TCP Performance over Wi-Fi: Joint Impact of Buffer and Channel Losses

Shiva Raj Pokhrel, Manoj Panda, Hai L. Vu, *Senior Member, IEEE*, and Michel Mandjes

Abstract—We propose an analytical model for a Wi-Fi network acting as a last-mile Internet access with multiple long-lived TCP connections on both the up and down links. Our model considers the joint impact of buffer losses at the access point, contention at the medium access control layer, and packet losses due to the wireless channel being erroneous. We show that the model accurately quantifies the probability of an arbitrary TCP packet being discarded, and the total throughput obtained on the up and down links. Furthermore, quantitative insights can be gained into the throughput that long-lived TCP flows achieve under the joint impact of all aforementioned types of losses. In particular, we find that the wireless channel errors and buffer overflows both lead to throughput unfairness, but that they do so in the opposite direction on the up and down links, respectively. We demonstrate that this insight can be exploited so as to significantly mitigate the throughput unfairness without compromising the total obtainable network throughput.

Index Terms—IEEE 802.11 WLAN, congestion control, fixed point, wireless errors, unfairness, TCP, bandwidth-delay product

1 Introduction

With the growing presence of Wi-Fi and cellular last-mile Internet access, the evolution towards *Internet of Things* (IoT) is already evident [1]. About 90 percent of the DATA traffic in the Internet today is carried by the Transmission Control Protocol (TCP), and a majority of that traffic in IoT will be preferably transferred via a path with Wi-Fi last-mile which is significantly faster and cheaper than a cellular connection. An in-depth understanding of TCP dynamics over Wi-Fi networks is thus essential to effectively design, deploy and manage a large number of devices in IoT, and analytical modeling can provide such important insights.

Wi-Fi networks, or more precisely, IEEE 802.11 wireless local area networks (WLANs), often operate in a so-called *infrastructure* mode. In infrastructure Wi-Fi [2], user devices or stations (STAs) associate with an access point (AP) and transceive the DATA using a variant of the carrier sense multiple access (CSMA) medium access control (MAC) protocol. An accurate prediction of the performance of TCP over infrastructure Wi-Fi Internet access will be highly useful for efficient selection of low-cost paths. For example, *multihomed* STAs that are in the vicinity of two or more APs may dynamically switch their associations between the APs based on some analytically predicted performance metrics, and thereby, achieve optimal performance.

The performance of TCP over wireless networks has been widely studied in the literature using simulations, analytical methods and real-world experiments. Detailed surveys can be found in [3], [4], and [5]. In this paper, we restrict our discussion to the analytical modeling of TCP over last-mile wireless networks. One class of analytical models for such networks studies the detailed dynamics of TCP with an abstraction for the lower layers. This is done via approximating the actual packet loss process by an analytically tractable loss model. The analysis with an i.i.d. packet loss model can be found in [6] and that with stationary random losses is developed in [7]. Analyses with more elaborate correlated loss models that better capture the loss characteristics of the wireless channel are carried out in [8] and [9]. However, these works [6], [7], [8], [9] do not explicitly consider the losses due to the AP's buffer overflows.

Abouzeid et al. [10] analyze the performance of TCP Reno accounting for both wireless and buffer losses. A comprehensive model for the performance of TCP NewReno with cellular last-mile access has been recently reported in [11], where the impact of auto- and cross-correlations between wireless and buffer losses is investigated.

Nevertheless, all the aforementioned works do not take into account the impact of the MAC access layer (e.g. channel contention and collisions) on the performance of TCP. Our aim in this paper is to understand the performance of TCP over an infrastructure Wi-Fi last-mile access where the MAC layer plays an important role, and one cannot approximate the lower layers simply by a packet loss model. Modeling the joint impact of buffer and wireless losses over Wi-Fi last-mile access is significantly more challenging than that over cellular last-mile for the following two reasons:

i. In cellular last-mile, each user device is allocated a separate channel. However, in Wi-Fi last-mile, the common medium has to be shared by multiple STAs, leading to contentions for medium access and collisions. Unlike in cellular networks, where the impact of other users can be modelled through an interference power, the time sharing of the common medium in Wi-Fi networks requires a detailed and careful modeling of the contention for medium access.

• *S.R. Pokhrel, M. Panda, and H.L. Vu are with the Faculty of Science, Engineering and Technology, Swinburne University of Technology, Melbourne, Victoria, Australia. E-mail: {spokhrel, mpanda, hvu}@swin.edu.au.*
• *M. Mandjes is with the Korteweg-de Vries Inst. for Mathematics, University of Amsterdam, The Netherlands. E-mail: m.r.h.mandjes@uva.nl.*

Manuscript received 26 Mar. 2015; revised 15 June 2015; accepted 6 July 2015. Date of publication 22 July 2015; date of current version 31 Mar. 2016. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TMC.2015.2456883

TABLE 1
Literature on Performance Modeling of TCP Over Infrastructure WLAN

References	buffer loss	wireless loss	Scenario of STAs
[12], [13], [14]	–	–	only downloads
[15], [16], [17], [18], [19]	–	–	uploads+downloads
[20], [21], [22], [23], [24], [25]	✓	–	uploads+downloads
[26]	✓	✓	only downloads
this work	✓	✓	uploads+downloads

ii. In cellular last-mile, it is often the case that each flow is allocated a separate buffer. In Wi-Fi last-mile, however, the same bottleneck buffer at the AP has to be shared by multiple TCP connections.

A brief summary of the literature on the performance modeling of TCP over infrastructure Wi-Fi WLANs is provided in Table 1.

In particular, the authors in [12], [13], [14], analyze the performance of long-lived TCP connections with only downloading STAs. In [15], [16], [17], [18], [19], the authors study TCP's flow control with both uploading and downloading STAs. It is worth noting that the overflows at the bottleneck AP buffer lead to an unequal division of the total throughput, or the so-called *throughput unfairness*, where the uploading STAs obtain a higher share of throughput than the downloading STAs. This well-known unfairness is studied and evaluated in [20], [21], [22], [23], [24], [25]. However, all of the above works ignore wireless channel errors [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25] and consider an ideal channel where MAC collisions are the only source of wireless losses. Herein, we will study a non-ideal channel where errors, in addition to the MAC collisions, might also occur due to a low signal-to-noise ratio (SNR) which is a more realistic setting but requires a different modeling approach as described later in this paper. It is due to the heterogenous impact of channel errors on TCP DATA packets and its acknowledgements (ACKs) depending on their sizes.

The only work in the literature with both buffer and channel losses is [26], where the authors analyze the case of multiple classes of downloading STAs characterized by different probabilities of the wireless channel error. Nevertheless, they only consider downloading STAs. In contrast, we will consider a scenario with both uploading and downloading STAs (see Fig. 1), and develop an accurate and comprehensive model for the MAC discards, i.e. packet losses due to the combination of wireless errors and MAC collisions. The inclusion of both upload and download in our model is non-trivial because of the coupling between the two directional TCP traffic influenced by the elaborate contention access mechanism. Furthermore, the analytical model in [26] involves a computationally complex Markov chain with a large state space. We, however, adopt a fixed-point approach [11], [27], [28] that allows tremendous flexibility in developing and upgrading the analytical models. Our three main contributions in this paper are summarized below.

C$_1$ *Our model accurately captures the long-lived throughput that competing uploading and downloading TCP connections obtain over an infrastructure WLAN, under an erroneous wireless channel and finite AP buffer (Section 3).*

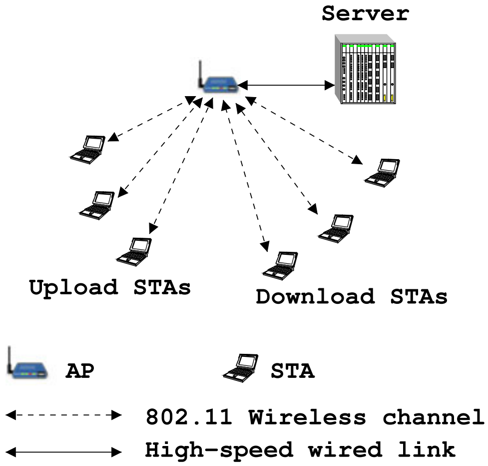

Fig. 1. An infrastructure WLAN consisting of an AP, N_u uploading STAs and N_d downloading STAs. The STAs communicate with servers in the Internet through the AP using long-lived TCP connections in errorneous wireless channel. The servers are connected to the AP by high-speed wired links with negligible delay.

Our analysis is comprehensive as we consider the interplay of four key factors: (i) buffer overflows at the bottleneck AP, (ii) wireless channel errors, (iii) TCP's flow and congestion control, and (iv) contention and collisions in the MAC layer. Our model facilitates the accurate computation of the probability that an arbitrary TCP packet is discarded by the MAC layer (see Fig. 8). It also captures the joint impact of wireless channel errors and bottleneck buffer overflows on the throughputs obtained by uploading and downloading STAs (Eq. (23)), and the impact of the bandwidth delay product (BDP) with a Wi-Fi last-mile access (Eq. (20)). The accuracy of the results from our model is validated by extensive simulations.

C$_2$ *We discover that wireless channel errors cause a throughput unfairness that is opposite by nature to the well-known unfairness due to buffer overflows. Our analytical model provides the means to gain detailed insights into this unfairness, as well as an explanation of this behavior.*

Specifically, if the capacity of the AP buffer is large enough so that the probability of buffer overflow is negligible, then, under non-zero wireless channel errors, download connections obtain a higher share of throughput than upload connections (Eq. (14)). This unfairness is in an opposite sense to the unfairness due to buffer overflows. To the best of our knowledge, we are the first to report and provide analytical explanations for this interesting observation (Section 3.1.5). The reasons for this new type of unfairness is discussed with relevant results in Section 4.2.

C$_3$ *We exploit one type of throughput unfairness against the other, and propose two ways to mitigate the unfairness, namely, admission control and buffer sizing, to achieve any desired ratio of upload to download throughput (Section 5). This is achieved without any negative impact on the total throughput.*

In general, there exists a trade-off between throughput and fairness [29], [30]. However, by means of the developed model we know that the unfairness due to buffer overflows and that due to channel errors are opposite by nature, and thus we could achieve any desired ratio of throughput without compromising the total throughput by compensating one type of unfairness against the other (Section 5). We also illustrate how the appropriate solution may be selected based on the magnitude of the wireless channel errors.

The remainder of the paper is organized as follows. In Section 2, we describe our network scenario in detail. An analytical model using the fixed point approach is developed in Section 3. In Section 4, the model is validated using simulations and the new type of unfairness is explained in detail. Two methods to achieve throughput fairness are proposed and their effectiveness is studied in Section 5. The paper concludes in Section 6.

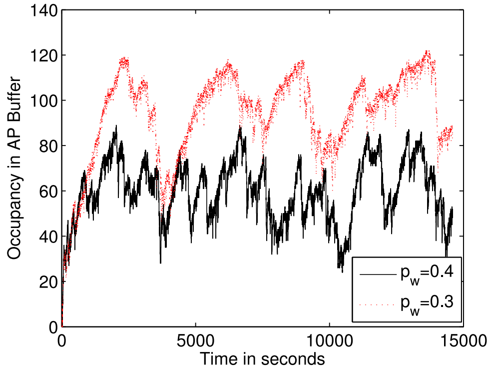

Fig. 2. Saturated AP with wireless channel error probability, p_w and five upload and five download long lived TCP connections as observed in NS-2 simulations.

2 Network Scenario

We consider an 802.11 infrastructure WLAN as depicted in Fig. 1. There are N_u STAs uploading and N_d STAs downloading data through the AP using *long-lived* TCP connections. The AP and STAs communicate using the Distributed Coordination Function (DCF) Medium Access Control protocol as defined in the IEEE 802.11 standard. The TCP sources of the upload connections and the TCP sinks of the downloading connections reside at the upload and download STAs, respectively. The TCP sources of the download connections and the TCP sinks of the uploading connections reside in servers outside the WLAN. The servers are connected to the AP by high-speed wired links and propagation delays are negligible. Each TCP/IP packet (i.e., a TCP DATA packet or a TCP ACK) is encapsulated into one MAC frame and is transferred between the AP and the STAs using the DCF MAC protocol.

Transmissions of MAC frames may fail due to *collisions* and/or *channel errors*. Collisions occur when two or more nodes (i.e., AP or STA) attempt transmission in the same so-called *slot* [31], [32]. Channel errors occur due to high signal attenuation, shadowing and multipath fading or due to non-802.11 interference. In the DCF MAC protocol, every failed transmission is interpreted as a collision even though the transmission might have been failed due to channel errors. After every transmission failure (detected by non-reception of MAC-level ACK within a timeout), the MAC contention window is doubled (unless it has already attained the maximum allowed value) and the packet is retransmitted up to K times. A packet is *discarded* if all the $(K+1)$ (re)transmissions for that packet fail, in which case it is dequeued and the next head-of-line (HOL) packet, if any, is taken up for transmission.

To overcome the adverse impact of bad channel conditions, MAC frames containing payload are protected with forward error correction (FEC) that can recover payload DATA from up to a certain number of bit-level errors. The MAC frames containing TCP ACKs are so short that, due to FEC, they are seldom lost under realistic channel errors. Moreover, since TCP ACKs are cumulative, the information contained in the lost ACK is compensated by a subsequent successfully reached ACK. TCP DATA packets, however, are long enough to suffer from channel errors. The probability of transmission failure of a DATA frame with one transmission attempt is the probability that the transmission of a MAC frame containing a TCP DATA packet fails due to wireless channel error and/or collision.

We consider the cases of both infinite and finite bottleneck AP buffers to understand the impact of wireless channel errors. In reality, the AP buffer is always finite, and a TCP packet may be lost due to buffer overflow. A TCP DATA packet that is either lost due to buffer overflow or discarded by the MAC due to repeated (re)transmission failures has to be recovered by TCP-level retransmission. The TCP DATA packet loss is interpreted by the TCP source as an indication of congestion and always leads to a multiplicative reduction of TCP's congestion window and a loss recovery process (the details of which depend on the version of TCP).

The TCP DATA packets and TCP ACKs belonging to all the download and upload connections, respectively, must be served by the AP regardless whether successfully transmitted or discarded. Thus, the wireless channel in the direction from the AP to the STAs is the bottleneck. Under zero channel errors, it is known that due to the closed-loop nature of TCP, on average there are only 1.5 STAs *active* (i.e., have TCP DATA packets or TCP ACKs in their MAC queues and contend for the channel access) irrespective of how many STAs are actually present in the WLAN [16], [22]. We remark that this property essentially carries over to our case with wireless channel errors as well (see (7) and (8)), and almost all of the TCP DATA packets and TCP ACKs belonging to the download and upload connections, respectively, queue up at the AP's buffer for transmission.

3 Analytical Model

For tractability of the analysis, we use the standard assumptions as in most prior works [16], [22].

A1. The average impact due to lost ACKs is negligible. TCP ACKs are cumulative. Also, the MAC frames containing TCP ACKs are so short that, with FEC, they are seldom lost under realistic channel errors.

A2. The AP is saturated, i.e., the AP buffer is never empty.

(A2) is observed in simulation as depicted in Fig. 2 and also in practice [33] (see justification below).

A3. A STA can have at most one packet in its MAC queue. The successful service of a TCP DATA packet (resp. TCP ACK) from the AP makes one more

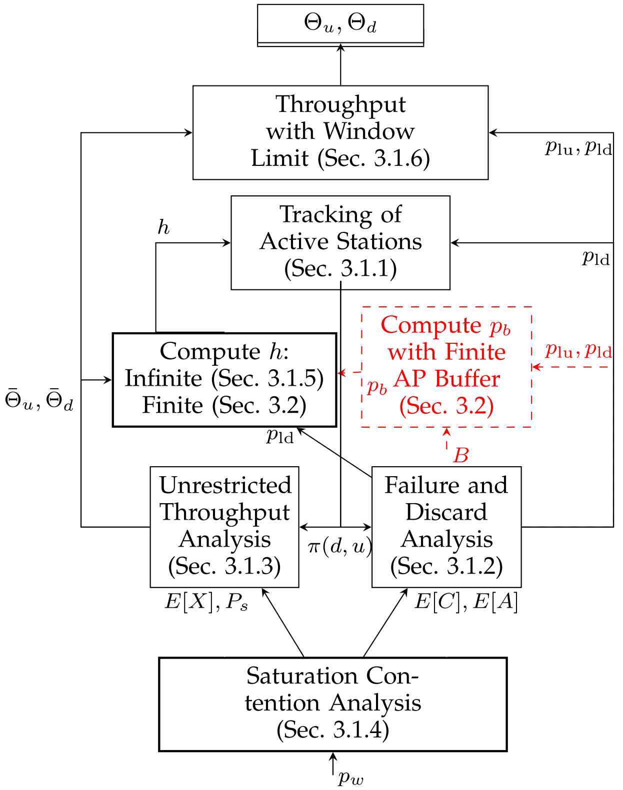

Fig. 3. Analytical Block Diagram. The block shown by a dotted rectangle exists only for the case of finite AP buffer. The block shown by a thick rectangle for computation of h is different for the infinite and finite buffer cases. All other blocks are identical for the infinite and finite AP buffer.

download (resp. upload) STA active, and any service from a STA (success or discard) makes it inactive since its MAC queue becomes empty.

Assumptions (A2) and (A3) hold in practice due to the following reasons. Since the TCP connections are long-lived, the TCP sessions are mostly in the congestion avoidance phase with approximately every received ACK bringing one new TCP packet from the TCP sender. Due to this flow control of TCP, an upload (resp. download) STA's MAC queue receives a TCP DATA packet (resp. TCP ACK) only after the AP has successfully transmitted a TCP ACK (resp. TCP DATA packet) to it. Indeed, the AP has to serve multiple STAs but, under DCF, has equal opportunity of channel access. Thus, the TCP ACK (resp. DATA) packets belonging to the upload (resp. download) STAs are mostly backlogged at the bottleneck AP buffer and (A2) holds. Only few packets remain in STAs' MAC queues and since the AP serves packets to the STAs in a fair manner, each active STA mostly has one packet in its MAC queue and (A3) holds.

Due to (A3), the number of active STAs in the WLAN changes whenever there is a successful service by the AP or there is any service from a STA (either successful service or discard).

Our approach to developing the analytical model is modular where we divide the analysis of different aspects of the system behaviour into different modules. The components of our overall analytical model and their inter-dependencies are shown in Fig. 3. The analytical model for each module is developed by assuming that the output quantities from other modules are known. The notation used in the remainder of the paper is summarized in Table 2.

TABLE 2
Notations

Notations	Description
1. Given inputs	
N_u	Total number of uploading STAs
N_d	Total number of downloading STAs
K	Maximum number of (re)transmissions per packet, $K = 7$
M	Maximum number of failures up to which the backoff window is doubled, $M = 5$
p_w	Probability that a packet transmission fails due to wireless error.
$W_{\max}$	Maximum TCP receive window
B	AP buffer size (in packets)
r	Weight of fairness, desired ratio of aggregate download throughput to upload throughput
2. Internal variables	
$\pi(d, u)$	Stationary probability that there are u active uploading and d active downloading STAs
P_s^u	Probability that the success belongs to upload STAs given that it is a success slot
$P_s^{\mathrm{Ap,Data}}$	Probability that the success belongs to the AP given that it is a success slot with AP holding TCP DATA at the HOL position
p_{lu}	Probability that a TCP DATA packet is discarded by an upload station
p_{ld}	Probability that a TCP DATA packet is discarded by the AP
p_d	Probability that a TCP DATA packet is discarded
h	Fraction of services by the AP that are TCP DATA packets; also equal to the probability that the HOL packet at the AP is a TCP DATA packet
$E[C]$	Expected number of failures in a renewal cycle
$E[A]$	Expected number of attempts in a renewal cycle
$E[W_d]$	Expected congestion window for a download connection
$E[W_u]$	Expected congestion window for an upload connection
$\bar{\Theta}_u$	Unrestricted aggregate TCP throughput of upload connections
$\bar{\Theta}_d$	Unrestricted aggregate TCP throughput of download connections
$E_{(d,u)}[X]$	Expected duration of a cycle
γ_u	Probability of transmission failure for an upload station
$\gamma_{\mathrm{Ap}}^{\mathrm{Data}}$	Probability of transmission failure for the AP
3. Outputs	
p_b	Probability of buffer overflow at the AP or the probability that a packet arriving to the AP is blocked
B_r	Desired AP buffer size (in packets)
Θ_u	Aggregate TCP Throughput of upload connections (in packets/second)
Θ_d	Aggregate TCP Throughput of download connections (in packets/second)

Fig. 3 is explained as follows. Given the fraction $h, 0 \leq h \leq 1$ of services from the AP that are TCP DATA packets and the discard probability of TCP DATA packets of the downloading connections, p_{ld}, we develop a Markov chain to track the number of active download and upload STAs, d, u (Section 3.1.1). The stationary probability $\pi(d, u)$, obtained from the Markovian analysis is applied to a renewal framework for updating the discard probabilities,

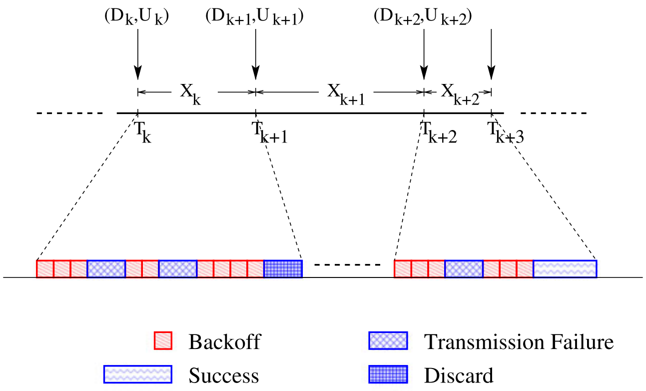

Fig. 4. Evolution of network activity in the WLAN. The time instants immediately after the completion of a service have been marked by arrows. The time between two consecutive arrows is referred to as a *cycle*. A cycle terminates with either a successful transmission or a discard.

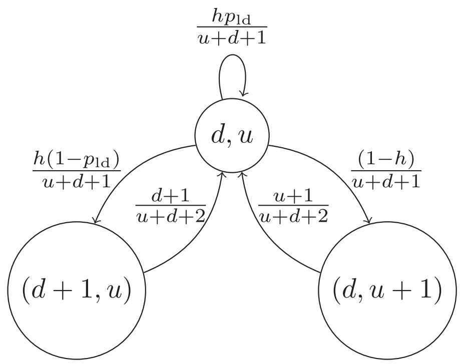

Fig. 5. State Transistion Diagram of $\{(D_k, U_k)\}$.

$p_{\rm lu}, p_{\rm ld}$ (Section 3.1.2), and throughputs $\bar{\Theta}_u, \bar{\Theta}_d$, with unrestricted TCP receiver window (Section 3.1.3). A saturation analysis similar to [31], but extended with wireless channel errors, provides the success probability, P_s, the expected values of attempts, $E[A]$, collisions, $E[C]$, and renewal cycle duration, $E[X]$, for each possible number of active upload and download STAs (Section 3.1.4). The throughputs, combined with the discard probabilities, provides the new estimate of h and thus completes the fixed point (Section 3.1.5). The converged values of throughputs and the discard probabilities are applied to an analysis that captures the impact of receiver window, $W_{\max}$, and provides the actual attainable throughputs, Θ_d, Θ_u (Section 3.1.6). The aforementioned analysis is expanded to encompass the impact of buffer loss probability, p_b, at the bottleneck AP buffer using the bandwidth delay product of WLAN (Section 3.2). In the following, we explain each of the blocks in detail.

3.1 Impact of Wireless Channel Errors

We begin our analysis for an infinite AP buffer and then extend to the finite buffer case (Section 3.2).

3.1.1 Tracking the Number of Active STAs

Consider the sequence of time instants T_k, $k = 0, 1, 2, \dots$, with $T_0 = 0$ and where T_k denotes the time instant immediately after the completion of the kth service *(success or discard)* in the WLAN. In Fig. 4, these time instants have been marked by arrows. We consider a process $\{(D_k, U_k), k \geq 0\}$ embedded at T_k, $k \geq 0$, where D_k and U_k represent the number of active downloading and uploading STAs, respectively, at T_k. Recalling that the AP is saturated and always contending, $(D_k, U_k) = (d, u)$ implies that there are $u + d + 1$ nodes contending for channel access (the 1 corresponds to the AP). Let h denote the probability that the HOL packet at the AP's MAC queue is a TCP DATA packet (belonging to a download connection). Let $p_{\rm ld}$ denote the probability that the service of a TCP DATA packet at the AP is a discard. Given h and $p_{\rm ld}$, the process $\{(D_k, U_k), k \geq 0\}$ is a discrete time Markov chain (DTMC) with state space $\mathcal{S} = \{(d, u) : d \geq 0, u \geq 0\}$. The transition structure of this Markov chain is shown in Fig. 5.

Remark 3.1. The number of active download and upload STAs must satisfy $d \leq N_d$ and $u \leq N_u$. However, the state space of the Markov chain $\{(D_k, U_k), k \geq 0\}$ is unbounded due to our assumption (A3) that every successful transmission by the AP activates a new STA. This simplification does not compromise the accuracy of the results.

Using the short-hand notation

$$\begin{aligned}&\Pr((d', u')/(d, u))\\&\quad := \Pr(D_{k+1} = d', U_{k+1} = u' | (D_k = d, U_k = u)),\end{aligned}$$

the transition probabilities of $\{(D_k, U_k), k \geq 0\}$ are given as follows (see Fig. 5):

$$\begin{aligned}&\Pr(\text{upload station's service})\\&\quad = \Pr((d, u-1)/(d, u)) = \frac{u}{u+d+1},\end{aligned} \tag{1}$$

$$\begin{aligned}&\Pr(\text{download station's service})\\&\quad = \Pr((d-1, u)/(d, u)) = \frac{d}{u+d+1},\end{aligned} \tag{2}$$

$$\begin{aligned}&\Pr(\text{AP'ssuccessful service of DATA})\\&\quad = \Pr((d+1, u)/(d, u)) = \frac{h(1-p_{\rm ld})}{u+d+1},\end{aligned} \tag{3}$$

$$\begin{aligned}&\Pr(\text{AP's unsuccessful service of DATA})\\&\quad = \Pr((d, u)/(d, u)) = \frac{hp_{\rm ld}}{u+d+1},\end{aligned} \tag{4}$$

$$\begin{aligned}&\Pr(\text{AP's service of ACK})\\&\quad = \Pr((d, u+1)/(d, u)) = \frac{(1-h)}{u+d+1}.\end{aligned} \tag{5}$$

By (A3), a service (successful or discard) from an uploading STA causes the number of active uploading STAs to decrease by one (Eq. (1)) and similarly for downloading STAs (Eq. (2)). However, only a successful service of a TCP DATA packet from the AP increases the number of active downloading STAs by one (Eq. (3)). Due to (A1), the probability of an unsuccessful service of an ACK by the AP is negligible and approximately every service of a TCP ACK from the AP increases the number of uploading STAs by one (Eq. (5)).

Solving the balance equations of the Markov chain $\{(D_k, U_k)\}$ to obtain the stationary distribution $\pi(d, u)$,

$d, u \geq 0$, is simple because it satisfies the 'Kolmogorov criterion for reversibility'. Solving the detailed balance equations together with normalization leads to

$$\pi(d,u) = \frac{u+d+1}{(2-hp_{\mathrm{ld}})e^{1-hp_{\mathrm{ld}}}} \times \frac{(h(1-p_{\mathrm{ld}}))^d(1-h)^u}{d!u!}. \tag{6}$$

The average number of active download and upload STAs is

$$E[D] = \frac{h(1-p_{\mathrm{ld}})(3-hp_{\mathrm{ld}})}{(2-hp_{\mathrm{ld}})}, \tag{7}$$

and

$$E[U] = \frac{(1-h)(3-hp_{\mathrm{ld}})}{(2-hp_{\mathrm{ld}})} \tag{8}$$

respectively. Then, the average number of active contending STAs is given by

$$E[D]+E[U] = \frac{(1-hp_{\mathrm{ld}})(3-hp_{\mathrm{ld}})}{(2-hp_{\mathrm{ld}})} \leq \frac{3}{2}; \quad 0 \leq h, p_{\mathrm{ld}} \leq 1.$$

Observe from the above expression that the average number of active STAs attains its maximum, i.e. $\frac{3}{2}$, with zero wireless channel error. As MAC discards due to wireless channel errors, p_{ld}, increases (resp. decreases), the average number of active STAs decreases (resp. increases).

3.1.2 Failure and Discard Analysis

Fig. 4 depicts the evolution of network activity in the WLAN. The time between two consecutive service completions in the WLAN is referred to as a *cycle*. The kth cycle corresponds to the time interval $[T_k, T_{k+1})$. A cycle consists of a sequence of so-called *slots*. There are three types of slots, namely, a backoff or idle slot, a transmission failure due to collisions and/or channel error, and a successful transmission. A cycle terminates with either a successful transmission or a discard, where a discard is a special case of transmission failure.

Let $A_{k,u}$ and $C_{k,u}$ represent the total number of (transmission) attempts and failures (due to collision and/or channel error), respectively, by the upload STAs up to the end of the kth cycle. Applying the *renewal reward theorem*, the (conditional) failure probability as seen by the uploading STAs, γ_u, can be obtained by

$$\gamma_u := \lim_{k\to\infty} \frac{C_{k,u}}{A_{k,u}} \approx \frac{\sum_{u=0}^{N_u}\sum_{d=0}^{N_d}\pi(d,u)E^u_{(d,u)}[C]}{\sum_{u=0}^{N_u}\sum_{d=0}^{N_d}\pi(d,u)E^u_{(d,u)}[A]}, \tag{9}$$

where $E^u_{(d,u)}[A]$ and $E^u_{(d,u)}[C]$ denote the expected number of attempts and failures, respectively, per cycle by the uploading STAs given that there are u upload STAs and d download STAs active.

Similarly, the (conditional) failure probability as seen by the AP when attempting a TCP DATA packet, $\gamma^{\mathrm{Data}}_{\mathrm{Ap}}$, is given by

$$\gamma^{\mathrm{Data}}_{\mathrm{Ap}} \approx \frac{\sum_{u=0}^{N_u}\sum_{d=0}^{N_d}\pi(d,u)E^{\mathrm{Ap,Data}}_{(d,u)}[C]}{\sum_{u=0}^{N_u}\sum_{d=0}^{N_d}\pi(d,u)E^{\mathrm{Ap,Data}}_{(d,u)}[A]}, \tag{10}$$

where $E^{\mathrm{Ap,Data}}_{(d,u)}[A]$ and $E^{\mathrm{Ap,Data}}_{(d,u)}[C]$ denote the expected number of attempts and failures, respectively, per cycle for the AP restricted to TCP DATA packets given that there are u upload STAs and d download STAs active.

The corresponding discard probabilities are given by

$$p_{\mathrm{lu}} = (\gamma_u)^{K+1} \qquad \text{and} \qquad p_{\mathrm{ld}} = (\gamma^{\mathrm{Data}}_{\mathrm{Ap}})^{K+1}, \tag{11}$$

where recall that K denotes the maximum number of retransmissions allowed before discard. Eqs. (9)-(11) form a submodel to estimate MAC discards directly.

Remark 3.2. Eqs. (9) and (10) are approximations because the sum is taken only for $0 \leq u \leq N_u$ and $0 \leq d \leq N_d$, even though the state space of the Markov chain $\{(D_k, U_k), k \geq 0\}$ is unbounded. However, the probability mass lying outside the set $\{(d,u) : 0 \leq u \leq N_u, 0 \leq d \leq N_d\}$ is indeed negligible and the approximations in (9) and (10) lead to accurate results as shown in Section 4.1.

3.1.3 Computation of Unrestricted Throughput

In this section, we assume that the maximum receiver window is unrestricted and the TCP congestion window can grow without bound. Let $H_{k,u}$ denote the total number of TCP DATA packets successfully transmitted by the upload connections up to the end of the kth cycle. Then the aggregate unrestricted upload throughput, $\bar{\Theta}_u$, is given by

$$\bar{\Theta}_u = \lim_{k\to\infty}\frac{H_{k,u}}{T_{k+1}} \approx \frac{\sum_{u=0}^{N_u}\sum_{d=0}^{N_d}\pi(d,u)P^u_s(u,d)}{\sum_{u=0}^{N_u}\sum_{d=0}^{N_d}\pi(d,u)E_{(d,u)}[X]}, \tag{12}$$

where $E_{(d,u)}[X]$ denotes the expected duration of a cycle and $P^u_s(u,d)$ denotes the probability that the cycle terminates due to a successful transmission by an upload STA given that there are u upload STAs and d download STAs active. Similarly, the unrestricted download throughput, $\bar{\Theta}_d$, is given by

$$\bar{\Theta}_d \approx \frac{\sum_{u=0}^{N_u}\sum_{d=0}^{N_d}\pi(d,u)P^d_s(u,d)}{\sum_{u=0}^{N_u}\sum_{d=0}^{N_d}\pi(d,u)E_{(d,u)}[X]}, \tag{13}$$

where $P^d_s(u,d)$ denotes the probability that the cycle terminates due to a success by a download STA given that there are u upload STAs and d download STAs are active.

3.1.4 Saturation Contention Analysis

We apply a renewal approach to compute the expectations $E^u_{(d,u)}[A]$, $E^u_{(d,u)}[C]$, $E^{\mathrm{Ap,Data}}_{(d,u)}[A]$, $E^{\mathrm{Ap,Data}}_{(d,u)}[C]$, $E_{(d,u)}[X]$, and the probabilities $P^u_s(u,d)$ and $P^d_s(u,d)$. Here, we assume that

A4. A service in the WLAN is discard with a (network-wide average) discard probability p_d and successful with the complementary probability $1-p_d$.

Due to (A4), between two consecutive successful services there are on average $(1-p_d)^{-1}$ services of any kind. We call the time between two successful services a *mega cycle*. We compute the expectations $E^u_{(d,u)}[A]$, $E^u_{(d,u)}[C]$, $E^{\mathrm{Ap,Data}}_{(d,u)}[A]$, $E^{\mathrm{Ap,Data}}_{(d,u)}[C]$, $E_{(d,u)}[X]$, and the probabilities $P^u_s(u,d)$ and $P^d_s(u,d)$ by first computing the corresponding quantities over a mega cycle coupled through fixed point equations as in Bianchi [31], [32], and then multiply each of them by the factor $1 - p_d$. The detailed derivations can be found in Appendix A (see supplementary material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TMC.2015.2456883).

Remark 3.3. Our model can be extended to handle rate *heterogeneity* among the nodes. The computation of $E_{(d,u)}[X]$ can be modified accordingly (see Appendix A, available in the online supplemental material).

3.1.5 Computing h

Let $\lambda_{\text{Ap}}^{\text{Data}}$ and $\lambda_{\text{Ap}}^{\text{Ack}}$ denote the arrival rate of TCP DATA packets and TCP ACKs, respectively, into the AP buffer. All TCP flows originate or terminate at AP. Then, in the steady state, the probability that the HOL packet is a TCP DATA packet is also equal to the ratio of the arrival rate of TCP DATA packets to the total arrival rate into the AP buffer, i.e.,

$$h = \frac{\lambda_{\text{Ap}}^{\text{Data}}}{\lambda_{\text{Ap}}^{\text{Data}} + \lambda_{\text{Ap}}^{\text{Ack}}} = \frac{\bar{\Theta}_d/(1 - p_{\text{ld}})}{\bar{\Theta}_d/(1 - p_{\text{ld}}) + \bar{\Theta}_u}. \tag{14}$$

Eq. (14) provides the relationship between the fraction of TCP DATA packets in the AP buffer, h, and the probability of MAC discards, p_{ld} (i.e., the probability of loss due to combined effect of wireless channel errors and MAC collisions). As h increases with increase in wireless channel error, TCP download connections obtain a higher share of throughput compared to the upload connections. Thus, Eq. (14) gives insights into TCP throughput unfairness due to wireless channel errors.

Eqs. (6) and (10)-(14) create a fixed point formulation with infinite AP buffer (see Fig. 3) from which h and the unrestricted throughputs can be computed. Letting $\boldsymbol{x} = (h, p_{\text{ld}})$, we observe that, in each iteration n of our fixed point computation, we start at a point $\boldsymbol{x}_n$ and apply Eqs. (6) and (10)-(14) to obtain a new point $\boldsymbol{x}_{n+1}$. The existence of a fixed point for $\boldsymbol{x}$ is formally proven in Appendix B (see supplementary material, available online). In our extensive numerical experiments, we have observed that the iterations always converge to the same solution (for the same setting) irrespective of the starting point.

3.1.6 Actual Throughput Accounting for the Window Limit

The computation of unrestricted throughput in Section 3.1.3 is based on the premise that *a TCP source generates DATA packets in response to every received ACK*. If the receiver window is bounded, then the TCP congestion window cannot exceed a certain maximum value, $W_{\max}$, and the actual throughput will be limited by this maximum window. To account for such upper limits on the congestion window, we have the following:

Approximation 3.1. The maximum achievable throughput is directly proportional to the maximum congestion window, i.e., if the *expected unconstrained congestion window*, $E[W]$, satisfies $E[W] > W_{\max}$, then the maximum achievable throughput of a connection, $\theta_{\max}$, is obtained from the unrestricted throughput, $\bar{\theta}$, as

$$\theta_{\max} \approx \frac{W_{\max}}{E[W]}\bar{\theta}, \tag{15}$$

where $E[W]$ is a known function of the packet loss probability, p. We denote the dependence of $E[W]$ on p by

$$E[W] = f(p). \tag{16}$$

Remark 3.4. Eq. (16) defines the relationship between expected congestion window, $E[W]$, and the TCP-level packet loss probability, p. For a loss-based TCP such as TCP Reno, NewReno and SACK (as opposed to a delay-based TCP such as TCP Vegas), the expected congestion window, $E[W]$, will be given by a function $f(p)$ of the TCP-level packet loss probability, p. For example, for TCP Reno, the function $f(p)$ is given by [34, Eq. (13)]. With i.i.d. channel errors, TCP New Reno behaves like TCP Reno [35] and the function $f(p)$ for TCP SACK is given by [36, Eq. (14)]. TCP Reno is taken as an example in the paper. But, any loss-based TCP for which the function $f(p)$ is known can be considered.

With N_u uploading and N_d downloading STAs, we have

$$\Theta_{\max}^u \approx \frac{W_{\max}}{E[W_u]}\bar{\Theta}_u \quad \text{and} \quad \Theta_{\max}^d \approx \frac{W_{\max}}{E[W_d]}\bar{\Theta}_d,$$

where $E[W_u]$ and $E[W_d]$, for TCP Reno, are given by

$$E[W_u] = f(p_{\text{lu}}) \approx \sqrt{\frac{8}{3p_{\text{lu}}}} \text{ and } E[W_d] = f(p_{\text{ld}}) \approx \sqrt{\frac{8}{3p_{\text{ld}}}}.$$

Here we note that if the expected congestion window of one download (resp. upload) connection exceeds the maximum window, then so are that of all other download (resp. upload) connections due to symmetry.

The general equations for the actual upload and download throughputs, Θ_u, Θ_d, accounting for the maximum congestion window, $W_{\max}$, are given by

$$\Theta_u = \min\{\Theta_{\max}^u, \bar{\Theta}_u\}, \quad \text{and} \tag{17}$$

$$\Theta_d = \min\{\Theta_{\max}^d, \bar{\Theta}_d\}. \tag{18}$$

3.2 With Buffer and Wireless Channel Errors

Consider the situation where the AP can store up to B packets. Our next objective is to obtain the probability of buffer overflow, p_b. The maximum number of packets in the network at any given point in time is given by the bandwidth delay product. Based on the BDP for a single TCP connection, the number of packets in the network circuit with unconstrained congestion window is given by (see [37, Eq. (2)])

$$BDP_{\text{in packets}}^{\text{Unconstrained}} \approx \frac{3}{4}E[W]. \tag{19}$$

Combining (19) with (16) and accounting for the maximum congestion window, $W_{\max}$ (see [34, Eq. (33)]), we obtain the maximum number of packets in our network as

TABLE 3
Parameter Setting

Parameters	Value
PHY rate	11 Mbps
Control rate	2 Mbps
PLCP premble	144 μs
Slot Time	20 μs
DIFS time	50 μs
SIFS time	10 μs
EIFS time	308 μs
Min. Contention Window	31
Max. Contention Window	1,023
Max. Retry limit	7
RTS/CTS	*disabled*
TCP version	*Reno*
Delayed Ack	*disabled*
TCP Header	20 bytes
IP Header	20 bytes
TCP ACK size	40 bytes
Data size	1,460 bytes

$$\begin{aligned} BDP^{\mathrm{WLAN}}_{\text{in packets}} &\approx N_u \min\left\{W_{\max}, \frac{3}{4}E[W_u]\right\} \\ &+ N_d \min\left\{W_{\max}, \frac{3}{4}E[W_d]\right\} \\ &\approx N_u \min\left\{W_{\max}, \frac{3}{4}f(p_{\mathrm{lu}})\right\} \\ &+ N_d \min\left\{W_{\max}, \frac{3}{4}f(p_b + (1-p_b)p_{\mathrm{ld}})\right\}, \end{aligned} \tag{20}$$

where we have ignored the impact of the loss of TCP ACKs belonging to upload connections due to buffer overflow because the loss of a few TCP ACKs does not adversely affect the upload throughput. It is because of the cumulative effect of TCP ACKs as noted earlier.

Since the propagation delay is negligible, the number of packets in flight at any point in time is negligible. The number of packets stored at the STAs is also negligible because there are approximately 1.5 STAs active. Thus, all of the $BDP^{\mathrm{WLAN}}_{\text{in packets}}$ packets are stored at the AP queue. Since TCP tends to fill up the available buffer, we approximate $BDP^{\mathrm{WLAN}}_{\text{in packets}}$ with B and obtain

$$\begin{aligned} B &\approx N_u \min\left\{W_{\max}, \frac{3}{4}f(p_{\mathrm{lu}})\right\} \\ &+ N_d \min\left\{W_{\max}, \frac{3}{4}f(p_b + (1-p_b)p_{\mathrm{ld}})\right\}. \end{aligned} \tag{21}$$

Eq. (21) can be solved to obtain p_b for a given value of B. This equation combines the impact of collisions, wireless channel errors and bottleneck buffer overflow into the BDP of Wi-Fi last-mile in terms of packets. Recalling that buffer overflows at the AP adversely affect the throughput of only download connections, we obtain h taking into account of bottleneck AP buffer overflow only as,

$$h = \frac{\bar{\Theta}_d(1-p_b)}{\bar{\Theta}_u + \bar{\Theta}_d(1-p_b)}. \tag{22}$$

h in Eq. (22) decreases with increase in probability of packet losses (p_b). Thus, by decreasing the AP buffer, competing TCP upload connections obtain a higher share of throughput than the download connections. Eq. (22) explains the well known throughput unfairness due to AP buffer overflows.

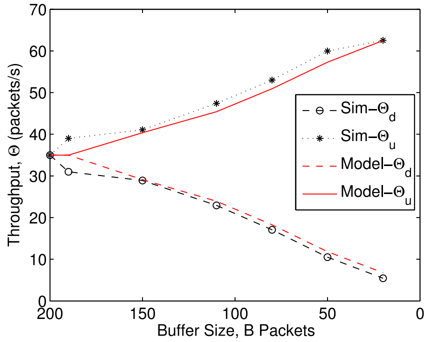

Fig. 6. Throughput unfairness due to buffer overflows with $N_u = 5$ and $N_d = 5$ and bottleneck AP buffer (B).

Finally, joint impact of bottleneck buffer overflow and packet discard probability due to MAC collisions and/or wireless channel errors can then be computed by combining Eqs. (14) and (22) as,

$$h = \frac{\bar{\Theta}_d(1-p_b)}{\bar{\Theta}_u(1-p_{\mathrm{ld}}) + \bar{\Theta}_d(1-p_b)}. \tag{23}$$

Eqs. (21), (23) correspond the dashed blocks of Fig. 3. These two equations together with Eqs. (6) and (9)-(13) create a fixed point formulation for joint impact of wireless channel errors with finite AP buffer. The existence of fixed point is addressed in Appendix B (see supplementary material, available online).

Magnitude of h in Eq. (23) increases with increase in p_{ld} and decreases with increase in p_b (and vice versa). Understanding Eq. (23) in terms of *joint impact of wireless channel errors and buffer losses*, we came to conclude that throughput unfairness due to wireless channel error is *opposite to the unfairness* due to bottleneck buffer overflows. Our observation are validated with numericals and simulations results in Section 4.

4 Results

In this section, we validate the accuracy of our analytical model and discuss the results. We created the network scenario depicted in Fig. 1. The parameter setting in the simulations is summarized in Table 3. We use the *Basic Access* mechanism and long-lived TCP Reno connections. We set the maximum segment size (MSS) of TCP to 1,460 bytes and maximum receiver window to $W_{\max} = 45$ packets (or 65,535 bytes).

4.1 Model Validation

4.1.1 Impact of Bottleneck Buffer Overflow

In Fig. 6, we observe that the throughput obtained by the uploading (resp. downloading) connections increases (resp. decreases) with decrease in the size of the AP buffer. Moreover, it does not have any adverse impact on the total throughput of the Wi-Fi system. Our fixed point model with h in Eq. (22) accurately captures this well-known throughput unfairness due to buffer overflows.

4.1.2 Impact of Wireless Channel Errors

For the infinite buffer case, the AP buffer is set to a very large value to prevent buffer overflows and results were

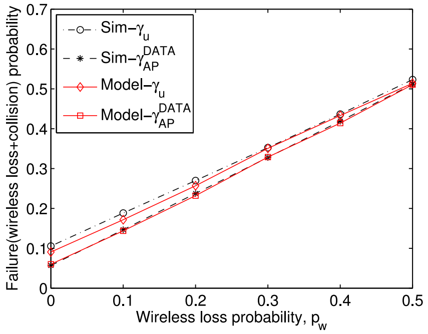

Fig. 7. Probability of transmission failure, γ with $N_u = 5$ and $N_d = 5$ and wireless channel error, p_w.

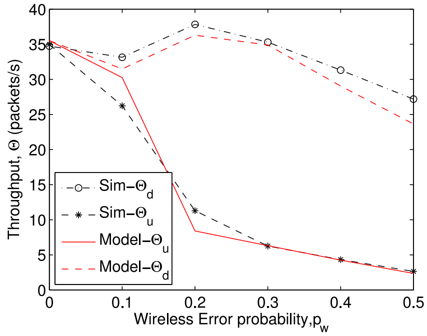

Fig. 9. Throughput unfairness due to wireless channel error, p_w, with $N_u = 5$ and $N_d = 5$ and without buffer overflows at bottleneck AP.

obtained with $N_u = N_d = 5$. Fig. 7 depicts the probability that an attempt to transmit a TCP DATA packet by an uploading station and an AP fails due to collision and/or channel error. Fig. 8 depicts the probability that a TCP DATA packet is discarded by an uploading STA and the AP. It can be observed that both γ_u and $\gamma_{\text{Ap}}^{\text{Data}}$ increase linearly and both p_{lu} and p_{ld} increase exponentially with the increase in channel error probability p_w; γ_u is slightly higher than $\gamma_{\text{Ap}}^{\text{Data}}$; p_{lu} is slightly higher than p_{ld}. Our analytical model accurately captures these facts.

Since the average number of so-called active STAs is equal to 1.5, the AP, on average, contends with 1.5 STAs. An active STA, however, contends with at least two nodes most of the time (1 contending AP and another STA). Hence, the (conditional) failure probability as seen by an upload STA is slightly larger than that seen by the AP, which causes the small difference between γ_u and $\gamma_{\text{Ap}}^{\text{Data}}$.

With small wireless channel errors (say $p_w \leq 0.1$), the small collision probabilities are masked by the $K+1$ transmissions in that the discard probabilities for both the upload and download connections are almost zero. With significant wireless channel errors, however, the $K = 7$ retransmissions are not enough to mask the transmission failures, and packets are lost due to MAC discards (Fig. 8).

The TCP throughputs obtained by uploading and downloading connections are as shown in Fig. 9. We observed that, the total throughput of the system, $\Theta_u + \Theta_d$, decreases with increase in wireless channel errors. Moreover, the throughput obtained by downloading connections is higher than that by uploading connections in the presence of wireless losses. This new type of unfairness that we discovered is in an opposite sense to the well-known unfairness depicted in Fig 6. Fig. 9 validates this unfairness with simulations for $N_u = 5$, $N_d = 5$ and $p_w \in [0, 0.5]$. The causes for this unfairness are discussed in Section 4.2.

4.1.3 Joint Impact of Wireless Errors and Buffer Losses

The accuracy of our analytical model for throughput unfairness due to joint impact of wireless channel errors and finite AP buffer overflows is depicted in Fig. 10. We observe that the throughput obtained by the uploading (resp. downloading) connections increases (resp. decreases) with decrease in the size of the AP buffer in the presence of wireless channel errors as well. This gives us an important insight that the losses due to buffer overflows and wireless errors are independent of each other. Note that our model is less accurate at smaller buffers. With small buffers, several TCP ACKs can be dropped in succession due to buffer overflow, and upload connections also suffer. However, it is quite challenging to capture this effect analytically.

4.2 Causes of Throughput Unfairness

In this section we explain the causes of the new type of unfairness caused by wireless channel errors considering a large AP buffer.

The difference in the failure (collision and wireless error) probability of TCP DATA packets transmitted by the AP and that by an uploading STA (Fig. 7) leads to a difference in the MAC discard probabilities, $p_{\text{lu}}, p_{\text{ld}}$ (Fig. 8). As a consequence, the TCP congestion windows of upload connections suffer more frequent window decreases than that of download connections (see Fig. 11). Note that TCP is designed to be highly sensitive to losses. The ratio of losses $\frac{p_{\text{ld}}}{p_{\text{lu}}}$ is amplified to a ratio $\sqrt{\frac{p_{\text{lu}}}{p_{\text{ld}}}}$ of congestion windows (due to the

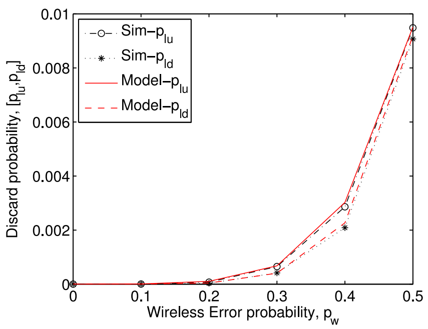

Fig. 8. Discard Probability, p_l with $N_u = 5$ and $N_d = 5$ and wireless channel error, p_w.

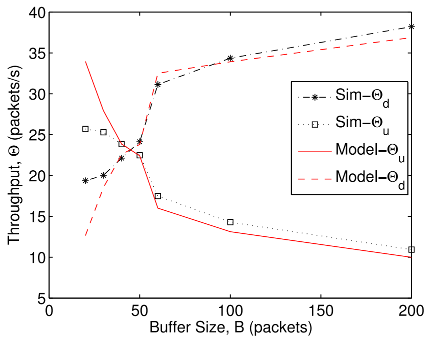

Fig. 10. Analytical and simulation results depicting throughput unfairness with $N_u = 5$ and $N_d = 5$ and joint impact of wireless error, p_w = 0.2 and AP buffer size, B.

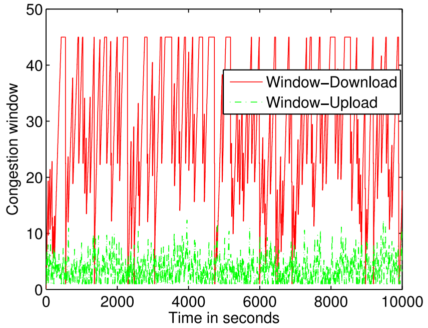

Fig. 11. Evolution of the congestion windows of upload and download connections with $W_{\max} = 45$ and $p_w = 0.3$ as observed in NS-2.

well-known inverse square root relationship). This difference in the congestion windows leads to a larger number of TCP DATA packets belonging to download connections than TCP ACKs belonging to upload connections in the AP buffer. The TCP ACKs of the upload connections are essentially *blocked* by a large number of TCP DATA packets of the download connections. Due to the closed-loop nature of TCP, this blocking and the higher loss rate of the upload STAs further slow down the arrival of TCP ACKs into the AP buffer. This creates a positive feedback loop in the bottleneck AP buffer such that the fractions of the packets that are TCP DATA packets belonging to the download connections keeps growing. This positive loop is restricted to a certain extent by the discards of DATA packets belonging to the download connections. However, the primary factor that restricts this positive feedback loop is the receiver window limit, $W_{\max}$. Therefore, with a smaller window limit of $W_{\max} = 20$, the download connections' windows cannot grow above 20 and the degree of unfairness (difference in upload and download throughputs) is smaller (see Fig. 12). Since the root cause of the unfairness is the difference between the discard probabilities, $p_{\rm lu}, p_{\rm ld}$, with a smaller retransmit limit of $K = 4$, the difference between $p_{\rm lu}$ and $p_{\rm ld}$ becomes significant at a smaller probability of wireless channel error, p_w, and the throughput unfairness becomes appreciable at a smaller, p_w (see Fig. 13).

5 Design of Solution to Achieve Fairness

Solutions for TCP throughput unfairness have been studied with seperate queuing at AP [24], tuning the MAC layer parameters [23] and by prioritizing the channel access [25].

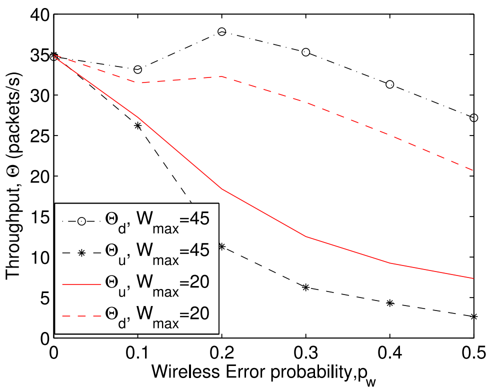

Fig. 12. Scaling of throughput unfairness with receiver window, $W_{\max}$, with infinite bottleneck AP buffer as observed in NS-2 simulations.

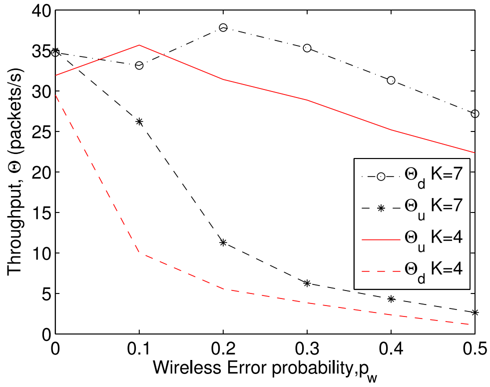

Fig. 13. Translation of throughput unfairness with smaller retransmission, K, with infinite AP buffer as observed in NS-2 simulations.

However, our approach to achieving fairness is to make one type of throughput unfairness work against (i.e. compensate) the other without modifying the default parameters of the TCP protocol and 802.11 standards.

If the wireless error probability was zero, the throughput unfairness due to buffer overflows could have been balanced by simply increasing the buffer capacity at the AP. However, since wireless channel errors are unavoidable in real networks, simply increasing the AP buffer will not work. Indeed, increasing the AP buffer beyond a certain value may lead to a higher share of throughput to the downloading STAs.

Our first key Insight is that

I$_1$ *The AP buffer should be dynamically adjusted in such a way that the adverse impact of buffer losses on downloads would balance out the adverse impact of wireless channel errors on uploads resulting in balanced throughputs.*

Our second key Insight is that

I$_2$ *As long as the AP buffer is the bottleneck, such tuning of the AP buffer will only affect the division of throughput between uploading and downloading connections, and not lead to the reduction of the total throughput of the system (see Figs. 6 and 10).*

For realistic round-trip propagation delays and at least two STAs in the WLAN, the AP buffer is indeed the bottleneck and remains saturated [33].

We apply the insights from our joint analysis to tune the AP buffer so that a desired ratio of download to upload throughput r (desired degree of throughput fairness), may be achieved. Rearranging (23) provides,

$$r := \frac{\bar{\Theta}_d}{\bar{\Theta}_u} = \frac{h}{1-h}\frac{1-p_{\rm ld}}{1-p_b}. \tag{24}$$

As stated in $\mathbf{C}_3$, to achieve a specific value of the ratio of throughputs, r, rather than fixing the AP buffer size B a priori, we apply (24) as the terminating condition when solving the fixed point equations and obtain the value of $p_b(r)$ that corresponds to the desired value of r. Then, the desired buffer size $B_r := B(p_b(r))$ corresponding to $p_b(r)$ can be obtained by applying (21). In particular, the upload and download throughputs can be made equal by setting $r = 1$.

As illustrated in Tables 4 and 5, given the wireless channel error probability, p_w, and number of uploading and

TABLE 4
B_r and $p_b(r)$ for $r = 1$ with $N_u = N_d = 5$ and $N_u = N_d = 1$ for Different Channel Error Probabilities, p_w

p_w	$p_b(r)$	$B_r, N_u = N_d = 5$	$B_r, N_u = N_d = 1$
0.2	0.068409	46.64492	9.328984
0.3	0.178994	28.80515672	5.761031344
0.4	0.285748	22.7361712	4.54723424
0.5	0.411432509	18.81632414	3.763264828

TABLE 5
B_r and $p_b(r)$ for $r = 2$ with $N_u = N_d = 5$ for Different Channel Error Probabilities, p_w

p_w	$p_b(r)$	B_r
0.2	0.038079765	62.6389163
0.3	0.091319571	40.3717802
0.4	0.152956007	31.1943946
0.5	0.232144981	25.32096082

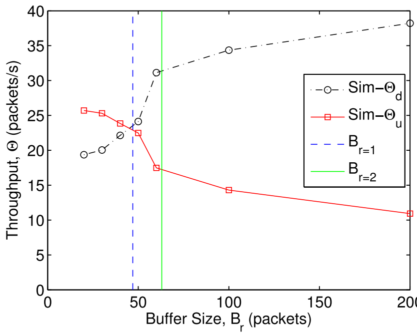

Fig. 14. Buffer Sizing with $p_w = 0.2$ and $N_u = N_d = 5$.

downloading connections, corresponding bottleneck AP Buffer size in packets (B_r) and the blocking probability $p_b(r)$, for any desired value of r can be obtained by solving the fixed point equations. The existence of fixed point is addressed in Appendix B (see supplementary material, available online).

In fact, the desired throughput ratio can be achieved by adopting one of the two different approaches at the AP:

1. *Buffer sizing*. Set the AP buffer to $B_r := B(p_b(r))$.
2. *Admission control*. Admit each arriving packet into the AP buffer with probability $1 - p_b(r)$, or blocking probability $p_b(r)$.

However, there exists a trade-off between buffer sizing and admission control. TCP works well in congestion avoidance only if the AP buffer can store at least six packets per connection [38] so that it does not go for frequent timeouts. Therefore, we propose to apply the buffer sizing approach only if the predicted value of B_r satisfies $B_r \geq 6N$, where $N = N_u + N_d$ denotes the total number of TCP connections sharing the AP buffer.

Furthermore, the desired ratios of throughput can also always be achieved by the admission control approach. However, when the blocking probability is relatively high, this may lead to underutilizing the available link capacity, thus decreasing the total throughput of the system. So, we propose to apply admission control only when the predicted value of B_r satisfies $B_r < 6N$.

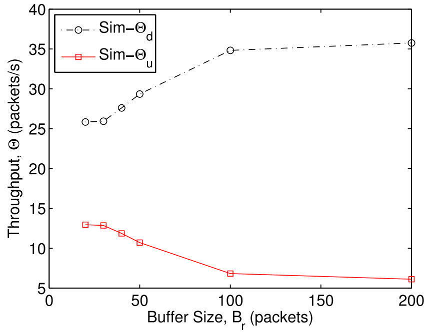

Fig. 15. Buffer sizing with $p_w = 0.3$ and $N_u = N_d = 5$.

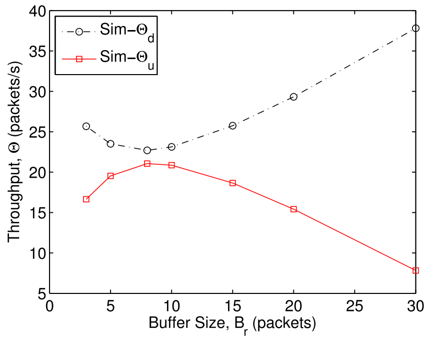

Fig. 16. Buffer sizing with $p_w = 0.3$ and $N_u = N_d = 1$.

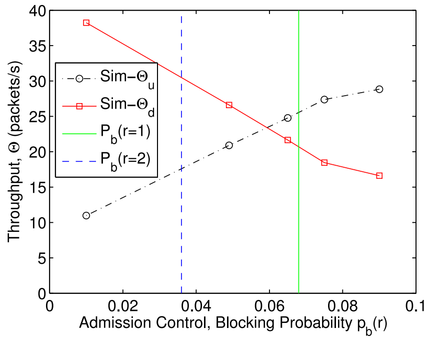

Fig. 17. Admission control with $p_w = 0.2$ and $N_u = N_d = 5$.

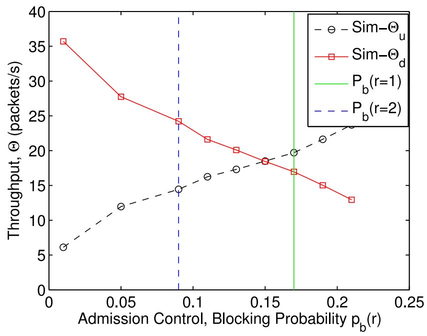

Fig. 18. Admission control with $p_w = 0.3$ and $N_u = N_d = 5$.

5.1 Fairness with Buffer Sizing

Figs. 14 and 15 depict the upload and download throughputs obtained by NS-2 simulations for $p_w = 0.2$ and $p_w = 0.3$, respectively, when we set the AP buffer to B_r as shown in Tables 4 and 5, respectively. For $p_w = 0.2$ and $N_u = N_d = 5$, Table 4 shows that $r = 1$ is achieved with

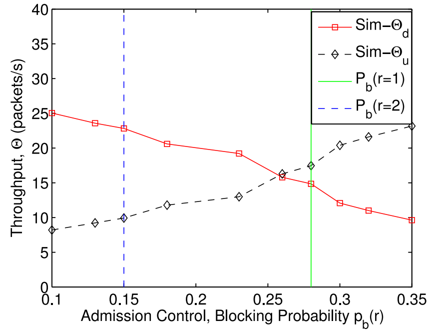

Fig. 19. Admission control with $p_w = 0.4$ and $N_u = N_d = 5$.

$B_r \approx 47$ packets, and Table 5 shows that $r = 2$ is achieved with $B_r \approx 63$ packets. These values of B_r have been marked in Fig. 14. It can be observed in Fig. 14 that the desired throughput ratios $r = 1$ and $r = 2$ are indeed achieved at the computed values of B_r. The difference is due to 1.5 packets stored at the STAs.

For $p_w = 0.3$ and $N_u = N_d = 5$, Table 4 shows that $r = 1$ is achieved with $B_r \approx 29$ packets. As $B_r < 6N$, it can be observed in Fig. 15 that the actual throughput ratio with $B_r \approx 29$ packets is not equal to 1. However, the difference in throughputs is indeed minimized at the computed value of B_r in Figs. 15 and 16.

For any desired throughput ratio, $r = \frac{\Theta_d}{\Theta_u}$, our solution recommends an AP buffer size, $B_r(p_w)$, corresponding to the probability of transmission failure due to wireless channel error, p_w. The B_r recommended by our solution decreases with increase in p_w. Observe that our solution recommends smaller AP buffer to combat the impact of higher channel errors, restricting the delay variation at the AP.

5.2 Fairness with Admission Control

Tables 4 and 5 provide an admission control parameter (in terms of blocking probability, $p_b(r)$ for $r = 1$ and $r = 2$, respectively), with different channel error probabilities, p_w.

Figs. 17, 18, and 19 depict the upload and download throughputs obtained by NS-2 simulations for $p_w = 0.2,\ 0.3,\ 0.4$, respectively. For $p_w = 0.2$ and $N_u = N_d = 5$, Table 4 shows that $r = 1$ is achieved with $p_b(r = 1) \approx 0.068$, and Table 5 shows that $r = 2$ is achieved with $p_b(r = 2) \approx 0.038$ packets. These values of $p_b(r)$ have been marked in Fig. 17. It can be observed in Fig. 17 that the desired throughput ratios $r = 1$ and $r = 2$ are indeed achieved at the computed values of $p_b(r)$. The small difference is due to an average of 1.5 packets at the STAs. Similarly, Figs. 18 and 19 also validate our results with simulation runs for $p_w = 0.3,\ 0.4$ respectively.

6 Conclusions

We have developed a comprehensive analytical model for TCP flow and congestion controls over Wi-Fi by examining the combined impact of bottleneck buffer losses, MAC collisions and wireless channel errors. The model provides means to accurately obtain both the upload and download throughput of long-live TCP connections in an infrastructure Wi-Fi WLAN taking into account the joint impact of possible wireless and buffer losses while fully capturing the dynamics of TCP protocol and the behaviour of the MAC access layer.

Based on our model, we have discovered and explained a new type of long-lived TCP throughput unfairness in Wi-Fi due to wireless channel errors. This unfairness is opposite to the well-known unfairness due to the bottleneck buffer overflow at the AP. Finally, we provided two simple solutions to achieve any desired ratio of download to upload throughput in these networks.

Acknowledgments

This work was supported by the Australian Research Council (ARC) Future Fellowships FT120100723 and Discovery Project DP130100156 grants.

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (IoT): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.
[2] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std. 802.11-2007, Jun. 2007.
[3] K.-C. Leung and V. O. Li, "Transmission control protocol (TCP) in wireless networks: Issues, approaches, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 8, no. 4, pp. 64–79, 4th Quarter, 2006.
[4] A. Al Hanbali, E. Altman, and P. Nain, "A survey of TCP over ad hoc networks," *IEEE Commun. Surveys Tuts.*, vol. 7, no. 1–4, pp. 22–36, 3rd Quarter 2005.
[5] B. Sardar and D. Saha, "A survey of TCP enhancements for last-hop wireless networks," *IEEE Commun. Surveys Tuts.*, vol. 8, no. 3, pp. 20–34, 3rd Quarter 2006.
[6] A. Kumar, "Comparative performance analysis of versions of TCP in a local network with a lossy link," *IEEE/ACM Trans. Netw.*, vol. 6, no. 4, pp. 485–498, Aug. 1998.
[7] E. Altman, K. Avrachenkov, and C. Barakat, "A stochastic model of TCP/IP with stationary random losses," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 30, no. 4, pp. 231–242, 2000.
[8] M. Zorzi, A. Chockalingam, and R. R. Rao, "Throughput analysis of TCP on channels with memory," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 7, pp. 1289–1300, Jul. 2000.
[9] F. Anjum and L. Tassiulas, "Comparative study of various TCP versions over a wireless link with correlated losses," *IEEE/ACM Trans. Netw.*, vol. 11, no. 3, pp. 370–383, Jun. 2003.
[10] A. A. Abouzeid, S. Roy, and M. Azizoglu, "Comprehensive performance analysis of a TCP session over a wireless fading link with queueing," *IEEE Trans. Wireless Commun.*, vol. 2, no. 2, pp. 344–356, Mar. 2003.
[11] M. Panda, H. Vu, M. Mandjes, and S. Pokhrel, "Performance analysis of TCP NewReno over a cellular Last-Mile: Buffer and channel losses," *IEEE Trans. Mobile Comput.*, vol. 14, no. 8, pp. 1629–1643, Aug. 1, 2015.
[12] D. Miorandi, A. A. Kherani, and E. Altman, "A queueing model for HTTP traffic over IEEE 802.11 WLANs," *Comput. Netw.*, vol. 50, no. 1, pp. 63–79, 2006.
[13] T. Sakurai and S. Hanly, "Modelling TCP flows over an 802.11 wireless LAN," in *Proc. 11th Eur. Wireless Conf. 2005-Next Gener. Wireless Mobile Commun. Services*, 2005, pp. 1–7.
[14] G. Kuriakose, S. Harsha, A. Kumar, and V. Sharma, "Analytical models for capacity estimation of IEEE 802.11 WLANs using DCF for internet applications," *Wireless Netw.*, vol. 15, no. 2, pp. 259–277, 2009.
[15] J. Yu and S. Choi, "Modeling and analysis of TCP dynamics over IEEE 802.11 WLAN," in *Proc. 4th Annu. Conf. Wireless Demand Netw. Syst. Services*, 2007, pp. 154–161.
[16] R. Bruno, M. Conti, and E. Gregori, "Throughput analysis and measurements in IEEE 802.11 WLANs with TCP and UDP traffic flows," *IEEE Trans. Mobile Comput.*, vol. 7, no. 2, pp. 171–186, Feb. 2008.
[17] S. H. Nguyen, L. L. Andrew, and H. Le Vu, "Performance of multi-channel IEEE 802.11 WLANs with bidirectional flow control," in *Proc. IEEE 38th Conf. Local Comput. Netw.*, 2013, pp. 755–758.
[18] R. Bruno, M. Conti, and E. Gregori, "An accurate closed-form formula for the throughput of long-lived TCP connections in IEEE 802.11 WLANs," *Comput. Netw.*, vol. 52, no. 1, pp. 199–212, 2008.

[19] F. Keceli, I. Inan, and E. Ayanoglu, "Fair and efficient transmission control protocol access in the IEEE 802.11 infrastructure basic service set," *Wireless Commun. Mobile Comput.*, vol. 15, pp. 1376–1390, 2015.

[20] S. Pilosof, R. Ramjee, D. Raz, Y. Shavitt, and P. Sinha, "Understanding TCP fairness over wireless LAN," in *Proc. 22nd Annu. Joint Conf. IEEE Comput. Commun.*, 2003, vol. 2, pp. 863–872.

[21] M. Hegde, P. Kumar, K. Vasudev, N. N. Sowmya, S. Anand, A. Kumar, and J. Kuri, "Experiences with a centralized scheduling approach for performance management of IEEE 802.11 wireless LANs," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 648–662, Apr. 2013.

[22] O. Bhardwaj, G. Sharma, M. Panda, and A. Kumar, "Modeling finite buffer effects on TCP traffic over an IEEE 802.11 infrastructure WLAN," *Comput. Netw.*, vol. 53, no. 16, pp. 2855–2869, 2009.

[23] D. J. Leith, P. Clifford, D. Malone, and A. Ng, "TCP fairness in 802.11e WLANs," *IEEE Commun. Lett.*, vol. 9, no. 11, pp. 964–966, Nov. 2005.

[24] Y. Wu, Z. Niu, and J. Zheng, "Study of the TCP upstream/downstream unfairness issue with per-flow queuing over infrastructure-mode WLANs," *Wireless Commun. Mobile Comput.*, vol. 5, no. 4, pp. 459–471, 2005.

[25] A. Gupta, J. Min, and I. Rhee, "WiFox: Scaling WiFi performance for large audience environments," in *Proc. 8th Int. Conf. Emerging Netw. Experiments Technol.*, 2012, pp. 217–228.

[26] S. Krishnasamy and A. Kumar, "Modeling the effect of transmission errors on TCP controlled transfers over infrastructure 802.11 wireless LANs," in *Proc. 14th ACM Int. Conf. Model., Anal. Simul. Wireless Mobile Syst.*, 2011, pp. 275–284.

[27] H. Vu and S. Hanly, "A study of TCP performance and buffer occupancy over a fading wireless link," in *Proc. IEEE Global Telecommun. Conf.*, 2001, vol. 6, pp. 3478–3482.

[28] P. Lassila, H. van den Berg, M. Mandjes, and R. Kooij, "An integrated packet/flow model for TCP performance analysis," *Teletraffic Sci. Eng.*, vol. 5, pp. 651–660, 2003.

[29] N. Blefari-Melazzi, A. Detti, I. Habib, A. Ordine, and S. Salsano, "TCP fairness issues in IEEE 802.11 networks: Problem analysis and solutions based on rate control," *IEEE Trans. Wireless Commun.*, vol. 6, no. 4, pp. 1346–1355, Apr. 2007.

[30] D. Pong and T. Moors, "Fairness and capacity trade-off in IEEE 802.11 WLANs," in *Proc. 29th Annu. IEEE Int. Conf. Local Comput. Netw.*, 2004, pp. 310–317.

[31] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[32] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, "New insights from a fixed point analysis of single cell IEEE 802.11 WLANs," in *Proc. 24th Annu. Joint Conf. IEEE Comput. Commun. Soc.*, 2005, vol. 3, pp. 1550–1561.

[33] S. Datta and S. Das, "Analyzing the effect of client queue size on VoIP and TCP traffic over an IEEE 802.11e WLAN," in *Proc. 16th ACM Int. Conf. Model., Anal. Simul. Wireless Mobile Syst.*, 2013, pp. 373–376.

[34] 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. Netw.*, vol. 8, no. 2, pp. 133–145, Apr. 2000.

[35] N. Parvez, A. Mahanti, and C. Williamson, "An analytic throughput model for TCP NewReno," *IEEE/ACM Trans. Netw.*, vol. 18, no. 2, pp. 448–461, Apr. 2010.

[36] R. Dunaytsev, D. Moltchanov, Y. Koucheryavy, and J. Harju, "Modeling TCP SACK performance over wireless channels with completely reliable ARQ/FEC," *Int. J. Commun. Syst.*, vol. 24, no. 12, pp. 1533–1564, 2011.

[37] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 3, pp. 67–82, 1997.

[38] R. Morris, "Scalable TCP congestion control," in *Proc. 19th Annu. Joint Conf. IEEE Comput. Commun. Soc.*, 2000, vol. 3, pp. 1176–1183.

Shiva Raj Pokhrel received the MEng (CE) degree from NCIT Pokhara University. He is currently working toward the PhD dgeree at Swinburne University of Technology, Australia. He has been a telecom engineer at Nepal Telecom since 2007. His areas of research are performance modeling and optimization of data communication networks, Transport protocols, WiFi, and 3G/4G.

Manoj Panda received the MTech (EE) degree from IIT Kanpur and the PhD (ECE) degree from IISc Bangalore. He is a research fellow at Swinburne University of Technology, Australia. Previously, he was a postdoctoral researcher at INRIA Sophia Antipolis (MAESTRO Group) and Telecom SudParis. His areas of research are performance modeling and optimization of computer communication networks, vehicular networks, and transportation networks.

Hai L. Vu received the BSc/MSc, and PhD degrees in electrical engineering from the Technical University of Budapest, Hungary, in 1994 and 1999, respectively. From 1994 to 2000, he was a research engineer with Siemens AG, Hungary. Between 2000 and 2005, he was with the Department of Electrical and Electronic Engineering, University of Melbourne, Australia. In 2005, he joined the Swinburne University of Technology, where he currently is the head in the Intelligent Transport Systems Research Lab. Since 2012, he has been a professor in the Faculty of Science, Engineering & Technology, Swinburne, and is an Australian Research Council (ARC) Future fellow. He has authored or coauthored more than 150 scientific journals and conference papers. His research interests include performance analysis and design of wireless data networks, and stochastic optimization with applications to intelligent transport systems. He has been a senior member of the IEEE since 2006.

Michel Mandjes received the MSc (in both mathematics and econometrics) and PhD degrees from the Vrije Universiteit (VU), Amsterdam, the Netherlands. After having worked as a member of technical staff at KPN Research, Leidschendam, the Netherlands, and Bell Laboratories/Lucent Technologies, Murray Hill, NJ, as a part-time full professor at the University of Twente, and as the department head at CWI, Amsterdam, he currently holds a full professorship (chair of Applied Probability) at the University of Amsterdam, the Netherlands. He is also affiliated (as an advisor) with EURANDOM, Eindhoven, the Netherlands. In 2008, he spent a sabbatical at Stanford. His research interests include performance analysis of communication networks, queueing theory, advanced simulation methods, Gaussian traffic models, traffic management and control, and pricing in multiservice networks. He is the author of *Large Deviations for Gaussian Queues*, Wiley, 2007.