2.1 Protocol Architecture

Figure 6.1 shows the protocol architecture of the proposed cross-layer design. The architecture has two main components: communication protocol stack and enhancement module. The layers involved in the communication protocol stack include the transport layer (wireless profiled TCP), the network layer (mobile IP and Internet Control Message Protocol (ICMP)), and the data link layer (WLAN and WWAN). Direct inter-layer communication is restricted to adjacent layers (vertical arrows). The enhancement module handles critical protocol configurations and runs schemes to enhance the overall performance. It can query or change state variables of the communication protocols (horizontal arrows). Table 6.1 summaries the information and protocols parameters needed to be accessed and controlled by the enhancement module. More details on accessed information and controlled protocol parameters are given in the the next subsection.

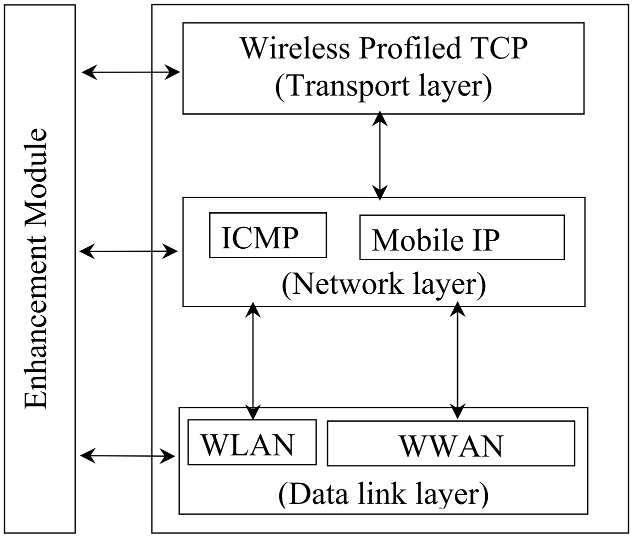


Figure 6.1. Protocol architecture of the proposed cross-layer design

Table 6.1. Cross-layer information and parameters

Layer	Information available	Controlled parameter
Transport	Packet header information	Packet header parameters
Network	Handover events	RTT measurement through ICMP
Data Link	Link status and bandwidth	Active network interfaces

6.2.2 Vertical Handover Schemes

In this subsection, we develop two proactive schemes which are used by the enhancement module to improve the performance in upward and downward vertical handovers. We call the developed schemes for upward and downward vertical handovers *RTT Inflation* and *RTT Equalization*, respectively.

6.2.2.1 RTT Inflation Scheme

We address the problem of premature timeout in upward vertical handover by carefully inflating the round-trip time samples measured at the WP-TCP sender. We achieve this by modifying the standard procedure for round-trip time measurement at the WP-TCP receiver (described in Subsection 2.3.2). The proposed procedure at the WP-TCP receiver is shown in Figure 6.2. Note that we introduce new variables $TS.correction$ and $TO.initial$ at the WP-TCP receiver. The variable $TS.correction$ tracks elapsed time after a new in-order data packet is received. The amount of elapsed time is used to correct the round-trip time samples measured at the WP-TCP sender from being overestimated. The variable $TO.initial$ stores the amount of time needed to be inflated in the round-trip time. On the conservative side, the inflated time is kept slightly higher than the estimated round-trip time in the WWAN. The estimated round-trip time in the WWAN can be computed as $icmp.RTT + MSS/b_{BS}$, where $icmp.RTT$ is the round-trip time found by using ICMP messaging, MSS is maximum segment size (data packet length), and b_{BS} is data transmission rate of the base station in the WWAN. If the ICMP messaging method fails to estimate the round-trip time before the upward vertical handover is triggered, the $TO.initial$ is set to the value of the WP-TCP sender initial retransmission timeout. This value is usually between 2.5 to 3 seconds. The proposed *RTT Inflation* scheme at the WP-TCP receiver for upward vertical handover is summarized as follows. If hard upward vertical handover is expected (can be predicted using link status), the standard procedure at the WP-TCP receiver is followed and all generated acknowledgment packets are transmitted through the WWAN interface. Otherwise, if hard upward vertical handover is not expected the following steps are used:

a) Just before initializing the handover, estimate the round-trip time in the WWAN (Figure 6.2, STEP #1). After initializing the handover, generate a duplicate acknowledgement packet with inflated round-trip time (Figure 6.2, STEP #3) and transmit it through the WLAN interface. Also, generate a duplicate acknowledgement packet (Figure 6.2, STEP #4) and transmit it through the WWAN interface.

b) When a new out-of-order data packet is received from the WLAN interface, generate the corresponding duplicate acknowledgement packet with inflated round-trip time (Figure 6.2, STEP #3) and transmit it through the WLAN interface.

c) When a new in-order data packet is received from the WLAN interface, update the timer variables (Figure 6.2, STEP #2). Generate a corresponding acknowledgement packet with inflated round-trip time (Figure 6.2, STEP #3) and transmit it through the WLAN interface.

d) When a new out-of-order data packet is received from the WWAN, generate a corresponding duplicate acknowledgement packet (Figure 6.2, STEP #4) and transmit it through the WWAN interface.

e) If a new in-order data packet is received from the WWAN interface, it implies that the vertical handover is completed. Generate a duplicate acknowledgement packet with half of inflated round-trip time (Figure 6.2, STEP #3) and transmit it through the WLAN interface. Furthermore, generate a duplicate acknowledgement packet (Figure 6.2, STEP #4) and transmit it through the WWAN interface. Deactivate the *RTT Inflation* scheme and follow the standard procedure at the WP-TCP receiver.

```
STEP #1
If icmp rtt estimation is successful {
   $TO.initial = icmp.RTT + MSS/b_{BS}$
} else {
   $TO.initial = default(3\,\text{sec})$
}
STEP #2
If receive new in-order data packet {
   $TS.recent = data.TSval$
   $TS.correction = 0$
}
STEP #3
If send acknoledgment packet with inflated round-trip time {
   $ack.TSecr = TS.recent + TS.correction - TO.initial$
}
STEP #4
If send acknoledgment packet with real round-trip time{
   $ack.TSecr = TS.recent$
}
```

Figure 6.2. Modified round-trip time measurement procedure at the WP-TCP receiver

6.2.2.2 RTT Equalization Scheme

We prevent false fast retransmit due to packet reordering during downward vertical handover by equalizing the round-trip delay experienced by all packets. The round-trip delay constitutes the packet transmission delay and propagation delay along the network path. We equalize the propagation delays of all packets by sending the acknowledgement packets correspond to data packets received from the WWAN and WLAN through the WLAN and WWAN, respectively.

We equalize the transmission delays for all packets by delaying the acknowledgement packets sent through the WWAN with approximately WWAN data packet transmission time *T*. If the WWAN data transmission rate is not available on time, the equalization of transmission delays can be ignored. Therefore *T* can be computed as

$$T = \begin{cases} MSS / b_{BS}, \text{ if } b_{BS} \text{ is available} \\ 0, \text{ otherwise,} \end{cases} \tag{6.1}$$

where *MSS* is maximum segment size (data packet length) and b_{BS} is data transmission rate of the base station in the WWAN. The proposed *RTT Equalization* scheme at the WP-TCP receiver for downward vertical handover is summarized as follows:

a) When a new out-of-order data packet is received from the WWAN interface, transmit the duplicate acknowledgement packet through the WLAN interface.

b) When a new out-of-order data packet is received from the WLAN interface, wait for the time *T* computed in (6.1) and transmit the duplicate acknowledgment packet through the WWAN interface.

c) When a new in-order data packet is received from the WWAN interface, transmit the corresponding acknowledgement packets through the WLAN interface. And then cancel the transmission of all delayed (outstanding) duplicate acknowledgment packets with lower acknowledgement numbers in step (*b*).

d) If a new in-order data packet is received from the WLAN interface, it implies the vertical handover is completed. Transmit the corresponding acknowledgement packet through the

WLAN interface. Deactivate the *RTT Equalization* scheme and follow the standard procedure at the WP-TCP receiver.

6.3 Performance Analysis

In this section, we analyze the *RTT Inflation* and *RTT Equalization* schemes in an ideal network environment where a single pair WP-TCP sender-receiver runs over integrated WLAN and WWAN without packet losses. Our main objective here is to analyze the behaviors of the *RTT Inflation* and *RTT Equalization* schemes under the influence of sudden change in network characteristics due to upward and downward vertical handovers, respectively. Let consider WP-TCP sender duplicate acknowledgement threshold (*Z*) and congestion window (*w)* measured in packets, size of data packet (*MSS*) and acknowledgement packet (*MAS*) measured in bits, one-way channel delay of WWAN (l_{BS}) and WLAN (l_{AP}) measured in seconds, bandwidth in WWAN (b_{BS}) and WLAN (b_{AP}) measured in bits per second, and bandwidth-delay product (BDP) of the network in use before the vertical handover measured in packets and computed as

$$BDP = 2l_i b_i / MSS + 1, \tag{6.2}$$

for all $i \in \{BS, AP\}$.

6.3.1 Upward Vertical Handover

Consider the *RTT Inflation* scheme proposed to suppress premature timeouts due to upward vertical handover in the Subsection 6.2.2.1. Let t'_{now} and t_{now} respectively denote the timestamp time in the last data packet received before step (*a*) is executed and the time when the duplicate acknowledgement packet sent through WLANs in step (*a*) is received at the WP-TCP sender. From Subsection 2.3.2, the sample RTT at the WP-TCP sender is computed as

$$RTT = t_{now} - TSecr. \tag{6.3}$$

By definition, t_{now} can be written as

$$t_{now} = t'_{now} + RTT_{WLAN} + TS.correction, \tag{6.4}$$

where RTT_{WLAN} is the round-trip time in the WLAN. From step (*a*) and STEP #3 in Figure 6.2, $TSecr$ can be computed as

$$TSecr = TS.recent + TS.correction - TO.initial. \tag{6.5}$$

From RTT measurement procedure at the WP-TCP sender (Subsection 2.3.2) and the WP-TCP receiver (Figure 6.2, STEP #2), $data.TSval = t'_{now}$ and $TS.recent = data.TSval.$ Consequently, from (6.3), (6.4), and (6.5), the first sample RTT measured after vertical handover can be found as

$$RTT = RTT_{WLAN} + TO.initial. \tag{6.6}$$

Just before upward vertical handover WLAN is considered to be in a steady state. Therefore, the smoothed round-trip time and round-trip time variation are given as $SRTT = RTT_{WLAN}$ and $RTTVAR = 0,$ respectively. Let RTO_i be the retransmission timeout computed after the i^{th} received inflated sample RTT. From (2.1)-(2.3), the retransmission timeout computed after the first inflated sample RTT, which is obtained from the duplicate acknowledgement packet (see step (a)), can be found as

$$RTO_1 = RTT_{WLAN} + 1.1250 \cdot TO.initial. \tag{6.7}$$

By using the same approach as above, the retransmission timeout computed after the second inflated sample RTT, which is obtained from the new acknowledgement packet (see step (*c*)), can be found as

$$RTO_2 = RTT_{WLAN} + 1.8594 \cdot TO.initial. \tag{6.8}$$

All subsequent retransmission timeouts, which are computed from inflated sample RTTs, are at least equal to inflated sample RTTs. Therefore,

$$RTO \geq RTT_{WLAN} + TO.initial, \tag{6.9}$$

Since the *TO.initial* is kept slightly higher than the estimated round-trip time in the WWAN (RTT_{WWAN}) in the *RTT Inflation* scheme, (6.9) can be rewritten as

$$RTO \geq RTT_{WLAN} + RTT_{WWAN}, \tag{6.10}$$

From (6.10) and the discussion presented in Subsection 5.2.4.2, it is evident that the premature timeouts can not occur when *w* is greater than half of BDP. When *w* is equal to or less than half of BDP, premature timeouts can rarely occur and only in extreme network setups.

6.3.2 Downward Vertical Handover

Consider the *RTT Equalization* scheme proposed to prevent false fast retransmit due to downward vertical handover in the Subsection 6.2.2.2. To analyze the *RTT Equalization* scheme, we consider two cases: i) when *w* is greater than BDP (i.e., $w > 2l_{BS}b_{BS} / MSS + 1$) and ii) when *w* is equal to or less than BDP (i.e., $w \leq 2l_{BS}b_{BS} / MSS + 1$).

Case I: When $w > 2l_{BS}b_{BS} / MSS + 1$. In this case, the false fast retransmit due to downward vertical handover can occur if the following two conditions are satisfied. The first condition is that, a burst of more than Z data packets must be injected in the WLAN. But the burst of data packets can only be created if the first acknowledgement packet sent through the WLAN interface overtakes the in-flight acknowledgement packets sent through the WWAN interface before the downward vertical handover is initiated. Therefore, to satisfy this condition, the inequality

$$l_{BS} - (MAS / b_{AP} + l_{AP}) \geq Z \cdot MSS / b_{BS} \tag{6.11}$$

must be valid. The second condition is that, at least Z duplicate acknowledgement packets transmitted in step (*b*) must arrive at the WP-TCP sender before the next new acknowledgement packet in step (*c*). Therefore, to satisfy this condition, the inequality

$$MSS / b_{BS} + (MAS / b_{AP} + l_{AP}) \geq Z \cdot MSS / b_{AP} + T + (MAS / b_{BS} + l_{BS}) \tag{6.12}$$

must be valid. If the first and second conditions are satisfied, the inequality

$$MSS / b_{BS} \geq Z \cdot MSS / b_{BS} + Z \cdot MSS / b_{AP} + T + MAS / b_{BS} \tag{6.13}$$

obtained by adding (6.11) and (6.12) must also be valid. Note that in (6.13) $Z \geq 1$ and the other variables are non-negative. Therefore, the inequality (6.13) is always not valid. This implies that the first and second conditions can not be satisfied simultaneously and therefore the false fast retransmit can not occur.

Case II: When $w \leq 2l_{BS}b_{BS} / MSS + 1$. In this case, the false fast retransmit due to downward vertical handover can occur either as described in Case I above or when the following two conditions are satisfied. The first condition is that at least a block of Z data packets sent through the WLAN interface overtakes at least one in-flight data packet sent through the WWAN during downward vertical handover. Therefore, to satisfy this condition, the inequality

$$l_{BS} + MSS / b_{BS} \geq l_{AP} + Z \cdot MSS / b_{BS} \qquad (6.14)$$

must be valid. Note that even though the transmission time of a data packet in the WLAN is MSS / b_{AP}, the inter-departure time of data packets transmitted in the WLAN is equal to the inter-arrival time of acknowledgement packets in the WWAN at the WP-TCP sender, which is MSS / b_{BS} (i.e., *TCP ACK-clocking* in the steady-state conditions). The second condition is that at least Z duplicate acknowledgement packets in step (*b*) arrive at the WP-TCP sender before the next new acknowledgement packet in step (*c*). Therefore, to satisfy this condition, the inequality

$$ACK / b_{AP} + l_{AP} \geq T + (MAS / b_{BS} + l_{BS}) \qquad (6.15)$$

must be valid. If the first and second conditions are satisfied, the inequality

$$MSS / b_{BS} + MAS / b_{AP} \geq Z \cdot MSS / b_{BS} + T + MAS / b_{BS} \qquad (6.16)$$

obtained by adding (6.14) and (6.15) must also be valid. Note that $b_{AP} \geq b_{BS}$, $Z \geq 1$, and the other variables are non-negative. Therefore, the inequality (6.16) is always not valid. This implies that the first and second conditions can not be satisfied simultaneously and therefore the false fast retransmit can not occur.

From Case I and Case II above, it is clear that false fast retransmit can not occur in an ideal network environment, even with the worst case configuration (i.e., $Z = 1$ and $T = 0$). Note that in the

RTT Inflation scheme, T is recommended to be set to WWAN data packet transmission time ($T = MSS / b_{BS}$) in order to reduce the number of unnecessary transmitted duplicate acknowledgement packets and further avoid false fast retransmit in a real network environment.

6.4 Performance Evaluation

We evaluate the performance by using the ns-2 simulator [69]. The simulation topology is shown in Figure 6.3. AP, BS, and CR represent an access point in the WLAN, a base station in the WWAN, and a common router, respectively. A single, long-lived WP-TCP flow is configured to run over the integrated static WLAN and WWAN, where the WP-TCP sender is the correspondent node (CN) and the WP-TCP receiver is the mobile node (MN). WP-TCP is simulated as described in Section 2.3. The independent duplex wireline links BS-CR and AP-CR are set with large bandwidths and small delays whereas the common duplex wireline link CR-CN is set with large bandwidth and variable delay. A packet traversing this link experiences a delay which is uniformly distributed over the interval $[l - 0.5\Delta l, l + 0.5\Delta l]$, where l is the one-way link delay and Δl is the maximum delay deviation. The duplex wireless links MN-AP and MN-BS are set with small bandwidths, link delays (l_{AP} and l_{BS}) and packet loss rates (e_{AP} and e_{BS}).

Each simulated data point is obtained as an average of 300 simulation runs repeated with different random seeds. If not stated otherwise, the simulation parameters are given as maximum delay deviation $\Delta l = 0.04$ sec, one-way wireline link delay $l = 0.1$ sec, WWAN one-way delay $l_{\mathrm{BS}} = 1.0$ sec, WLAN one-way delay $l_{\mathrm{AP}} = 0.1$ sec, WWAN packet loss rate $e_{\mathrm{BS}} = 0.00$, WLAN packet loss rate $e_{\mathrm{AP}} = 0.00$, WP-TCP maximum window size = 37 packets, WP-TCP duplicate ACK threshold = 3 packets, minimum retransmission timeout = 0.2 sec, WP-TCP data packet size (maximum segment size (*MSS*)) = 1KB, and WP-TCP acknowledgement packet size (maximum acknowledgement size (*MAS*)) = 0.05KB. Other parameters are given in Figure 6.3.

The metric in evaluating the performance of the proposed schemes is the number of in-order data packets received in 4 seconds after downward vertical handover is initiated and 40 seconds after upward vertical handover is initiated. To facilitate the analysis, we also monitor the occurrence of false fast retransmit in downward vertical handover and premature timeouts in upward vertical handover. From (5.19), it is indicated that after a single timeout the slow start threshold is set to half of the current congestion window and the congestion window is set to one whereas after two or more consecutive timeouts slow start threshold is set to two (the minimum value) and the congestion window is set to one. Consequently, in this study we only monitor single premature timeouts and two consecutive premature timeouts. To appreciate the performance gained by the proposed schemes, we also present the performance of three schemes which are commonly used during vertical handover in multi-radio receivers. For simplicity, we refer to these schemes as *Bicast*, *Same Interface*, and *New Interface*. In the *Bicast* scheme, each acknowledgement packet is sent through both the WLAN interface and the WWAN interface. In the *Same Interface* scheme, each acknowledgement packet is sent through the network interface from which the corresponding data packet is received [76], and in the *New Interface* scheme, acknowledgement packets are always sent through a new network interface [35].

It is important to note that overheads and implementation complexities associated with the *RTT Inflation* or *RTT Equalization* schemes are fairly low. This is due to fact that the schemes only require a small memory space and involve light tasks such as choosing interface, modifying head information, and add/subtract/compare operations. Furthermore, functions needed in measuring RTT and detecting vertical handovers are readily available as standard APIs (application programming interface).

In the next subsections, we analyze the impact of WP-TCP protocol parameters, network conditions, and imperfect cross-layer information. In Figure 6.4 – 6.11, it is interesting to note that *Bicast* and *Same Interface* schemes perform almost the same in upward vertical handover whereas *Bicast* and *New Interface* schemes performance almost the same in downward vertical handover. In *Bicast* scheme, acknowledgment packet sent through WWAN is always ignored because it takes longer time to arrive at the WP-TCP than the duplicate copy sent through WLAN. Hence, in upward vertical handover the *Bicast* and *Same Interface* schemes have same effect to the WP-TCP sender most of times. Similarly, during downward vertical handover the *Bicast* and *New Interface* schemes have same effect to the WP-TCP sender most of times.

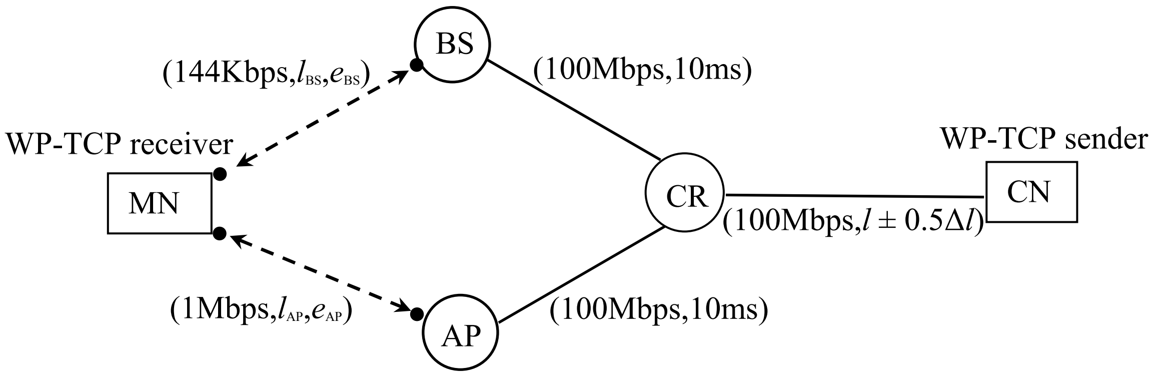

BS
(144Kbps,l_{BS},e_{BS})
(100Mbps,10ms)
WP-TCP receiver
WP-TCP sender
MN
CR
CN
(100Mbps,$l \pm 0.5\Delta l$)
(1Mbps,l_{AP},e_{AP})
(100Mbps,10ms)
AP

Figure 6.3. Simulation topology

6.4.1 Effects of WP-TCP Parameters

6.4.1.1 WP-TCP duplicate ACK threshold

To demonstrate the effectiveness of the *RTT Equalization* scheme, we observe the number of in-order data packets received in a specified time interval for various values of WP-TCP duplicate ACK threshold. Figure 6.4a shows the number of in-order data packets received in 4 seconds. It can be seen that when duplicate ACK threshold is set to 4 packets, the number of in-order data packets received in the *RTT Equalization* scheme is nearly 45% more than those in the other schemes. This performance gain is due to the fact that the *RTT Equalization* scheme significantly prevents false fast retransmit, which is triggered because of packet reordering during downward vertical handover, by equalizing the round-trip delay experienced by all packets. This explanation is supported with the results in Figure 6.4b, where the percentage of false fast retransmit in 4 seconds is presented for various values of WP-TCP duplicate ACK threshold. It can be seen that in the *RTT Equalization* scheme the duplicate ACK threshold only needs to be set to 4 packets to completely prevent false fast retransmit whereas in other schemes the duplicate ACK threshold needs to be set to more than 10 packets. In other words, the *RTT Equalization* scheme has on average suppressed more than 6 unnecessary duplicate acknowledgements packets which would otherwise trigger the false fast retransmit.

6.4.1.2 WP-TCP congestion window

In Section 6.3, it is indicated that the size of congestion window affects the vertical handover performance of the proposed schemes in an ideal network environment. In order to understand the effects of congestion window on the proposed schemes in more realistic network environments, we monitor its size just before vertical handover is initialized and investigate the performance. The effects of congestion window in upward and downward vertical handover are respectively illustrated in Figure 6.5 and Figure 6.6. From Figure 6.5a and Figure 6.6a, the performance of all schemes improves as congestion window increases but improvement in *RTT inflation* and *RTT Equalization* schemes is much more than those in other schemes. It can be seen that for all schemes the percentage of premature timeouts (Figure 6.5b and 6.5c) remain fairly unchanged while the percentage of false fast retransmit (Figure 6.6b) increases significantly as congestion window increases. However, the proposed *RTT Inflation* and *RTT Equalization* schemes respectively have lower percentage of premature timeouts and false fast retransmits as compared to other schemes. Since premature timeouts and false fast retransmits invoke WP-TCP congestion control that subsequently reduce the congestions window, the performance of the proposed schemes improves more than other scheme as the congestion window increases.

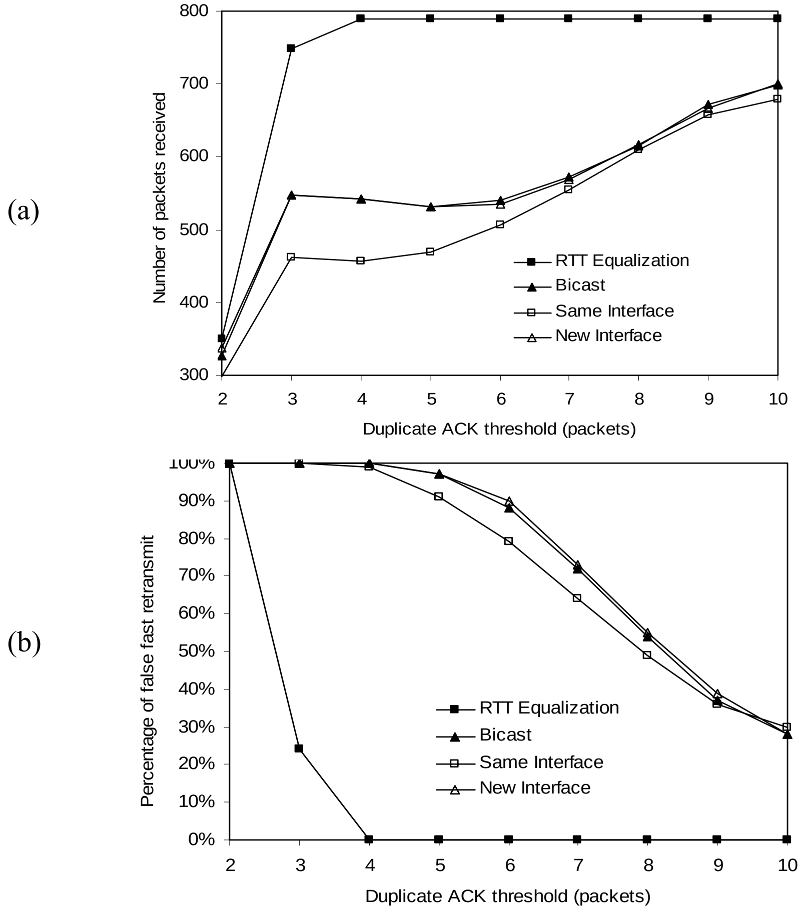


Figure 6.4. Impact of duplicate ACK threshold in downward vertical handover

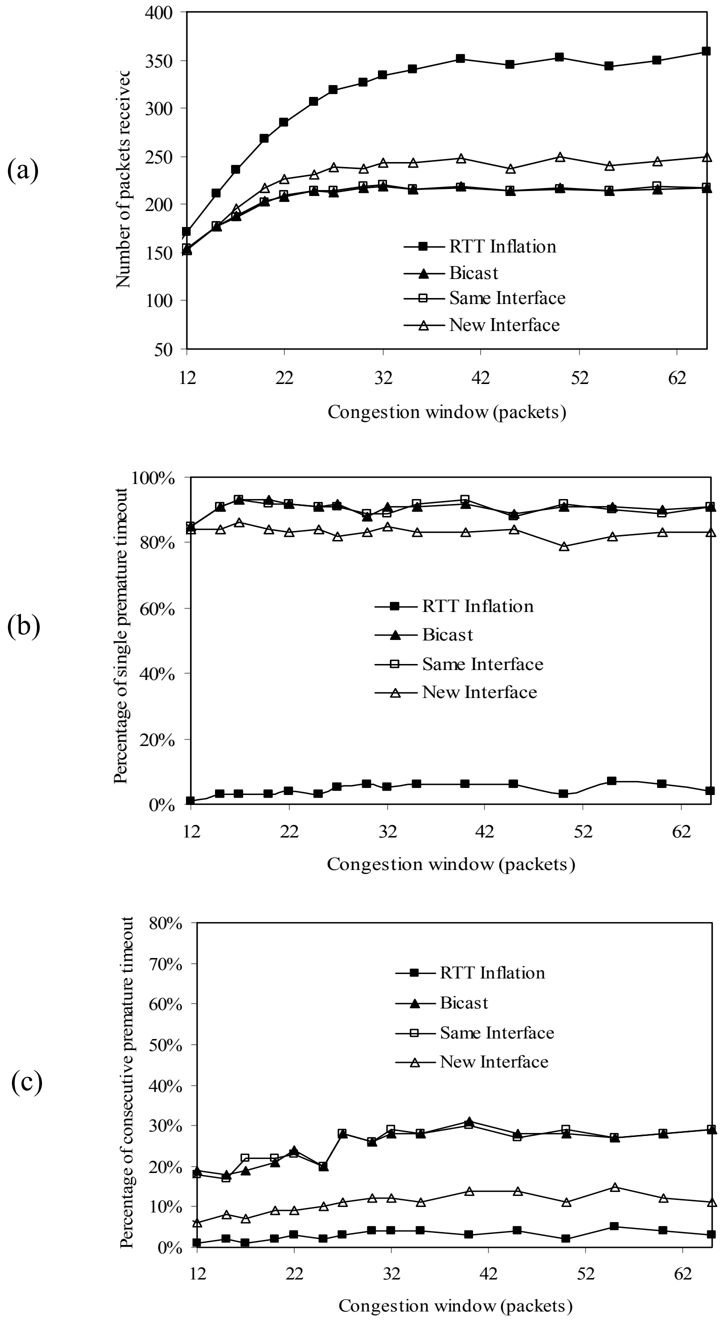


Figure 6.5. Impact of congestion window in upward vertical handover

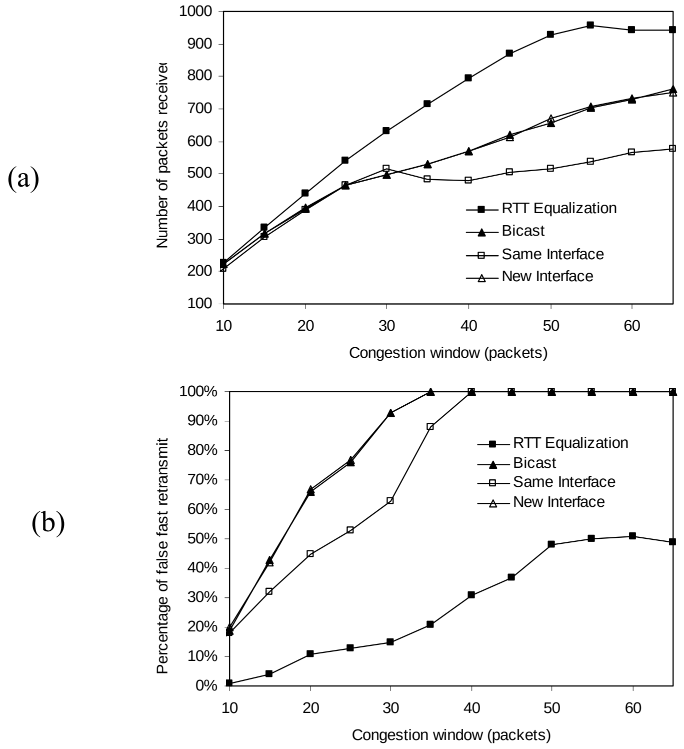


Figure 6.6. Impact of congestion window in downward vertical handover

6.4.2 Effects of Network Conditions

6.4.2.1 Impact of network delay

The effects of the WWAN one-way delay in upward vertical handover and the WLAN one-way delay in downward vertical handover are respectively shown in Figure 6.7 and Figure 6.8. Due to an increase in round-trip time, the performance of all schemes in vertical handover degrades as the WWAN and WLAN one-way delays increase. In upward vertical handover the performance of the *RTT Inflation* scheme degrades much slower than other schemes (Figure 6.7a).

In downward vertical handover the performance of the *RTT Equalization* scheme degrades much faster than other schemes (Figure 6.8a). These observations can be explained as follows. The WWAN one-way delay is always greater than or equal to the WLAN one-way delay. Therefore, the increase of the WWAN one-way delay or decrease of the WLAN one-way delay increases a step change of round-trip time which is experienced in vertical handover. The step change of round-trip time significantly affect the percentage of premature timeouts (Figure 6.7b and 6.7c) and false fast retransmits (Figure 6.8b) in other schemes but not in the *RTT Inflation* scheme and *RTT Equalization* scheme, respectively. Note that the premature timeouts and false fast retransmits observed in *RTT Inflation* and *RTT Equalization* respectively are mainly due to delay variation (Δl) set in the common duplex wireline link CR-CN.

6.4.2.2 Impact of network delay variation

Figure 6.9 and Figure 6.10 show the effects of the delay variation (Δl) in upward and down ward vertical handovers, respectively. In upward vertical handover, the increase of delay variation causes dual effects as shown in Figure 6.9b and 6.9c. If the percentage of premature timeout is already high, it slightly reduces the occurrence of premature timeouts by inflating retransmission timeout time. If the percentage of premature timeout is low, it slightly increases the occurrence of premature timeouts by inducing delay spikes. Since delay variation significantly affect the occurrence of false fast retransmit (Figure 6.10b), and hence a dominant factor, the performance of all schemes degrades as delay variation increases (Figure 6.9a). In downward vertical handover, the percentage of false fast retransmit is significantly affected by the delay variation (Figure 6.10b). Consequently, the performance in all schemes severely degrades as the delay variation increases (Figure 6.10a).

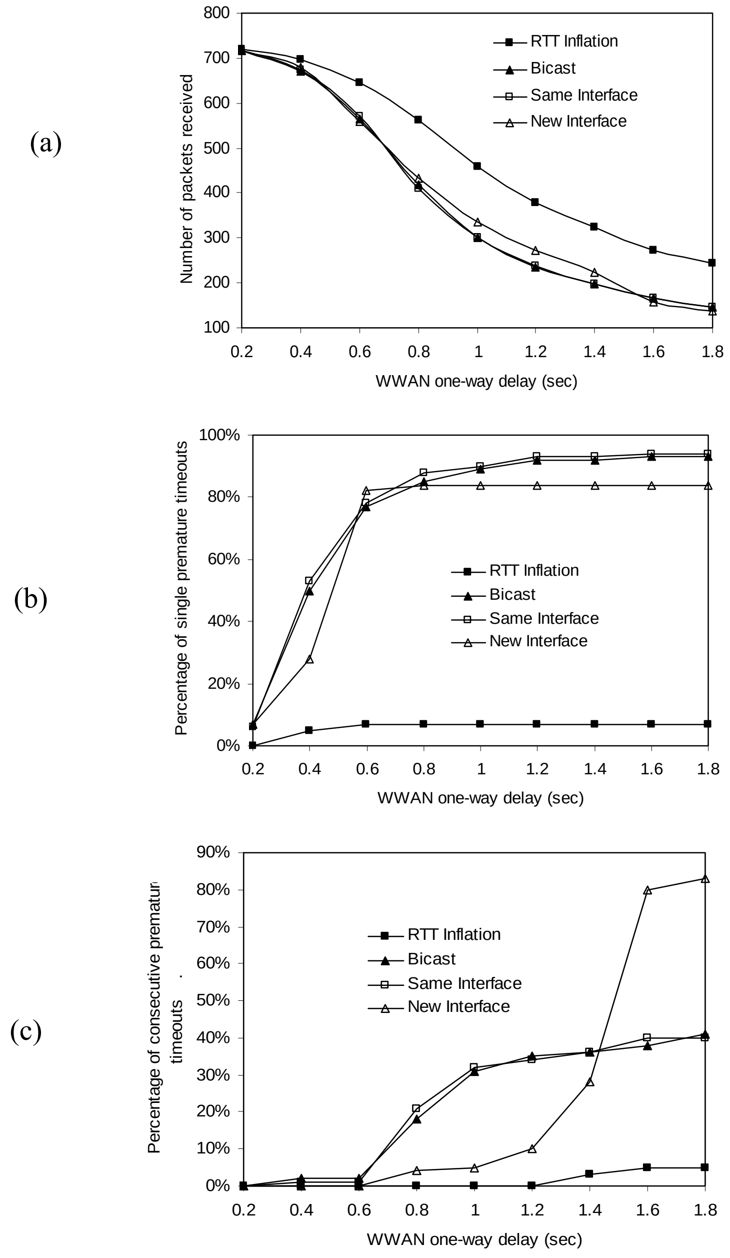


Figure 6.7. Impact of WWAN one-way delay in upward vertical handover

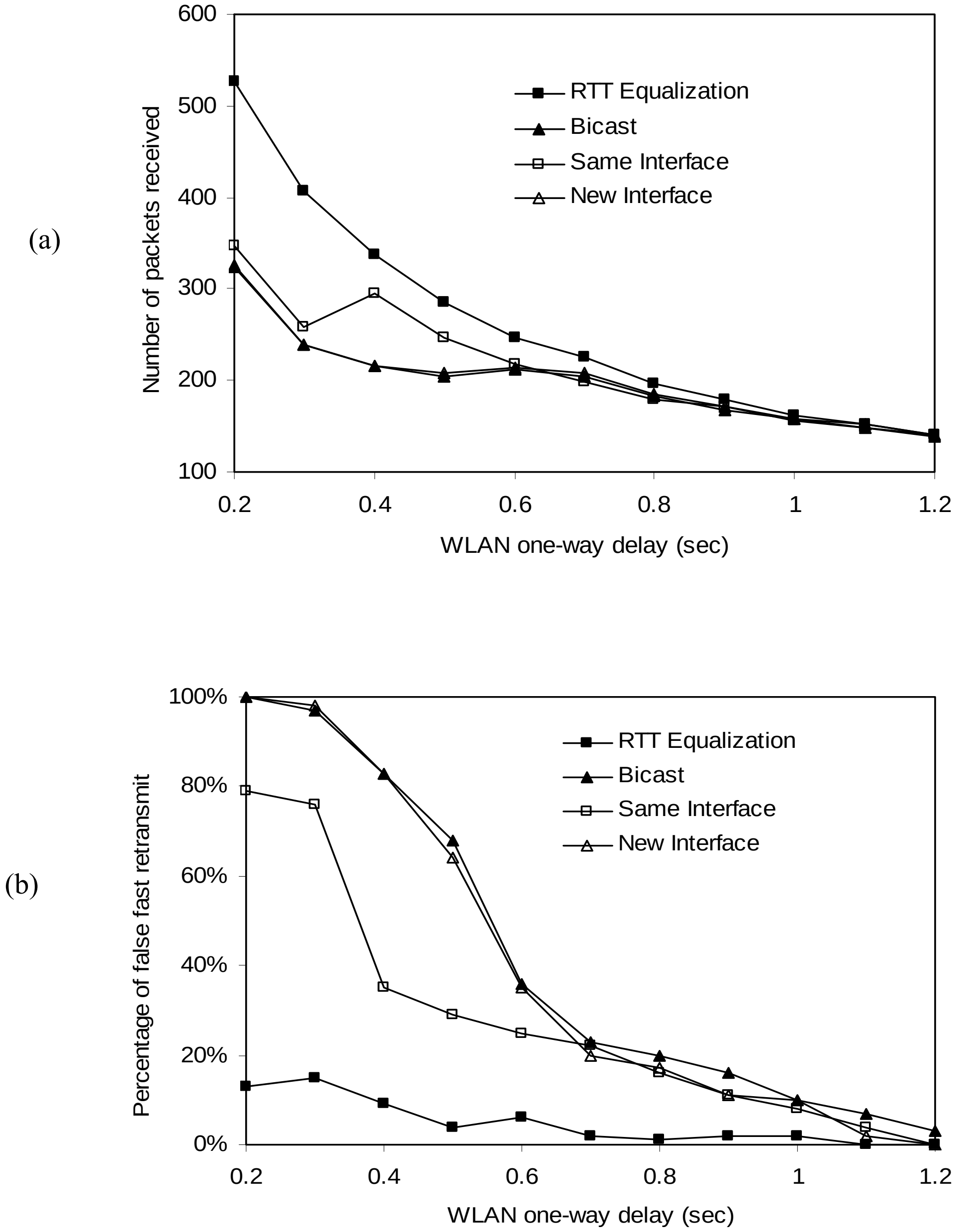


Figure 6.8. Impact of WLAN one-way delay in downward vertical handover

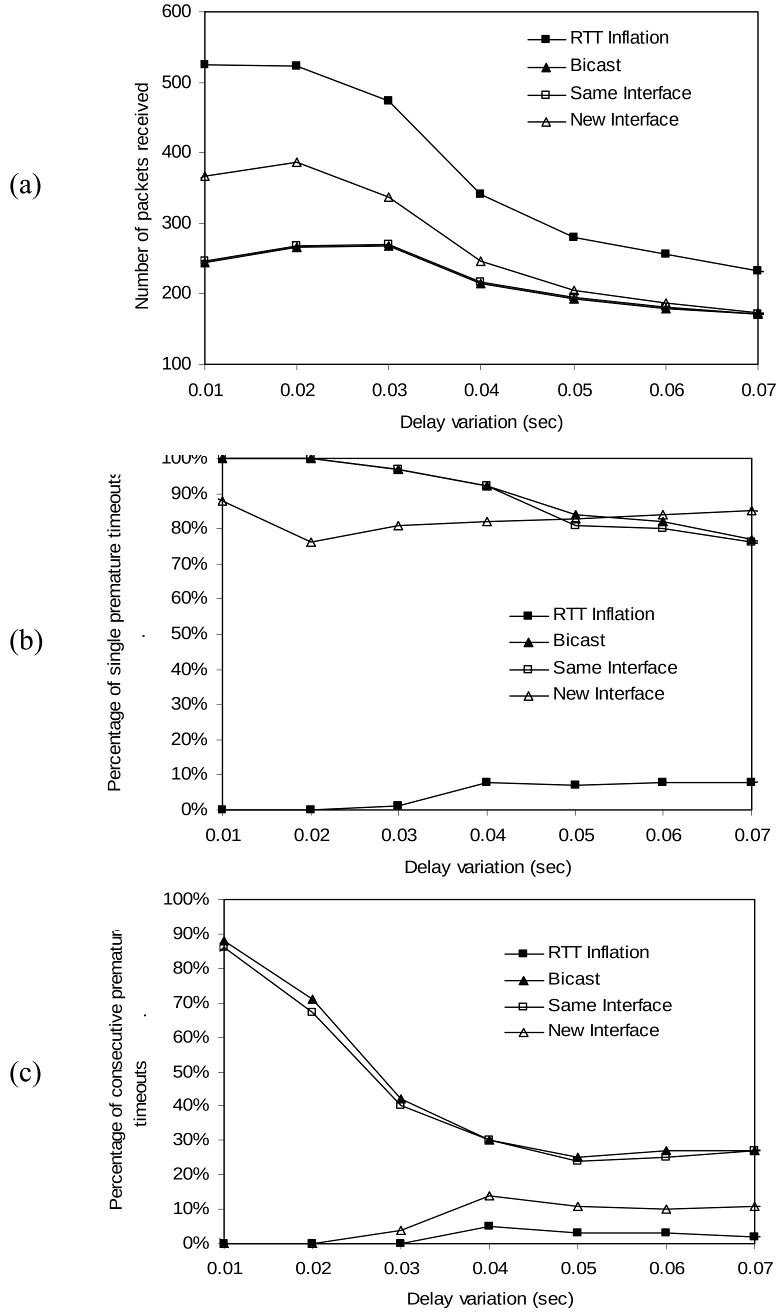


Figure 6.9. Impact of delay variation in upward vertical handover

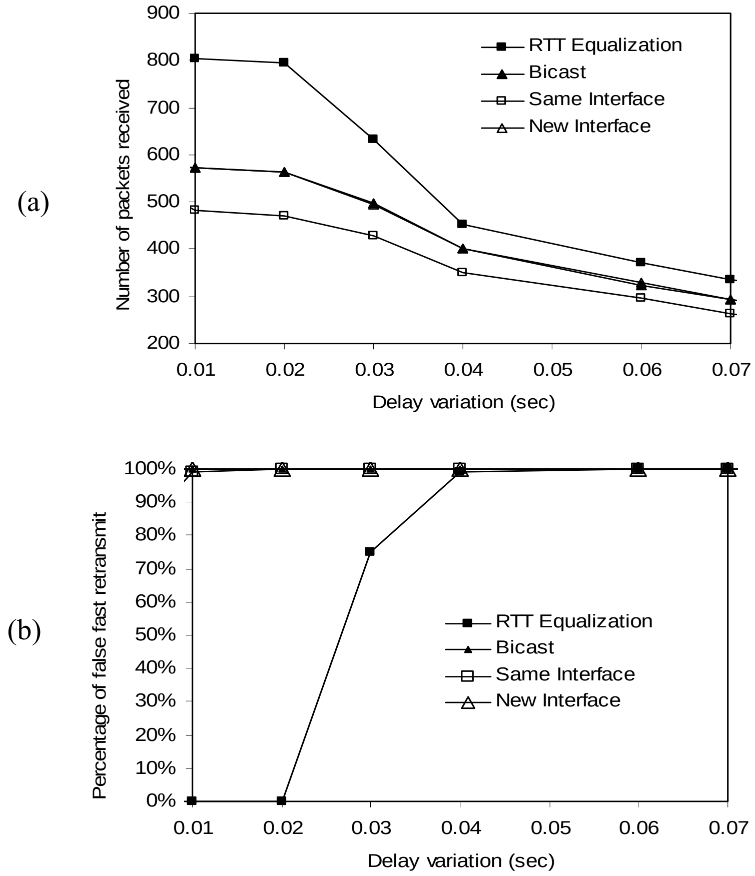


Figure 6.10. Impact of delay variation in downward vertical handover

6.4.2.3 Impact of packet loss rate

The *RTT Inflation* scheme suppresses premature timeouts in upward vertical handover by inflating retransmission timeout time at the WP-TCP sender. On the other hand, the retransmission timeout time is used to determine the moment to start recovering lost packets. Therefore, a long retransmission timeout time can negatively effect the recovery of lost packets and so the performance. To study this tradeoff in the *RTT Inflation* scheme, we allow packet losses in the downlink of the WWAN and observe the timeouts. Note that in this case, the observed timeouts include desirable and premature (undesirable) timeouts. In Figure 6.11a, it is clear that in the absence of packet losses the *RTT Inflation* scheme outperforms other schemes. However, as the packet loss rate increases, the performance of the *RTT Inflation* scheme degrades significantly. In Figure 6.11b and 6.11c, the *RTT Inflation* scheme shows a significant increase of percentage of single timeout (to nearly 70%) and an insignificant increase of percentage of consecutive timeouts (to nearly 8%) as packet loss rate increases. These results confirm that the timeout mechanism is used to recover lost packet and therefore the performance of the *RTT Inflation* scheme is affected in two ways: 1) because of inflated retransmission timeout time, the WP-TCP sender takes longer to respond to packet losses and 2) because of the WP-TCP congestion control response, the congestion window (packet sending rate) is reduced. Despite of all these effects, the *RTT Inflation* scheme still exhibit comparable or better performance than other schemes.

6.4.3 Effects of Imperfect Cross-layer Information

The *RTT Inflation* and *RTT Equalization* schemes utilize the cross-layer information in improving the vertical handover performance. In the previous subsections we assume that the cross-layer information is perfect. However, this is not always the case in the real network environment. Therefore, in this subsection, we study the impact of imperfect cross-layer information on the performance of the *RTT Inflation* and *RTT Equalization* schemes. We classify the cross-layer information imperfection into two categories: imperfection due to information failure (i.e., the requested information is not available on time) and imperfection due to information error (i.e., the requested information is available on time but is not accurate).

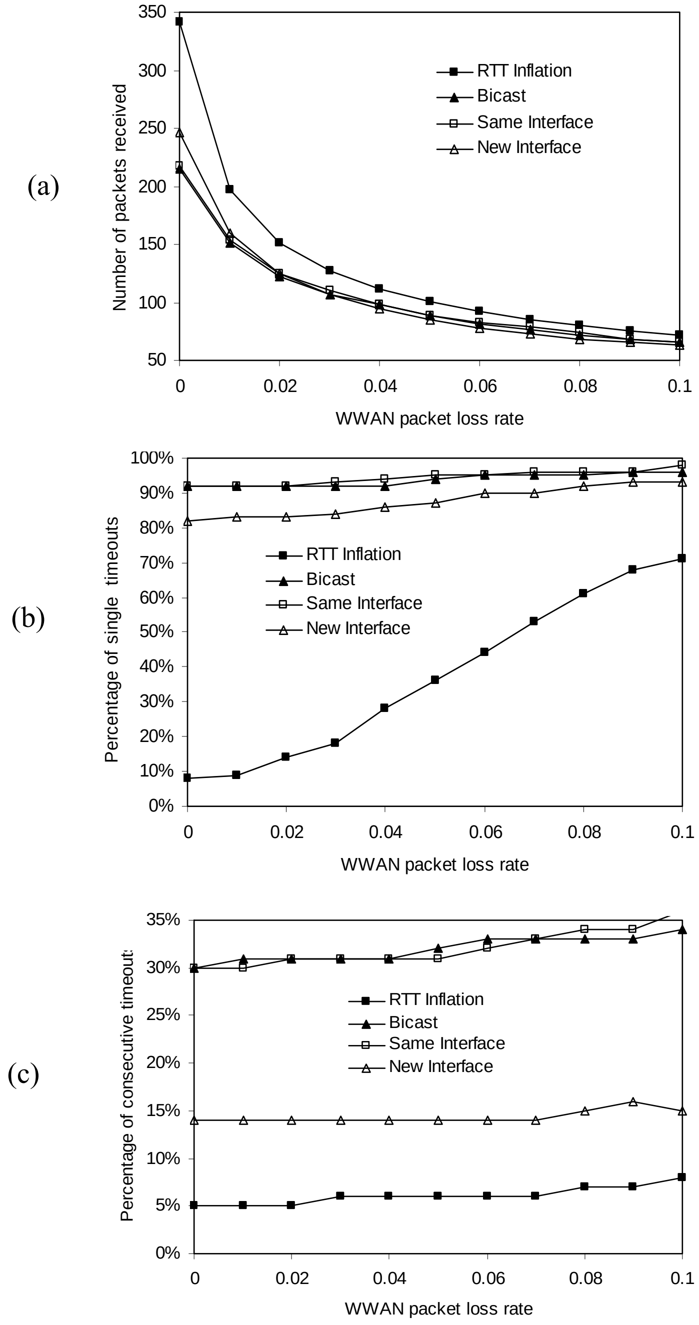


Figure 6.11. Impact of WWAN packet loss rate in upward vertical handover

6.4.3.1 Impact of information error

In the *RTT Equalization* scheme the equalization of round-trip delay experienced by all packets is mostly achieved by sending the acknowledgement packets correspond to data packets received from the WWAN and WLAN through the WLAN and WWAN, respectively. Therefore, estimation error in variable T has insignificant impact on the performance. In the *RTT Inflation* scheme, the premature timeouts may not be suppressed sufficiently if the inflated round-trip time set in variable *TO.initial* is too small. On the other hand, the process of recovering lost packets may be slowed down if the inflated round-trip time set in variable *TO.initial* is too large. Therefore, it is interesting to study the impact of inflated round-trip time estimation error in the variable *TO.initial*.

Figure 6.12 shows the number of in-order data packets received in 40 seconds as a function of percentage of RTT through WWAN set in *TO.initial* for WWAN packet loss rates (e_{BS}) = 0.050, 0.075, 0.100. Note that 100% is a perfect estimation, less than 100% is underestimation, and more than 100% is overestimation of the RTT through WWAN. It is found that the underestimation errors result in higher performance degradation than the overestimation error. Furthermore, it is noted that as e_{BS} increases, the significance of the underestimation errors decreases but the significance of overestimation errors slightly increases. These results imply that when configuring the *RTT Inflation* scheme it is much better to overestimate than underestimate the inflated round-trip time.

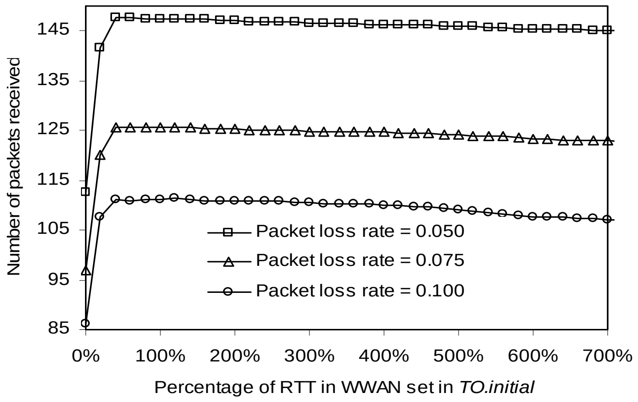


Figure 6.12. Impact of RTT estimation error in upward vertical handover

6.4.3.2 Impact of information failure

The prediction of hard upward vertical handover using link status information, the estimation of round-trip time using ICMP messaging (*icmp.RTT*), and the measurement of data transmission rate in the WWAN (b_{BS}) are the processes most vulnerable to failure in the proposed cross-layer design. In the *RTT Equalization* scheme, if b_{BS} is not available on time the variable *T* is initialized to zero. In the *RTT Inflation* scheme, if b_{BS} or *icmp.RTT* is not available on time the variable *TO.initial* is initialized to the default value (3 sec.). In both cases, the system does not crash. However, depending on the underlying network parameters, the initial values may not be suitable and therefore they can be considered as estimation errors which affect the system performance as discussed in Subsection 6.4.31. To study the impact of unexpected hard upward vertical handover on the *RTT Inflation* scheme, we compare two cases: when hard upward vertical handover occurs expectedly and unexpectedly. In each case, we first initialize the upward vertical handover and set the WLAN and WWAN available concurrently. Then, after a time interval called *breaking-time*, set WLAN not available. Note that if hard upward vertical handover is expected all generated acknowledgment packets are transmitted through the WWAN interface and standard procedure at the WP-TCP receiver are followed right after initializing the handover.

Figure 6.13 shows the number of in-order data packets received in 40 seconds for both expected and unexpected hard upward vertical handover. It can be seen that when undetected hard upward vertical handover occurs immediately after the initialization (*breaking-time* = 0 sec.), the performance degrades by approximately 27%, and when *breaking-time* increases to 0.1 sec. the performance improves by approximately 20%. Furthermore, when *breaking-time* = 0.12 sec., the performances in both cases increase abruptly to the same value and remain unchanged when the *breaking-time* increases. The performance penalty in the case of unexpected hard upward vertical handover is partially due to inflated round-trip time that slows down the recovery of excessive packet losses. A sharp increase noted at 0.12 sec. is due to the fact that after 0.1 sec. all in-flight data packets in the WLAN are received successfully and hence no data packet loss is experienced due to hard handover.

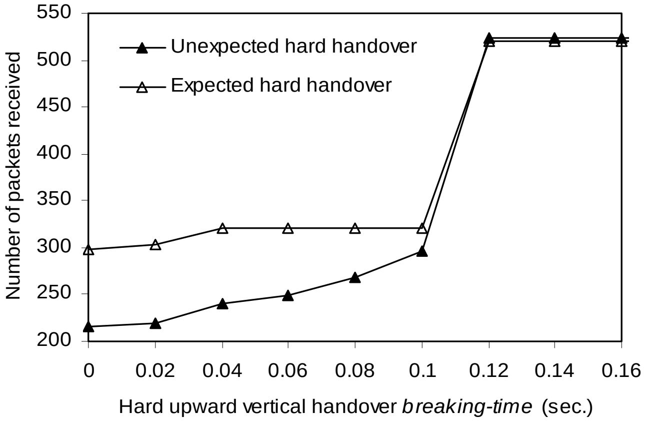


Figure 6.13. Impact of breaking-time in upward vertical handover

6.5 Summary

We have developed a cross-layer design to improve the vertical handover performance of WP-TCP in integrated static WLAN and WWAN. The cross-layer design is easy to implement and deploy, backward compatible with the WAP 2.0 architecture, and robust in the absence of perfect cross-layer information. Based on this design, we have proposed two proactive schemes, called *RTT Inflation* and *RTT Equalization*, which respectively prevent premature timeouts and suppress false fast retransmit. Extensive simulations are used to study and evaluate the performance of the proposed schemes. It is shown that the *RTT Inflation* and *RTT Equalization* schemes outperform other schemes and significantly improve the performance in a wide range of network conditions.

Chapter 7

Conclusions and Future Work

We conclude this thesis by presenting a summary of our contributions and future research directions.

7.1 Contributions

7.1.1 Wireless Links Models

One of the main issues in studying the performance of transport mechanisms is the availability of appropriate packet level models for wireless links. In our research, we have constructed a novel analytical model for WLAN link. Unlike the previous WLAN models [7], [8], [9], [10], [11], our analytical model captures the impact of both uncorrelated and correlated transmission errors. Also, we have developed analytical model which presents packet loss probability and packet loss burst length of WWAN-WLAN link. The developed models are suitable for studying the performance of transport mechanisms in integrated WLAN and WWAN. Furthermore, we have conducted an extensive analysis that complements the simulation and measurement studies [28], [29], [30] and sheds some light into future wireless networks design.

7.1.2 Analytical Frameworks

In this thesis, we have developed analytical frameworks for studying the performance of WP-TCP flows. Since the connection life time of the WP-TCP flows significantly influence the assumptions used in analyzing the performance, we have developed the frameworks for short-lived flows and long-lived flows separately.

- **Short-lived flows:** Most of the previous analytical frameworks for studying the performance of WP-TCP short-lived flows are based on an independent packet loss assumption which does not fit well in the wireless environment. In our research, we have developed an analytical framework for WP-TCP short-lived flows based on a two-state Markov channel model. The two-state Markov channel model captures both correlated and uncorrelated packet losses occur in WWAN, WLAN, and WWAN-WLAN links.
- **Long-lived flows:** We have developed an analytical framework for studying the performance of WP-TCP long-lived flows. The frameworks describe the short-term performance during vertical handover and long-term performance of WP-TCP for given network and protocol parameters. It captures the effects of vertical handover, such as excessive packet losses and sudden change in network characteristics, experienced in integrated static WLAN and WWAN.

Through the developed analytical framework, we have extensively analyzed the design parameters for WP-TCP short- and long-lived flows in various network conditions. The developed analytical frameworks are important in system planning and performance evaluation of future integrated heterogeneous wireless networks.

7.1.3 Cross-layer Design and Analysis

WP-TCP is fully compatible with standard TCP and it is optimized to operate over wireless links. Howerver, when WP-TCP is deployed in the integrated static WLAN and WWAN, its performance can dramatically degrade during and shortly after vertical handover due to premature timeouts or false fast retransmits. Previous solutions proposed in the literature to address the problem of premature timeouts and false fast retransmit are ineffective, complex, or impractical. By taking a different direction from previous works, we introduce a new solution which is more effective and practical. We first study the conditions that prevent the occurrence of premature timeout and false fast retransmit using our analytical framework for WP-TCP long-lived flows. Then, we propose a cross-layer design along with two proactive schemes at the mobile WP-TCP receiver. The proposed solution significantly improves the performance in a wide range of practical WP-TCP parameters and network conditions. In addition, the solution is easy to implement and deploy, compatible with traditional TCP, and robust in the events of imperfect of cross-layer information.

7.2 Future Research

This section presents our future research directions, which are related to the work described in this thesis.

7.2.1 Adaptive Transport Mechanisms

The need to improve efficiency, safety, and security of the transportation systems while providing users on the road broadband access to new wireless Internet applications and services has led to development of advanced intelligent transportation systems (ITS). We envision that ITS-based wireless vehicular networks (WVN) will support greedy multimedia wireless Internet applications while running vital safety applications with strict quality of service (QoS) requirements such as reliable delivery, delay bounds, and message sequencing. In addition, participating vehicles will use multiple wireless access technologies and simultaneously communicate with other vehicles in ad hoc mode (vehicle-to-vehicle communications) and roadside facilities in infrastructure mode (vehicle-to-roadside communications).

Creating scalable, high-performance, and robust transport mechanisms for ITS-base WVNs presents a great challenge to research community. Most of the existing transport mechanisms cannot be directly deployed due to highly dynamic and heterogeneity nature of future ITS-based WVNs. In the future, we plan to extend our cross-layer approach presented in Chapter 5 and develop new transport mechanisms which can adapt and perform well in both vehicle-to-vehicle and vehicle-to-roadside communications modes. The new transport mechanisms also need to support coexistence of safety applications with strict requirement and greedy wireless Internet applications. Therefore, we intend to investigate joint congestion control and service differentiation mechanisms at the transport layer, routing schemes at the network layer, and rate control and priority queuing mechanisms at the MAC sub-layer in order to support the coexistence of the applications.

7.2.2 Network Residence Times

Tractable distributions, which describe network residence times in integrated heterogeneous wireless networks, are very important in studying various network aspects such as security [79], resource allocations [80], and transport mechanisms. Unfortunately, at present, there is no well-established distribution which can describe network residence time in integrated heterogeneous wireless networks. In Chapter 5, we use exponentially distributed residence times to study the performance of WP-TCP in integrated static WLAN and WWAN. In the future, we plan to investigate and develop distributions which are tractable and consistent with empirical data from integrated heterogeneous wireless networks that involve WWANs (e.g., cellular networks), WMANs (e.g., IEEE 802.16/20), WLANs (e.g., IEEE 802.11), and WPANs (e.g., IEEE 802.15). Developing tractable distributions is very challenging task due to the fact that most of wireless networks are still in the infant stage and they are evolving very fast. For example, Rogers Communications and Bell Canada are jointly building and managing a Canada-wide wireless broadband network, which is expected to reach more than two-thirds of all Canadians by the end of 2008.

7.2.3 Cross-layer Architecture

The results from this thesis and other research [81] indicate that cross-layer coupling can significantly improve the performance of transport mechanisms in wireless networks. On the other hand, some studies [78] have shown that cross-layer coupling can lead to unintended interactions and inflexibility that may respectively cause system instabilities and potentially hinder future innovations. The establishment of well-balanced design methodologies for wireless networks is still an open issue and more research is needed.

7.3 Final Remarks

In this thesis, we have developed new analytical frameworks for studying the performance of WP-TCP flows in integrated WLAN and WWAN. The developed analytical frameworks are flexible, cost-effective, and produce results with reasonable accuracy. Also, we have presented the analysis of WP-TCP in integrated WLAN and WWAN, which provides insights for further research. In addition, we have proposed a solution to improve the performance of WP-TCP in vertical handover. The solution is relatively simple, easy to implement, and compatible with standard TCP implementations.

Appendix A

Derivation of E[*slot*]

The average slot length $E[slot]$ is given in [8] as

$$E[slot] = (1 - P_{tr})\sigma + P_{tr} P_S T_S + P_{tr}(1 - P_S) T_C,$$

where σ is the time duration of an empty slot. $P_{tr} = 1 - (1 - \tau)^n$ is the probability that there is at least one transmission in a given time slot. P_S is the probability that the transmission is successful and is calculated as

$$P_S = \frac{n\tau(1-\tau)^{n-1}}{1-(1-\tau)^n} \pi_g^{WLAN},$$

where π_g^{WLAN} is the steady state probability of a good WLAN link condition. T_C and T_S are time durations that the WLAN link is sensed during a collided frame transmission and busy during a successful packet transmission, respectively. Therefore, $T_C = DIFS + H + P + SIFS + ACK$ and $T_S = DIFS + H + P + \delta + SIFS + ACK + \delta$, where δ is the propagation delay. *DIFS* and *SIFS* represent DCF inter frame space and small inter frame space, respectively. *H*, *P*, *ACK* are respectively the transmission times for the header, payload, and ACK frame.

Appendix B

List of Abbreviations

ACK	Acknowledgement
AIMD	Additive Increase and Multiplicative Decrease
AP	Access Point
ARQ	Automatic Repeat reQuest
BDP	Bandwidth-Delay Product
BS	Base Station
CDMA	Code-Division Multiple Access
CN	Correspondent Node
CR	Common Router
DCF	Distributed Coordination Function
DIFS	DCF Inter Frame Space
ECN	Explicit Congestion Notification
FIN	Finish
FTP	File Transfer Protocol
GPRS	General Packet Radio Service
HiperLAN	High Performance Radio Local Access Network
HTTP	Hyper Text Transfer Protocol
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ISM	Industrial, Scientific, and Medical
ITS	Intelligent Transportation System

MAC	Medium Access Control
MN	Mobile Node
MSS	Maximum Segment Size
NLOS	Non-Line-Of-Sight
OSI	Open Systems Interconnection
QoS	Quality of Service
RFC	Request For Comments
RTO	Retransmission Timeout
RTT	Round-Trip Time
RTTVAR	Round-Trip Time Variation
SACK	Selective Acknowledgement
SIFS	Small Inter Frame Space
SRTT	Smoothed Round-Trip Time
SYN	Synchronize
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TO	Time-Out
TSecr	Timestamp Echo Reply
TSVal	Timestamp Value
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunication system
VHO	Vertical Handover
WAE	Wireless Application Environment
WAP	Wireless Application Protocol
WDP	Wireless Datagram Protocol
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WPAN	Wireless Personal Area Network
WP-TCP	Wireless Profiled TCP
WVN	Wireless Vehicular Network
WWAN	Wireless Wide Area Network

Bibliography

[1] Computer Industry Almanac Inc., http://www.c-i-a.com 2006.

[2] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*, New York: Wiley-Interscience, 1991.

[3] M. Zorzi, R. Rao, and L. Milstein, "ARQ Error Control for Fading Mobile Radio Channels," *IEEE Trans. on Vehicular Technology*, vol. 46, no. 2, pp. 445-455, 1997.

[4] J. Yeo and A. Agrawala, "Packet Error Model for the IEEE 802.11 MAC Protocol," *Proc. IEEE PIMRC 2003,* September 2003.

[5] O. Fernandez, M. Domingo, and P. Torres, "Experimental analysis of wireless data transmission systems in space platforms," *IEEE Antennas and Propagation Magazine*, vol. 46, no. 4, pp. 38-60, 2004.

[6] S. Khayam, S. Karande, H. Radha, and D. Loguinov, "Performance Analysis and Modeling of Errors and Losses over 802.11b LANs for High-bit-rate Real-time Multimedia," *Elsevier Signal Processing: Image Communication*, vol. 18, no. 47, pp. 575-595, 2003.

[7] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535-547, 2000.

[8] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma, "Performance of Reliable Transport Protocol over IEEE 802.11 Wireless LAN: Analysis and Enhancement," *Proc. IEEE INFOCOM 2002,* June, 2002.

[9] P. Chatzimisios, A. Boucouvalas, and V. Vitsas, "IEEE 802.11 Packet Delay - A Finite Retry Limit Analysis," *Proc. IEEE GLOBECOM 2003,* San Francisco, California, USA, December 1-5, 2003.

[10] P. Chatzimisios, A. Boucouvalas, and V. Vitsas, "Performance analysis of IEEE 802.11 DCF in presence of transmission errors," *Proc. IEEE ICC 2004,* June, 2004.

[11] Z. Hadzi-Velkov and B. Spasenovski, "Saturation Throughput-Delay Analysis of IEEE 802.11 DCF in Fading Channel," *Proc. IEEE ICC 2003,* May, 2003.

[12] A. Iera, A. Molinaro, S. Polito, and G. Roggeri, "End-to-End QoS Provisioning in 4G with Mobile Hotspots," *IEEE Network*, vol. 19, no. 5, pp. 26-34, 2005.

[13] P. Rodriguez, R. Chakravorty, J. Chesterfield, I. Pratt, and S. Banerjee, "MAR: A Commuter Router Infrastructure for the Mobile Internet," *Proc. ACM MOBISYS 2004,* June, 2004.

[14] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol," IETF RFC 3963, January 2005.

[15] R. Gau and C. Lin, "Location Management of Correlated Mobile Users in the UMTS," *IEEE Trans. on Mobile Computing*, vol. 4, no. 6, pp. 641-651, 2005.

[16] G. Korkmaz, E. Ekici, and F. Ozguner, "A New High Throughput Internet Access Protocol for Vehicular Networks," *Proc. ACM VANET 2005,* Cologne, Germany, September 2, 2005.

[17] R. Wakikawa, K. Okada, R. Koodli, A. Nilsson, and J. Murai, "Design of Vehicle Network: Mobile Gateway for MANET and NEMO Converged Communication," *Proc. ACM VANET 2005,* September 2005.

[18] K. Lan, H. Petander, E. Perrera, L. Libman, C. Dwertman, and M. Hassan, "MOBNET: The Design and Implementation of a Network Mobility Testbed for NEMO protocol," *Proc. IEEE LANMAN 2005,* September, 2005.

[19] T. Ernst, R. Kuntz, and R. Leiber, "A Live Light-Weight IPv6 Demonstration Platform for ITS Usages," *Proc. ITST 2005,* June, 2005.

[20] R. Kuntz, K. Mituya, and R. Wakikawa, "Performance Evaluation of NEMO Basic Implementations," *Proc. WONEMO 2006,* January, 2006.

[21] N. Cardwell, S. Savage, and T. Anderson, "Modeling TCP latency," *Proc. IEEE INFOCOM 2000,* Tel Aviv, Israel, March, 2000.

[22] M. Mellia, I. Stoica, and H. Zhang, "TCP model for short lived flows," *IEEE Communications Letters*, vol. 2, no. 2, pp. 85-87, 2002.

[23] B. Sikdar, S. Kalyanaraman, and K. S. Vastola, "Analytic models for the latency and steady-state throughput of TCP Tahoe, Reno, and SACK," *IEEE/ACM Trans. on Networking*, vol. 11, no. 6, pp. 959-971, 2003.

[24] C. Barakat and E. Altman, "Performance of short TCP transfers," *Lecture Notes in Computer Science, NETWORKING*, vol. 1815, pp. 567-579, Springer, 2000.

[25] M. Rossi, R. Vicenzi, and M. Zorzi, "Accurate analysis of TCP on channels with memory and finite round-trip delay," *IEEE Trans. on Wireless Communications*, vol. 3, no. 2, pp. 627-640, 2004.

[26] M. Zorzi, A. Chockalingam, and R. R. Rao, "Throughput analysis of TCP on channels with memory," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1289-1300, 2000.

[27] C. Barakat, "TCP/IP modeling and validation," *IEEE Network*, vol. 15, no. 3, pp. 83-74, 2001.

[28] A. Gurtov and J. Korhonen, "Effect of vertical handovers on performance of TCP-friendly rate control," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 3, pp., 2004.

[29] R. Chakravorty, P. Vidales, K. Subramanian, I. Pratt, and J. Crowcroft, "Performance issues with vertical handovers - experiences from GPRS cellular and WLAN hot-spots integration," *Proc. IEEE PerCom 2004,* Orlando, Florida, USA, March 14-17, 2004.

[30] Y. Gou, D. A. J. Pearce, and P. D. Mitchell, "A Receiver-based Vertical Handover Mechanism for TCP Congestion Control," *IEEE Trans. on Wireless Communications*, vol. 5, no. 10, pp. 2824-2833, 2006.

[31] M. Zhang, B. Karp, S. Floyd, and Peterson, "RR-TCP: A Reordering-Robust TCP with DSACK," *Proc. IEEE ICNP'03,* Atlanta, Georgia, November, 2003.

[32] E. Blanton and M. Allman, "On making TCP more robust to packet reordering," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 1, pp. 20-30, 2002.

[33] R. Ludwig and R. H. Katz, "The Eifel algorithm: making TCP robust against spurious retransmissions," *Computer Communications Review*, vol. 30, no. 1, pp. 30-36, 2000.

[34] H. Huang and J. Cai, "Improving TCP performance during soft vertical handoff," *Proc. IEEE AINA'05,* Taipei, Taiwan, March, 2005.

[35] W. Hansmann and M. Frank, "On things to happen during a TCP handover," *Proc. IEEE LCN'03,* König-swinter, Germany, October, 2003.

[36] H. Rutagemwa, X. Shen, and J. W. Mark, "Performance Evaluation of Wireless Application Protocol for Reliable Connection-Oriented Services," *Annual Review of Communications*, vol. 58, pp. 543-550, 2005.

[37] S. Pack, H. Rutagemwa, X. Shen, J. W. Mark, and L. Cai, "Performance Analysis of Mobile Hotspots with Heterogeneous Wireless Links," *IEEE Trans. on Wireless Communications*, to appear.

[38] H. Rutagemwa, X. Shen, and J. W. Mark, "Transfer Delay Analysis of WAP 2.0 over Wireless Links for Short-Lived Flows," *IEEE Trans. on Vehicular Technology*, to appear.

[39] H. Rutagemwa, M. Shi, X. Shen, and J. W. Mark, "Wireless Profiled TCP Performance over Integrated Wireless LANs and Cellular networks," *IEEE Trans. on Wireless Communications*, to appear.

[40] H. Rutagemwa, S. Pack, X. Shen, and J. W. Mark, "Robust Cross-layer Design of Wireless Profiled TCP Mobile Receiver for Vertical Handover," *IEEE Trans. on Vehicular Technology*, to appear.

[41] S. Pack, H. Rutagemwa, X. Shen, J. W. Mark, and L. Cai, "An Analytical Framework for Studying the Performance of Mobile Hotspots," *Proc. IEEE Globecom'06,* San Francisco, California, USA, Nov. 27-Dec. 1, 2006.

[42] S. Pack, H. Rutagemwa, X. Shen, J. W. Mark, and L. Cai, "An Integrated WWAN-WLAN Link Model in Mobile Hotspots," *Proc. CHINACOM 2006,* Beijing, China, October 25-27, 2006.

[43] H. Rutagemwa, X. Shen, and J. W. Mark, "Latency Analysis of WAP 2.0 for Short-Lived Flows," *Proc. IEEE WCNC'05,* New Orleans, Louisiana, USA, March 13-17, 2005.

[44] H. Rutagemwa, M. Shi, X. Shen, and J. W. Mark, "Performance of Wireless Profiled TCP in Heterogeneous Wireless Networks," *Proc. IEEE ICC'06,* Istanbul, Turkey, June 11-15, 2006.

[45] H. Rutagemwa, S. Pack, X. Shen, and J. W. Mark, "Cross-layer Design and Analysis of Wireless Profiled TCP for Vertical Handover," *Proc. IEEE ICC'07,* Glasgow, UK, June 24-28, 2007.

[46] K.-H. Liu, H. Rutagemwa, X. Shen, and J. W. Mark, "Efficiency and Goodput Analysis of Dly-ACK in IEEE 802.15.3," *IEEE Trans. on Vehicular Technology*, to appear.

[47] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," IETF RFC 3775, June 2004.

[48] H. Rutagemwa and X. Shen, "Modeling and Analysis of WAP Performance over Wireless Links," *IEEE Trans. on Mobile Computing*, vol. 2, no. 3, pp. 221-232, 2003.

[49] "Wireless application protocol," WAP Forum, http://www.wapforum.org, 2001.

[50] "WAP architecture specification," WAP Forum, http://www.wapforum.org, 2001

[51] "Wireless profiled TCP," WAP Forum, http://www.wapforum.org, version 31-March-2001, 2001.

[52] J. Postel, "Transmission Control Protocol," RFC 793, September 1981.

[53] R. Braden, "Requirements for Internet Hosts - Communication Layers," IETF STD 3, RFC 1122, October 1989.

[54] M. Allman, V. Paxson, and W. Stevens, "TCP Congestion Control," IETF RFC 2581, April 1999.

[55] M. Mathis, J. Mahdavi, S. Floyd, and R. Romanow, "TCP Selective Acknowledgment Options," IETF RFC 2018, October 1996.

[56] V. Jacobson, R. Braden, and D. Borman, "TCP Extensions for High Performance," IETF RFC 1323, May 1992.

[57] M. Allman, S. Floyd, and C. Partridge, "Increasing TCP's Initial Window," IETF RFC 2414, September 1998.

[58] J. Mogul and S. Deering, "Path MTU Discovery," IETF RFC 1191, November 1990.

[59] J. McCann, S. Deering, and J. Mogul, "Path MTU Discovery for IP version 6," IETF RFC 1981, August 1996.

[60] K. Ramakrishnan and S. Floyd, "A Proposal to add Explicit Congestion Notification (ECN) to IP," *IETF RFC 2481*, to appear.

[61] Y. Tian, K. Xu, and N. Ansari, "TCP in Wireless Environments: Problems and Solutions," *IEEE Radio Communications Magazine*, vol. 43, no. 3, pp. 27-32, 2005.

[62] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, Addison-Wesley, 2000.

[63] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification: High-speed Physical Layer Extension in the 2.4 GHz Band," IEEE 802.11b WG Part 11, September 1999.

[64] D. Qiao, S. Choi, and K. Shin, "Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs," *IEEE Trans. on Mobile Computing*, vol. 1, no. 4, pp. 278-292, 2002.

[65] W. C. Jakes, *Microwave mobile communications*, Piscataway: IEEE Press, 1993.

[66] H. Shen, L. Cai, and X. Shen, "Performance Analysis of TFRC over Wireless Link with Truncated Link Level ARQ," *IEEE Trans. on Wireless Communications*, vol. 5, no. 6, pp. 1479-1487, 2006.

[67] T. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed., New Jersey: Prentice Hall, 1996.

[68] V. Jimenez, M. Garcia, and A. Armada, "Channel Estimation for Bit-loading in OFDM-based WLAN," *Proc. IEEE ISSPIT 2002,* December, 2002.

[69] The Network Simulator ns-2, http://www.isi.edu/nsnam/ns, 2007.

[70] R. A. Howard, *Dynamic Probabilistic Systems*, New York: Wiley, 1971.

[71] S. M. Ross, *Introduction to Probability Models*, 7th ed., Harcourt, 2000.

[72] M. Zorzi and R. R. Rao, "Lateness probability of a retransmission scheme for error control on a two-state Markov channel," *IEEE Trans. on Communications*, vol. 47, no. 10, pp. 1537-1548, 1999.

[73] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, "Modeling TCP Reno performance: A simple model and its empirical validation," *IEEE/ACM Trans. on Networking*, vol. 8, no. 2, pp. 133-145, 2000.

[74] T. Goff, J. Moronski, D. S. Phatak, and V. Gupta, "Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments," *Proc. IEEE Infocom 2000,* Tel Aviv, Israel, 2000.

[75] S. Kim and J. A. Copeland, "Interworking between WLANs and 3G Networks: TCP Challenges," *Proc. IEEE WCNC'04,* Atlanta, Georgia, USA, March 21-25, 2004.

[76] Y. Matsushita, T. Matsuda, and M. Yamamoto, "TCP congestion control with ACK-Pacing for vertical handover," *Proc. Proc. IEEE WCNC'05,* New Orleans, Louisiana, USA, March, 2005.

[77] V. T. Raisinghani and S. Iyer, "Cross-layer design optimizations in wireless protocol stacks," *Computer Communications Review*, vol. 27, pp. 720-24, 2004.

[78] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross layer design," *IEEE Wireless Communications Magazine*, vol. 12, no. 1, pp. 3-11, 2005.

[79] M. Shi, H. Rutagemwa, X. Shen, and J. W. Mark, "A Service Agent Based Roaming Architecture for WLAN/Cellular Integrated Networks," *IEEE Trans. on Vehicular Technology*, to appear.

[80] W. Song, H. Jiang, W. Zhuang, and X. Shen, "Resource Management for QoS Support in WLAN/Cellular Interworking," *IEEE Network - special issue on 4G Network Technologies for Mobile Telecommunications*, vol. 19, no. 5, pp. 12-18, 2005.

[81] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," *IEEE Communications Magazine*, vol. 41, no. 10, pp. 74- 80, 2003.