Consiglio Nazionale delle Ricerche

Stochastic Models of CP Flows over 802.11 WLANs

R. Bruno, M. Conti, E. Gregori

IIT TR-11/2005

Technical report

Giugno 2005

Istituto di Informatica e Telematica

Stochastic Models of TCP Flows over 802.11 WLANs

Raffaele Bruno, Marco Conti, Enrico Gregori
Institute for Informatics and Telematics
Italian National Research Council (CNR)
Via G. Moruzzi, 1 – 56100 Pisa, Italy
email:{r.bruno,m.conti,e.gregori}@iit.cnr.it

Abstract

This technical report develops an analytical framework to model the interaction between TCP and 802.11 MAC protocol over a WLAN, when concurrent TCP downlink and uplink connections are active. Assuming a TCP advertised window equal to one, we formulate a Markov model to characterize the dynamic network contention level, defined as the expected number of wireless stations having at least a frame to transmit. Exploiting the stochastic characterization of the dynamic contention level induced by the TCP flow control, we develop an accurate model of the MAC protocol behavior to evaluate the TCP throughput performance. Comparison with simulation results validates the model, which provides the analytical basis for the optimization of the system performance. In particular, we prove that using a TCP advertised window equal to one ensures a fair access to the TCP flows of the channel bandwidth, irrespective of the number of TCP downlink or uplink connections. Moreover, we show that the aggregate TCP throughput is almost independent of the number of wireless stations in the network.

Index Terms

WLAN, 802.11 MAC Protocol, TCP, Throughout Analysis, Markov Model, Performance Evaluation.

I. Introduction

The last few years have seen an exceptional growth of the Wireless Local Area Network (WLAN) industry, with the substantial increase in the number of wireless users and applications. This growth was due, in large part, to the availability of inexpensive and highly interoperable network solutions based on the IEEE 802.11b standard [1], and to the growing trend of providing built-in wireless network cards into mobile computing platforms. Today, the attention of the industry and service providers is turning to deploying WLANs also over public places, or *hot spots*, such as cafes, retail shops, convention centers, airports and so on, from which people can benefit from a seamless public access to the Internet. Market analysts forecast that the hot spot locations will rise from the current $28,000$ to more than $160,000$ sites worldwide by 2007 [2]. Nevertheless, there are several business and technological challenges that have to be addressed to make public WLANs a global network infrastructure [3]. In particular, as the number of users increases and new emerging applications appear, the capability of ensuring minimum level of quality of service (QoS) to the users, in terms of guaranteed throughput, end-to-end delay bounds, etc., is one of

This work was carried out under the financial support of the Italian Ministry for Education and Scientific Research (MIUR) in the framework of the Projects: FIRB-PERF and FIRB-VICOM

the most crucial concerns for the service providers and the hot spot operators. However, to design adaptive resource management and bandwidth provisioning schemes it is important to have a clear understanding of which are the limitations of such networks in terms of scalability and efficiency, and how the resource availability depends on the traffic characteristics and users' behaviors. The goal of this technical report is to present a comprehensive and accurate mathematical analysis to investigate the hot spot throughput performance in the presence of multiple TCP flows, which will be useful to guide the network design and dimensioning process. We considered the TCP because it is the core of today's Internet transport layer, and more than 95% of the Internet traffic is carried over TCP [4].

The hot spot networks rely on a shared wireless channel, therefore the behavior of the medium access control (MAC) protocol, and the overheads introduced by the MAC protocol to perform its coordination tasks, are fundamental factors in determining the efficiency of the channel access. For this reason, our analysis will specifically focus on modeling the 802.11 MAC protocol operations. The 802.11 MAC protocol provides as basic access method the Distributed Coordination Function (DCF), which is based on a CSMA/CA scheme using a slotted binary exponential backoff (BEB)[1]. In the literature, it is widely recognized that the backoff protocol plays a crucial role in achieving a high aggregate throughput and a fair allocation of channel bandwidth to stations [6]. As a consequence, a considerable attention has been given within the research community to the modeling of the 802.11 backoff mechanism to derive the maximum channel utilization achievable by the 802.11 MAC protocol in different network conditions and configurations, e.g., [7]–[14] and references herein. These analytical studies have highlighted that the standard backoff algorithm significantly degrades the channel utilization in conditions of high contention, i.e., with a large number of stations, because this policy has to pay the cost of a collision to increase the backoff time when the network is congested. However, these previous results cannot be directly applied to hot spot networks because they where derived assuming *inelastic* traffic, which doesn't adapt its sending rate to the available channel bandwidth. Specifically, almost all the above mentioned papers assumed that the stations operate in asymptotic conditions, i.e., as sources with an unlimited amount of data to send and having always a packet ready for transmission[2]. Moreover, it is usually assumed a uniform traffic distribution, i.e., each frame transmitted by a node is destined to another randomly selected node.

To precisely model the hot spot network performance in presence of TCP flows is necessary to relax, or modify, several assumptions usually adopted to derive the saturation throughput of 802.11-based wireless networks. First of all, from a network perspective, a hot spot is an infrastructure-based network where the mobile users (hereafter indicated also as STAs) access the network services through a base station or access point (AP). As a consequence, the traffic is not uniformly distributed over the stations in the network, but all the frames are sent to/delivered from the AP, which behaves as traffic aggregator. Hence, the AP is the real bottleneck of the network throughput, affecting and limiting the performance of the wireless stations. Moreover, the TCP traffic is *elastic* because the TCP sending rate is regulated through a window-based flow control protocol that avoids continuous packet transmissions. In particular, two mechanisms – the flow control and the congestion control – are employed by the TCP to limit the transmission

[1]We assume that the reader is familiar with the IEEE 802.11 MAC protocol (see [1] for the standard specification and [5] for a survey on the basic mechanisms used in the wireless MAC protocols).

[2]A station with a non empty transmission buffers is indicated as *backlogged* station.

rate at the source [15]. Flow control determines the rate at which data is transmitted between senders and receivers, such as to avoid that the sender transmits more packets than the receiver can process. Congestion control defines the methods for implicitly interpreting signals from the network (e.g., received TCP acknowledgment packets or timer expirations) in order to adjust the sender's transmission rate. According to these mechanisms, wireless stations that are the receivers of TCP sessions will have feedback traffic (i.e., TCP acknowledgment packets) to transmit to the AP depending on the number of TCP data packets they receive from the AP. Similarly, wireless stations that are the senders of TCP sessions will have TCP data packets to transmit depending on the pace of TCP acknowledgment packets received from the AP. As a consequence, the complex interaction between the contention avoidance part of the 802.11 MAC protocol, and the closed-loop nature of TCP, induce a bursty TCP behavior with the TCP senders (or the TCP receivers) that could be frequently in idle state, without data (or acknowledgment) packets to transmit. Thus, the correct characterization of the wireless stations' behavior requires an accurate modeling of the *distribution* of the number n_b of backlogged stations in the network.

Keeping in mind the above-discussed peculiarities, in this technical report we will develop a *generalized* analytical framework capable of modeling the interactions between TCP and 802.11 MAC protocol over a WLAN, when concurrent TCP downlink and uplink sessions are active. Our model will be exploited to obtain closed-form expressions of the throughput of persistent TCP connections having an infinite amount of data to send (e.g., infinite file transfers). There is a vast literature on modeling of TCP sessions, e.g., [16]–[21] and references herein; and several mathematical techniques have been used to develop these analytical studies, from the network queuing theory, to stochastic models using discrete Markovian chains or fluid models. However, these models mainly focus on characterizing the TCP evolution, either at packet-based level or macroscopic level, and on investigating the throughput performance of TCP connections in the presence of congestion and loss events caused by bottlenecked networks and noisy channels. These models can help to understand the impact of network and TCP parameters on the throughput of the TCP connections, but cannot describe the impact of the TCP on the underlying network, and the MAC protocol in particular. Indeed, the interaction between TCP and 802.11 MAC protocol is a fundamental point to identify and explain eventual performance issues. To the best of the authors' knowledge a few pioneering attempts have been recently made to model this interaction in hot spot networks [22], [23]. However, these works lack completeness in one way or the other. In [23], approximate expressions are derived for the TCP throughput in the case of multiple competing connections, identifying lower and upper bounds for the TCP throughput. While, the authors in [22] developed an analytical model to express the throughout of TCP connections, but that study was limited to TCP downlink flows.

In this technical report we step forward with respect to the above-mentioned papers because we develop a precise model of the dynamic *network contention* level induced by the competition of concurrent TCP downlink and uplink connections. Specifically, assuming a TCP advertised window equal to one, we formulate a discrete Markov model to characterize the steady-state distribution of the number of backlogged wireless stations in the network. Then, we derive an accurate stochastic model of the MAC protocol operations, embedding in the protocol model the distribution of the network backlog. The characterization of the steady-state MAC protocol behavior is exploited to obtain closed-form expressions for the throughput of persistent TCP flows, which are the analytical basis for the optimization of the network performance. Comparison with simulation results validates our analysis: the observed behavior shows a very good match with the model predictions in all the considered network configurations. Exploiting the proposed analysis, we have proved two counter-intuitive results, differing from previous findings on throughput performance of saturated wireless networks [10], [18]: $i)$ the aggregate throughput is almost independent of the number of wireless stations in the network because the average network backlog is negligibly affected by the network size in the case of TCP-like elastic traffic; $ii)$ using a TCP advertised window equal to one ensures a fair access of the channel bandwidth to the TCP flows, irrespective of the number of TCP downlink or uplink connections.

The technical report is organized as follows. The next Section II describes the network configuration considered in this work, and it introduces the assumptions adopted to make the problem analytically tractable. In Section III we develop the analytical model for the throughput of concurrent persistent TCP downlink and uplink connections. The comparison between analytical and simulations results, validating the model is reported in Section IV. Finally, Section V discusses on future research directions.

II. System Model

We consider a hot spot network formed by n mobile users connected to an AP through an ideal wireless channel, i.e., not affected by channel errors. Let us assume that n_D wireless stations are the receivers of TCP downlink connections opened between the AP and the mobile users (hereafter indicated also as S_i^D, with $i = 1,\ldots,n_D$), while the remaining $n_U = n - n_D$ wireless stations are the senders of TCP uplink connections opened between the mobile users (hereafter indicated as S_i^U, with $i = n - n_D,\ldots,n$) and the AP. Each wireless station is involved in a single *long-lived* TCP session, i.e., a persistent TCP connection with an unlimited amount of data to send. In order to make analytically tractable the problem of modeling the interaction between the TCP window-based flow control mechanisms and the DCF access scheme, we have adopted some simplifying assumptions, which will be introduced and motivated in the remaining of this section.

Our model of the MAC protocol behavior is based on the assumption that the devices access the wireless slotted channel adopting a p-persistent IEEE $802.11b$ protocol [10]. This implies that a device uses a backoff interval sampled from a geometric distribution with parameter p. In [10], it was proved that by setting $p = 1/(E[B]+1)$ (where $E[B]$ is the average backoff time of the standard protocol[3]), the p-persistent IEEE 802.11 model provides an accurate approximation (at least from a capacity analysis standpoint) of the IEEE 802.11 protocol behavior. The off-line computation of the $E[CW]$ used by the standard $802.11b$ MAC protocol is carried out executing an iterative algorithm defined in Appendix II, which follows the approach firstly defined in [10].

As far as the TCP evolution is concerned, we assume that: (i) the transmission buffers at the AP and wireless stations are large enough to ensure that no packets are lost due to buffer overflows; (ii) the retransmission timers at the TCP senders are long enough to avoid timeout expirations and useless retransmissions of TCP data packets; (iii) the receiver advertised window W^* is equal to one for all the TCP connections; and (iv) each TCP data packet

[3]Note that $E[B]$ is related to the average contention window size, $E[CW]$, through the relationship $E[B] = (E[CW]-1)/2$ (see Lemma 2 in [10]).

is acknowledged separately, that is the delayed-ACK mechanism [15] is disabled. Assumption (i) implies that the system is well dimensioned to avoid packet losses due to congestion on the wireless channel. This is reasonable considering that in hot spot networks the shared radio channel and the high protocol overheads introduced by the DCF access scheme are the main bottlenecks for the TCP performance. Furthermore, it is quite common to consider short Round Trip Times (RTTs) in this kind of high speed networks [16], such that no retransmission timeouts occur as assumed in point (ii) above. It is worth noting that the same assumptions are adopted both in [22] and [23]. Since no TCP data-packet losses and timeout expirations occur, the congestion window increases up to reach, after a transient phase, the advertised window W^* value, which is the flow control imposed by the TCP receivers. Under the assumptions (i), (ii) and (iv) above, for the i^{th} TCP connection (with $i = 1, \ldots, n$) the sum of the number $X_{tcp}(i)$ of TCP data packets queued in the transmission buffer of the sender (i.e., the AP for downlink connections or the S_i^U for the uplink connections) and the number $X_{ack}(i)$ of TCP acknowledgment packets in the transmission buffer of the receiver ((i.e., the S_i^D for downlink connections or the AP for the uplink connections) is strictly equal to W^* at any time instant. For this reason, assumption (iii) above implies that $X_{tcp}(i) + X_{ack}(i) = 1$. Under this assumption we can precisely characterize the steady-state distribution of the number n_b of backlogged wireless stations, as it will be explained in Section III-A. Regarding assumption (iii), it is worth remarking that in [23] it was analytically studied the impact of the W^* value on the throughput performance for a single TCP connection. The authors in [23] proved that the maximum congestion window size has a negligible impact on the connection throughout. Furthermore, the throughout improvement for long-lived TCP connections due to $W^* > 1$ reduces as the number of TCP connections increases. For this reason, we claim that under the assumption $W^* = 1$ the proposed model provides a useful insight of the real system behavior. Finally, it is worth pointing out that we have assumed that the delayed-ACK mechanism is disabled for the sake of simplicity, however, our model can be easily extended to take into account the delayed-ACK interval.

III. Throughput Analysis

Let us consider the time interval between two consecutive AP's successful transmissions, hereafter denoted as *virtual transmission time* T_v. Let us indicate with $T_v(k)$ a virtual transmission time that begins with k active STAs (i.e., $n_b = k$). The number of active stations during the $T_v(k)$ is a random variable that changes according to the progress of STAs' transmissions occurring during the $T_v(k)$. Under the p-persistent assumption, and using classical renewal theoretical arguments [24], it is straightforward to note that the system behavior is regenerative with respect to the sequence of time instants corresponding to the beginning of $T_v(1)$[4]. Hence, it follows that the channel utilization ρ can be computed as

$$\rho = \frac{E[T_f]}{E[T_{renewal}]} \ , \tag{1}$$

[4]It is worth pointing out that after an AP's successful transmission there has to be at least a backlogged station, because the AP's transmission causes the generation of either a new TCP acknowledgment packet in one of the inactive S_i^D or a new TCP data packet in one of the inactive S_i^U.

where $E[T_{renewal}]$ is the average duration of the renewal period, i.e., the time interval between two consecutive $T_v(1)$, and $E[T_f]$ is the average time the channel is occupied by successful frame transmissions during a $T_{renewal}$. The following Lemma provides a closed-form expression for the ρ value.

Lemma 1: By denoting with $E[l_v]$ the average time the channel is occupied by successful frame transmissions during the average virtual time $E[T_v]$, and with Π_k the stationary probability of having $n_b = k$ active STAs at the beginning of a generic T_v, it holds that

$$\rho = \frac{E[l_v]}{\sum_{k=1}^{n} E[T_v(k)] \cdot \Pi_k} . \quad (2)$$

Proof: Let us denote with Ω the number of virtual transmission times forming a generic $T_{renewal}$, and with l_{v_j} and T_{v_j}, the j^{th} element in the sequence of l_v and T_v forming the considered $T_{renewal}$ ($j = 1, 2, \ldots, \Omega$). Hence, it follows that $E[T_{renewal}] = E[\sum_{j=1}^{\Omega} T_{v_j}]$ and $E[T_f] = E[\sum_{j=1}^{\Omega} l_{v_j}]$. Let us indicate with $\Omega(k)$, the number of $T_v(k)$ among the Ω virtual transmission times forming the $T_{renewal}$. From the p-persistent assumption, it follows that l_{v_j}, $T_{v_j}(k)$ and $\Omega(k)$ are i.i.d. random variables. Thus, the ρ expression can be rewritten as:

$$\rho = \frac{E[\sum_{j=1}^{\Omega} l_{v_j}]}{E[\sum_{k=1}^{n} \sum_{i=1}^{\Omega(k)} T_{v_i}(k)]} = \frac{\Omega \cdot E[l_v]}{\sum_{k=1}^{n} E[\Omega(k)] \cdot E[T_v(k)]} . \quad (3)$$

The ratio $E[\Omega(k)]/\Omega$ represents the stationary probability of having k active stations after an AP's successful transmission, that is the Π_k value by definition. Thus, formula (2) is straightforwardly obtained from formula (3), and this concludes the proof. ■

By inspecting formula (2), it is evident that the stochastic model of the system steady-state behavior is embedded at the AP's successful transmission instants. Moreover, we can also note that the analysis could be divided in two distinct phases. First, we derive the Π_k distribution, which corresponds to stochastically characterize the expected number n_b of backlogged stations in the network after an AP's successful transmission. Second, we will derive the $E[T_v(k)]$ and $E[l_v]$ expressions. Before describing our modeling solutions, let us introduce the notation that will be adopted in the following analysis:

- $E[X]_k$ (or $E[X(k)]$) is the expectation of the random variable X conditioned to having $n_b = k$[5];
- $Idle\_p$, $Coll$ and $Succ$ are, respectively, the duration of the idle period preceding a transmission attempt, the duration of a collision, and of a successful transmission;
- I_{AP} is a function that has value 1 when the AP performs a transmission attempt, 0 otherwise;
- Given the random variable X, we define we define $X^{AP} = \{X, I_{AP} = 1\}$ and $X^{STA} = \{X, I_{AP} = 0\}$;
- $SIFS$, $DIFS$, and $EIFS$ are the interframe spaces used in the $802.11b$ MAC protocol [1]; while t_B, t_H, and t_{ACK} are the time needed to transmit a byte, the MAC header and the MAC acknowledgment frame at the data rate r, respectively.

[5]For instance, $E[T_v(k)]$ and $E[T_v]_k$ will be used similarly throughout the document.

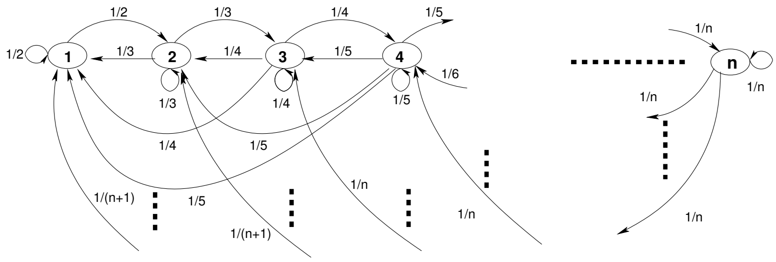


Fig. 1. Markov chain model for the number of backlogged stations n_b.

A. Derivation of the Π_k distribution

Let us consider the stochastic process $n_b(t)$, representing the number of backlogged STAs at the time instant t, corresponding to an AP's successful transmission (i.e., the beginning of a generic T_v), which is the embedding point of our model. A discrete and integer time scale is adopted, since t and $t+1$ correspond to the beginning of two consecutive T_v. From the p-persistent assumption, it follows that each transmission attempt is a successful transmission with constant and independent probability. As a consequence, the $n_b(t)$ process can be modeled as a discrete-time Markov chain. Thus, the probability Π_k (with $k=1,\ldots,n$) is the stationary state probability of the chain, that is $\Pi_k = \lim_{t\to\infty} Pr\{n_b(t) = k\}$. To derive the transition probabilities of this Markov chain, we will exploit the assumption that for the i^{th} TCP connection (with $i=1,\ldots,n$), it holds that $X_{tcp}(i)+X_{ack}(i)=1$, as explained in Section II. Specifically, for each TCP connection at most one TCP acknowledgment packet (for S_i^D stations) or TCP data packet (for S_i^U stations) is queued in the STA's transmission buffer. For this reason, after each STA's successful transmission the $n_b(t)$ value decreases by one, while after each AP's successful transmission the $n_b(t)$ value increases by one. The state transition diagram of the Markov chain modeling the $n_b(t)$ process is depicted in Figure 1. In this Markov chain the transition probabilities are provided by the following Lemma.

Lemma 2: Under the assumptions listed in Section II, the $P(i|j)$ value[6] can be computed as:

$$\begin{cases} P(i|j) = \frac{1}{j+1} & i\in\{1,\ldots,j+1\} \\ & j\in\{1,\ldots,n-1\} \\ P(i|n) = \frac{1}{n} & i\in\{1,\ldots,n\} \end{cases} \tag{4}$$

Proof: To prove formulas (4), we firstly derive the probability, indicated as $\Phi_k^{AP}\{l\}$, that the AP's successful transmission is the l^{th} successful transmission observed in the T_v, conditioned to having k active STAs at the beginning of the T_v. Since the AP's successful transmission is the l^{th} success, the previous $l-1$ successful transmissions involved only STAs. Hence, for $k<n$ it holds that

$$\Phi_k^{AP}\{l\} = P\{Succ^S|Succ\}_k \cdot P\{Succ^S|Succ\}_{k-1} \cdots P\{Succ^S|Succ\}_{k-l+1} \cdot P\{Succ^{AP}|Succ\}_{k-l} =$$

$$\left(1-\frac{1}{k+1}\right)\cdot\left(1-\frac{1}{k}\right)\cdots\left(1-\frac{1}{k-l+2}\right)\cdot\frac{1}{k-l+1} = \frac{1}{k+1}\,, \tag{5}$$

where $P\{S^{Succ}|Succ\}_k$ ($P\{Succ^{AP}|Succ\}_k$) is the probability that a successful transmission is a STA's (AP's) success, conditioned to having k active stations. In the case $k=n$, we have that the AP has an empty transmission

[6]We use the short notation $P(i|j) = P\{n_b(t+1) = i|n_b(t) = j\}$.

buffer, such that $P\{S^{Succ}|Succ\}_n = 1$. Consequently, it follows that

$$\Phi_n^{AP}\{l\} = P\{Succ^S|Succ\}_n \cdot P\{Succ^S|Succ\}_{n-1} \cdots P\{Succ^S|Succ\}_{n-l+1} \cdot P\{Succ^{AP}|Succ\}_{n-l} =$$

$$1 \cdot \left(1 - \frac{1}{n}\right) \cdots \left(1 - \frac{1}{n-l+2}\right) \cdot \frac{1}{n-l+1} = \frac{1}{n} \; . \qquad (6)$$

Inspecting formulas (5) and (6), we can note that the probability the AP's successful transmission is the l^{th} successful transmission observed in the T_v is independent of the l value. Now, we can easily compute $P(i|j)$, which denotes the probability that considering a generic $T_v(i)$, i.e., a virtual transmission time begins with $n_b(t) = i$, after the AP's successful transmission the new virtual transmission time is $T_v(j)$, i.e., it begins with $n_b(t+1) = j$. Specifically, if the first successful transmission observed in the T_v is an AP's success, a new STA is activated and $n_b(t+1) = j+1$. Hence, $P(j+1|j) = \Phi_j^{AP}\{1\}$. Generalizing, let us assume that the AP's successful transmission occurs after m STA's successful transmissions (with $m = 1, \ldots, j$). In this case, m STAs became inactive and the number of backlogged stations is $j - m$. Then, the AP's successful transmission activates a new STA and $n_b(t+1) = j - m + 1$. Hence, $P(j - m + 1|j) = \Phi_j^{AP}\{m+1\}$. Finally, the second expression in (4), takes into account that when all the n STAs are active, the AP has no packets to transmit and the first successful transmission observed in the T_v cannot be carried out by the AP. ■

Using the knowledge of the transition probabilities $P(i|j)$ given in (4), it is straightforward to find a closed-form formula for the Π_k. Specifically, by writing the equilibrium at each state of the chain, after some algebraic manipulations we obtain

$$\Pi_k = \frac{1}{(k-1)!} \cdot \Pi_1 \qquad k = 1, 2, \ldots, n \qquad (7)$$

Thus all the Π_k are expressed in terms of Π_1 and of the k parameter. Finally, by imposing the normalization condition on the steady state probabilities, Π_1 is determined as follows $\Pi_1 = 1 \cdot \left[\sum_{k=1}^{n} \frac{1}{(k-1)!}\right]^{-1}$.

The average number $E[n_b]$ of active stations, i.e., the average network backlog, can be computed as

$$E[n_b] = \sum_{k=1}^{n} k \cdot \Pi(k) \; . \qquad (8)$$

After standard algebraic manipulations of the $E[n_b]$ expression using formulas (7), it is straightforward to derive the following proposition.

Proposition 1: It holds that $E[n_b] \leq 2$. The relationship $E[n_b] = 2$ is asymptotically valid, i.e., it holds when $n \to \infty$.

This result is counter-intuitive because it demonstrates that, even when there are a large number of TCP flows, the average network backlog is limited. On average, less two STAs contend with the AP for the channel access. Numerical results substantiating this proposition are shown in Section IV.

In [23], it was proposed an alternative methodology to derive the statistics of the number of backlogged stations in the hot spot network. Specifically, under the assumption of a fair channel access[7], in that paper it was heuristically

[7]The assumption of a fair channel access implicitly comes from the fact that in [23] only TCP downlink connections are considered.

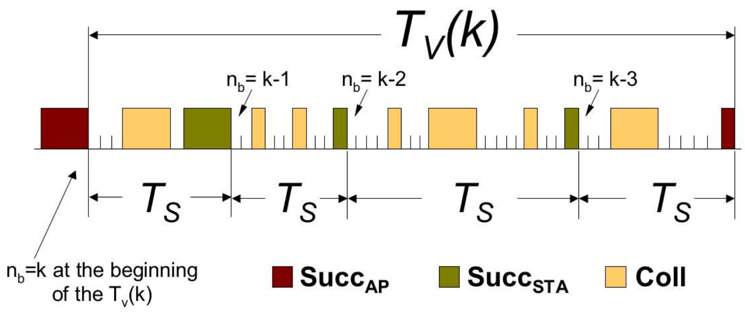


Fig. 2. Structure of channel events during a generic $E[T_v(k)]$.

derived that the probability a node is backlogged is simply $1/n$. Under the assumption of independence between the events that various stations have TCP acknowledgment packets to transmit, and denoting with π_k the fraction of time exactly k nodes are backlogged, it holds that

$$\pi_k = \binom{n}{k-1}\left(\frac{1}{n}\right)^{k-1}\left(\frac{n-1}{n}\right)^{n-k+1} \qquad k = 1, 2, \ldots, n+1 \ . \tag{9}$$

From formula (9), it is evident that the statistics of the number of backlogged stations has been derived without any specific embedding. This is due to the fact that in [23] the modeling of the underlying MAC protocol has been separated from the stochastic characterization of the network backlog.

B. Derivation of the $E[T_v(k)]$ expression

To derive the $E[T_v(k)]$ expression it is useful to consider the channel events occurring during a generic $T_v(k)$, as shown in Figure 2. Specifically, before the successful AP's transmission that completes the virtual transmission time, a number n_s of STAs' successful transmissions occurs. Let $T_s(k)$ denote the time interval between two consecutive successful transmissions when in the network there are k backlogged STAs. From the p-persistent model, it follows that the $T_s(k)$ are i.i.d. random variables. We can also note that a generic $T_s(k)$ can end with either an AP's successful transmission (in this case we indicate the $T_s(k)$ as a $T_s^{AP}(k)$), or a STA's successful transmission (in this case we indicate the $T_s(k)$ as a $T_s^{S}(k)$). In the latter case, the number of backlogged stations reduces to $k-1$, because that station has delivered the only TCP acknowledgment packet (for S_i^D stations) or TCP data packet (for S_i^U stations) queued in its transmission buffer. The following Lemma defines a recursive algorithm to compute the $E[T_v(k)]$.

Lemma 3: Assuming a p-persistent 802.11 MAC protocol and $W^* = 1$ for the TCP connections, it holds that

$$\begin{cases} E[T_v(0)] &= E[T_s^{AP}(0)] \\ E[T_v(k)] &= \Big\{E[T_s^{S}(k)] + E[T_v(k-1)]\Big\} \cdot Pr\{Succ^{S}|Succ\}_k + \\ & \quad E[T_s^{AP}(k)] \cdot Pr\{Succ^{AP}|Succ\}_k \qquad\qquad k < n \\ E[T_v(n)] &= E[T_s^{S}(n)] + E[T_v(n-1)] \end{cases} \tag{10}$$

Proof: The proof immediately follows by considering the structure of channel events depicted in Figure 2, and considering the stochastic independence of successful transmissions due to the p-persistent assumption. Specifically, the $T_v(k)$ interval ends when there is an AP's successful transmission. Let us consider the first successful transmission during the $T_v(k)$. It could be an AP's success with probability $Pr\{Succ^{AP}|Succ\}_k$, and this complete the $T_v(k)$, which have lasted $E[T_s^{AP}(k)]$. Otherwise, with probability $Pr\{Succ^{S}|Succ\}_k$, the first successful transmission will be a STA's success. In this case, the number of active STAs reduces to $k{-}1$, and the time remaining to complete the $T_v(k)$, after the $E[T_s^S(k)]$ period needed to perform the STA's successful transmission, is equal to the $E[T_v(k{-}1)]$[8]. Special cases is the $E[T_v(0)]$, because collisions cannot occur when no STA is active. ■

To complete the derivation of the $E[T_v(k)]$ expression, we have to derive the $E[T_s^S(k)]$ and $E[T_s^{AP}(k)]$ formulas. As shown in [8] and [10], the $E[T_s^x]_k$ (with $x \in \{S, AP\}$) can be written as

$$E[T_s^x(k)] = E[Idle\_p]_k \cdot \{E[n_c]_k + 1\} + E[n_c]_k \cdot \{E[Coll|Coll]_k + \tau + EIFS\} + E[Succ^x]_k \ , \tag{11}$$

where $E[n_c]_k$ is the average number of collisions occurring during the $T_s^x(k)$, and τ is the maximum propagation delay on the channel. To make easier the computation of $E[Coll|Coll]_k$, it is useful to distinguish between the collisions that either involve or not involve the AP. Specifically, $E[Coll|Coll]_k$ can be written as

$$E[Coll|Coll]_k = E[Coll|Coll^{AP}]_k \cdot P\{Coll^{AP}|Coll\}_k + E[Coll|Coll^{S}]_k \cdot P\{Coll^{S}|Coll\}_k \ , \tag{12}$$

where $P\{Coll^{AP}|Coll\}_k$ ($P\{Coll^{S}|Coll\}_k$) is the probability that a collision involves (doesn't involve) the AP, give that k stations are active. The average duration of collisions and successful transmissions will depend on the distribution of the transmitted frame sizes. Let us assume that the TCP connections generate data packets of fixed size equal to l_T bytes, while the TCP acknowledgment packets have size equal to l_A bytes. As a consequence, the S_i^D stations (i.e., the TCP receivers) will transmit MAC data frames with l_A byte-long payloads, and the S^U stations (i.e., the TCP senders) will transmit MAC data frames with l_T byte-long payloads . Let us define with $F_k^S(l)$ the probability that the length of a generic STA's frame transmission is lower than or equal to l bytes when $n_b = k$. Considering that the probability of having an active S^D station is independent of k, and it is equal to n_D/n, it follows that

$$F_k^S(l) = \begin{cases} 0 & l < l_A \\ n_D/n & l_A \le l < l_T \\ 1 & l \ge l_T \end{cases} \ . \tag{13}$$

In a similar way, we derive $F_k^{AP}(l)$, i.e., the probability that the length of a generic AP's frame transmission is lower than or equal to l bytes. In particular, it holds that

$$F_k^{AP}(l) = \begin{cases} 0 & l < l_A \\ n_U/n & l_A \le l < l_T \\ 1 & l \ge l_T \end{cases} \ . \tag{14}$$

From formulas (13 and (14) is evident that the distribution of MAC frame size is independent of the network backlog.

[8]It is worth remarking that $T_v(k-1)$ begins after an AP's successful transmission and not a STA's successful transmission. However, in our analysis we have approximated the average time needed to complete a virtual transmission time after a STA's successful transmission, which has reduced the network backlog to $k-1$ stations, with $E[T_v(k-1)]$.

The following Lemma provides closed-form expressions for the quantities contained in (11) and (12).

Lemma 4: Assuming a p-persistent 802.11 MAC protocol and $W^*=1$ for the TCP connections, and indicating with l_{Max} the maximum frame size supported by the MAC protocol, it holds that

$$E[Idle\_p]_k = \frac{(1-p)^{k+1}}{1-(1-p)^{k+1}} \cdot t_{SLOT} \ , \tag{15a}$$

$$E[n_c]_k = \frac{1-[(1-p)^{k+1}+(k+1)p(1-p)^k]}{(k+1)p(1-p)^k} \ , \tag{15b}$$

$$E[Coll^S|Coll]_k = t_H + \sum_{l=1}^{l_{Max}} \frac{t_B \cdot l}{1-[(1-p)^k+kp(1-p)^{k-1}]} \cdot \left\{[1-p(1-F_k^S(l))]^k - [1-p(1-F_k^S(l-1))]^k - [F_k^S(l)-F_k^S(l-1)]kp(1-p)^{k-1}\right\} \ , \tag{15c}$$

$$E[Coll^{AP}|Coll]_k = t_H + \sum_{l=1}^{l_{Max}} t_B \cdot l \cdot \sum_{i=2}^{k+1} \left\{[F_k^S(l)]^{i-1} \cdot F_k^{AP}(l) - [F_k^S(l-1)]^{i-1} F_k^{AP}(l-1)\right\} \cdot \frac{\binom{k+1}{i} p^i (1-p)^{k-i+1}}{1-[(1-p)^{k+1}+(k+1)p(1-p)^k]} \ , \tag{15d}$$

$$P\{Coll^{AP}|Coll\}_k = \sum_{i=2}^{k+1} \frac{i}{k+1} \cdot \frac{\binom{k+1}{i} p^i (1-p)^{k-i+1}}{1-[(1-p)^{k+1}+(k+1)p(1-p)^k]} \ , \tag{15e}$$

$$P\{Coll^{S}|Coll\}_k = \sum_{i=2}^{k+1} \frac{k-i+1}{k+1} \cdot \frac{\binom{k+1}{i} p^i (1-p)^{k-i+1}}{1-[(1-p)^{k+1}+(k+1)p(1-p)^k]} \ , \tag{15f}$$

$$E[Succ^{AP}]_k = 2\tau + SIFS + t_{ACK} + DIFS + t_H + t_B\left[\frac{n_D}{n} l_T + \frac{n_U}{n} l_A\right] \ , \tag{15g}$$

$$E[Succ^S]_k = E[Succ^{AP}]_k \frac{n-2n_D}{n} t_B (l_T - l_A) \ . \tag{15h}$$

Proof: See Appendix I. ■

Before concluding this section, it is worth pointing out that when $k=n$, the AP has an empty transmission buffer (as explained in section III-A). Thus, $P\{Coll^{AP}|Coll\}_n = 0$ and $P\{S^{AP}|S\}_n = 0$ and the formulas derived in Lemma 4 should be modified accordingly.

C. Derivation of the throughput performance for TCP downlink and uplink connections

Formula (2) provides the total channel utilization and, implicitly, the aggregate throughput. However, we need to distinguish between the throughout obtained by the TCP uplink and downlink connections. To this end, we have to compute the average number of successful transmissions carried out by S_i^U stations during the $E[T_v]$, say $E[n_s^U]$, and the average number of AP's successful transmissions destined to S_i^D stations during the $E[T_v]$, say $E[n_s^D]$. To derive the $E[n_s^U]$ value, we first compute the average number of STAs' successful transmissions during the $E[T_v]$, say $E[n_s]$. It holds that

$$E[n_s] = \sum_{k=1}^{n} E[n_s]_k \cdot \Pi_k \ . \tag{16}$$

The following Lemma defines a recursive algorithm to compute the $E[n_s]_k$ values.

Lemma 5: Assuming a p-persistent 802.11 MAC protocol and $W^* = 1$ for the TCP connections, it holds that

$$\begin{cases} E[n_s]_0 &= 0 \\ E[n_s]_k &= \{1 + E[n_s]_{k-1}\} \cdot Pr\{Succ^S|Succ\}_k \\ E[n_s]_n &= 1 + E[n_s]_{n-1} \end{cases}$$

Proof: The proof follows the same line of reasoning used in the proof of Lemma 3. Specifically, in this case only when there is a STA's successful transmission the n_s variable is incremented by one. ■

The probability that, given a STA's successful transmission, it has been carried out by a S_i^U station, is independent of the network backlog k, and it is equal to n_U/n. Thus, it follows that

$$E[n_s^U] = \frac{n_U}{n} \cdot E[n_s] \ . \tag{17}$$

Similarly, the probability that an AP's successful transmission is destined to an S_i^D station is independent of the k value, and it equal to n_D/n. Hence, $E[n_s^D]$ can be written as

$$E[n_s^D] = \sum_{k=1}^{n} \frac{n_D}{n} \cdot \Pi_k = \frac{n_D}{n} \ . \tag{18}$$

Finally, let us denote with ρ_D and with ρ_U the channel utilization of TCP downlink and uplink connections, respectively. It follows that

$$\rho_D = \frac{E[n_s^D] \cdot t_B \cdot (l_T - l_U)}{E[T_v]} = \frac{n^D \cdot t_B \cdot (l_T - l_U)}{E[T_v]} \ , \tag{19a}$$

$$\rho_U = \frac{E[n_s^U] \cdot t_B \cdot (l_T - l_U)}{E[T_v]} = \frac{E[n_s] \cdot n^U \cdot t_B \cdot (l_T - l_U)}{E[T_v]} \ , \tag{19b}$$

where l_U is the sum of the TCP header length and IP header length[9].

IV. Model Validation

In this section we validate the ability of our model to accurately capture the system behavior, in particular as far as the estimation of TCP throughput performance. To this end, in this section we compare the model predictions with numerical results obtained through realistic simulations of the system. The simulation environment we used is an extension of the one utilized in [12], with the implementation also of persistent TCP traffic flows. The TCP version we have considered is the TCP Reno [15]. If not otherwise specified, the TCP senders transmit TCP data

[9]Usually, $l_U = 40$ bytes.

TABLE I

IEEE 802.11B CONFIGURATION

t_{SLOT}	τ	$SIFS$	$DIFS$	$EIFS$
20 μsec	≤ 1 μsec	10 μsec	50 μsec	364 μsec
$PLCP_{hdr}$	MAC_{hdr}	CW_{min}	CW_{max}	Data Rate
192 bits	272 bits	32	1024	11 Mbps

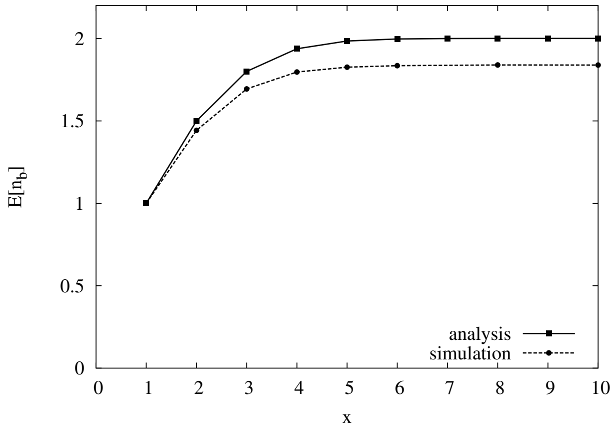


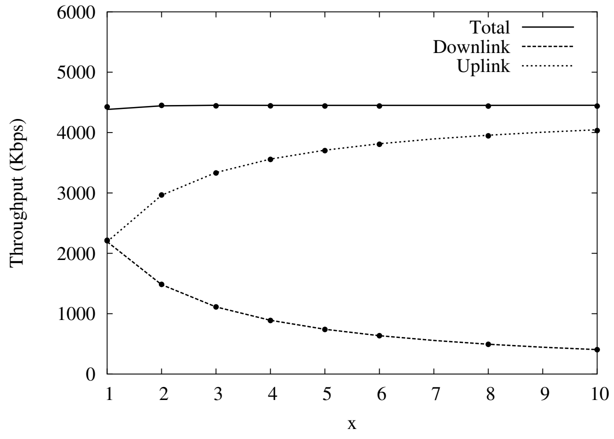
Fig. 3. Comparison of model predictions and observed behavior for the $E[n_b]$ value in the network scenario $DxU0$.

packets with a payload size equal to 1460 bytes. As a consequence, we used $l_T = 1500$ byets and $l_A = 40$ bytes[10] to compute the analytical results. As far as the p value, we used the value estimated through the iterative algorithm derived in Appendix II. Table I summarizes the values adopted for the physical and MAC protocol overheads during the simulations. This parameters' setting is fully compliant to the mandatory setting defined in the IEEE $802.11b$ standard for the 11 Mbps channel rate [1]. All the results presented henceforth are obtained by executing simulation runs long enough to guaranteed a 99% confidence level with a precision lower that 1%. For the sake of brevity, to indicate a network with $n_D = x$ and $n_U = y$, hereafter we will use the simplified notation $DxUy$.

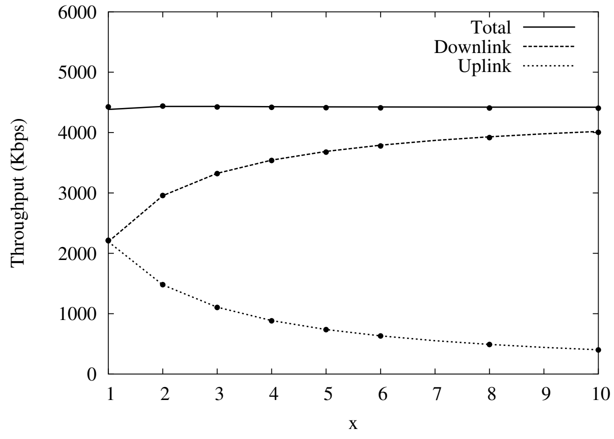
A first set of simulations has been carried out to validate the model of the network backlog distribution. This is a crucial point of our mathematical framework, because the network backlog distribution is embedded within the model of the MAC protocol operations. The graph in Figure 3 compares the average number of backlogged stations measured in the simulations with the predictions of our model, as given in formula (8). Since the distribution of the network backlog is independent of the n_D and n_U values, the figure shows results referring to the network scenario $DxU0$. From the results, we can note that the proposed model slightly overestimates the average network backlog. Moreover, the findings of Proposition 1 are confirmed because the average network backlog is lower than two STAs.

A second set of simulations has been conducted to validate the accuracy of formulas (19a) and (19b). In particular, Figures 4 compare the measured downlink, uplink and aggregate throughput achieved by the TCP connections for different network configurations. The lines in the graphs represent the model predictions, while the filled circles are the measured values observed during the simulations. From the shown graphs, we can note that our model

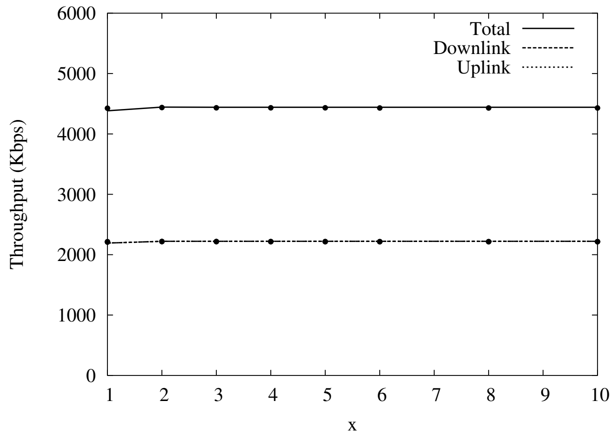
[10]The l_A and l_T values immediately follows by considering that the typical size of TCP and IP headers is 20 bytes each.



(a) Scenario $DxU1$



(b) Scenario $D1Ux$



(c) Scenario $DxUx$

Fig. 4. Comparison of model predictions and observed behavior for the throughput performance of TCP downlink and uplink connections.

provides a very good match to the observed behavior in all the considered traffic configurations. The results plotted in Figures 4 also indicate that the TCP flows fairly share the 802.11 channel bandwidth, irrespective of the number of wireless stations that are TCP receivers or senders. This result corresponds to what was also discovered in [18]. In particular, the authors in [18] proved that considerable TCP unfairness can occur in WLANs depending on the buffer availability B at the AP. In that paper, it was proposed to set the TCP advertised window equal to $\lfloor B/n \rfloor$ such as to avoid congestion events due to buffer overflows. In this technical report, we have analytically proved that setting the TCP advertised window equal to one is a sufficient condition to avoid TCP unfairness issues in hot spot networks.

V. Conclusions and Future Directions

This technical report presented an analytical framework to model the interaction between TCP and 802.11 MAC protocol in hot spot networks. Under the assumption of TCP advertised window equal to one, the proposed model accurately predict the throughput performance of concurrent long-lived TCP downlink and uplink connections. By exploiting the developed analysis, we have shown two counter-intuitive results: i) the channel utilization is almost independent of the number of TCP downlink or uplink connections; ii) the TCP connections fairly share the 802.11 channel bandwidth. However, several fundamental aspects of the performance of TCP sessions still require investigations, as

- TCP flows with different RTTs: In our analysis, as well as in the simulations, the TCP flows, either originate or terminate at the AP, such as to have the same RTT. However, in general the TCP flows originate and terminate in the wired part of the network, following different paths and experiencing different RTTs, which could significantly affect the TCP sending rate. Our model could be extended to take into account different RTTs.
- Competing UDP-like inelastic flows: Inelastic traffic is unresponsive traffic, which doesn't adapt its sending rate to the available link bandwidth. An example of this kind of traffic is given by the UDP traffic. TCP flows, which reduce their sending rate in response to congestion events, will be negatively affected by the presence of non-congestion-controlled traffic. Our model could be extended to take into account the competition of TCP flows with uncooperative UDP flows, such us to identify possible mechanisms to enforce fairness.
- Short-lived TCP connections: Today, the majority of TCP sessions in Internet deliver a quite limited amount of data (e.g., the web pages carried over HTTP traffic). For short-lived TCP flows a relevant performance index is not only the throughput achieved, but also the mean delay required to complete the session. Our model could be extended to take into account a realistic distribution of the TCP session durations, for instance following the modeling approach proposed in [25].

REFERENCES

[1] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification/Amendment 2: Higher-speed Physical Layer (PHY) in the* 2.4 *GHz band*, ANSI/IEEE Std. 802.11b, November 2001.

[2] "WiMAX and Broadband Wireless (Sub-11Ghz) Worldwide Market Analysis and Trends 2005 – 2010," Maravedis, Inc., March 2005.

[3] A. Balachandran, G. Voelker, and P. Bahl, "Wireless Hotspots: Current Challenges and Future Directions," in *Proceedings of 1st ACM Workshop on Wireless mobile applications and services on WLAN hotspots (WMASH'03)*, San Diego, CA, September 19 2003, pp. 1–9.

[4] K. Thompson, G. Miller, and R. Wilder, "Wide-area Internet traffic patterns and characteristics," *IEEE Network*, vol. 11, no. 6, pp. 10–23, Nov. 1997.

[5] A. Chandra, V. Gummalla, and J. Limb, "Wireless Medium Access Control Protocols," *IEEE Communications Surveys*, vol. 2, pp. 2–15, 2000. [Online]. Available: http://www.comsoc.org/pubs/surveys

[6] R. G. Gallager, "A Perspective on Multiaccess Channels," *IEEE Trans. Inform. Theory*, vol. 31, no. 2, pp. 124–142, 1985.

[7] L. Kleinrock and F. Tobagi, "Packet Switching in Radio Channels: Part I - Carrier Sense Multiple-Access Modes and their Throughput-Delay Characteristics," *IEEE Trans. Commun.*, vol. 23, pp. 1400–1416, 1975.

[8] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE J. Select. Areas Commun.*, vol. 18, no. 9, pp. 1787–1800, 2000.

[9] F. Calí, M. Conti, and E. Gregori, "Dynamic IEEE 802.11: design, modeling and performance evaluation," *IEEE J. Select. Areas Commun.*, vol. 18, no. 9, pp. 1774–1786, 2000.

[10] ——, “Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theorethical Throughput Limit,” *IEEE/ACM Trans. Networking*, vol. 8, no. 6, pp. 785–799, 2000.

[11] Y. Tay and K. Chua, “A Capacity Analysis for the IEEE 802.11 MAC Protocol,” *Wireless Networks*, vol. 7, pp. 159–171, 2001.

[12] R. Bruno, M. Conti, and E. Gregori, “Optimization of Efficiency and Energy Consumption in p-Persistent CSMA-Based Wireless LANs,” *IEEE Trans. Mob. Comp.*, vol. 1, no. 1, pp. 10–31, mar 2002.

[13] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, “Performance Anomaly of 802.11b,” in *Proceedings of IEEE Infocom 2003*, vol. 2, San Francisco, CA, March 30– April 3 2003, pp. 836–843.

[14] L. Bononi, M. Conti, and E. Gregori, “Runtime Optimization of IEEE 802.11 Wireless LAN Performance,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 15, no. 1, pp. 66–80, January 2004.

[15] W. Stevens, *TCP Illustrated, Volume 1: The Protocols*. New York, NY: Addison-Wesley, July 2001.

[16] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, “Modeling TCP Throughput: a Simple Model and its Empirical Validation,” *IEEE/ACM Trans. Networking*, vol. 8, no. 2, pp. 133–145, April 2000.

[17] A. Abouzeid, S. Roy, and M. Azizoglou, “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, March 2003.

[18] S. Pilosof, R. Ramjee, D. Raz, Y. Shavitt, and P. Sinha, “Understanding TCP fairness over Wireless LAN,” in *Proceedings of IEEE Infocom 2003*, vol. 2, San Francisco, CA, March 30– April 3 2003, pp. 863–872.

[19] M. Rossi, R. Vicenzi, and M. Zorzi, “Accurate Analysis of TCP on Channels with MEmory and Finite Round-Trip Delay,” *IEEE Trans. Wireless Commun.*, vol. 3, no. 2, pp. 627–640, March 2004.

[20] M. Garetto, R. Lo Cigno, M. Meo, and M. Marsan, “Closed Queueing Network Models of Interacting Long-Lived TCP Flows,” *IEEE/ACM Trans. Networking*, vol. 12, no. 2, pp. 300–311, April 2004.

[21] E. Altman, K. Avrachenkov, and C. Barakat, “A Stochastic Model of TCP/IP With Stationary Random Losses,” *IEEE/ACM Trans. Networking*, vol. 13, no. 2, pp. 356–369, April 2005.

[22] R. Bruno, M. Conti, and E. Gregori, “Analytical Modeling of TCP Clients in Wi-Fi Hot Spot Networks,” in *Proceedings of Networking 2004*, Athens, Greece, May 9–14 2004, pp. 626–637.

[23] D. Miorandi, A. Kherani, and E. Altman, “A Queueing Model for HTTP Traffic over IEEE 802.11 WLANs,” in *Proceedings of* $16th$ *ITC Specialist Seminar on Performance Evaluation of Mobile and Wireless Networks*, Antwerp, Belgium, September 1 2004.

[24] D. Heyman and M. Sobel, *Stochastic Modles in Operations Research*. New York, NY: McGraw-Hill, 2001, vol. 1.

[25] K. A.A. and K. Kumar, “On Processor Sharing as a Model for TCP Controlled HTTP-like Transfers,” in *Proceedings of ICC 2004*, Paris, FR, 2004.

APPENDIX I
PROOF OF LEMMA 4

- $E[Idle\_p]_k$ **computation**

The probability that no station decides to transmit at the beginning of an idle time slots when $n_b = k$ is $P\{n_{tr} = 0\}_k = (1-p)^{k+1}$. It is also straightforward to derive that $Pr\{n_{tr} \geq 1\}_k = 1 - P\{n_{tr} = 0\}_k$. The distribution of the number of consecutive idle slots that precede a transmission attempt is

$$Pr\{Idle\_p = i\}_k = \left[P\{n_{tr} = 0\}_k\right]^i P\{n_{tr} \geq 1\}_k \; i = 0, 1, \ldots \; .$$

Hence

$$E[Idle\_p]_k = t_{SLOT}[1-(1-p)^{k+1}] \sum_{i=1}^{\infty} i(1-p)^{i(k+1)} \; . \tag{I.1}$$

From (I.1), after standard algebraic manipulation, formula (15a) is obtained.

■

- $E[n_c]_k$ **computation**

Indicating with $P\{Coll|n_{tr}\geq 1\}_k$ the probability that a generic collision occurs conditioned to have at least one transmission in the slot and $n_b = k$; and with $P\{Succ|n_{tr} \geq 1\}_k$ the probability of a successful transmission conditioned to have at least one transmission in the slot and $n_b\!=\!k$, we have

$$P\{Coll|n_{tr}\geq 1\}_k = \frac{1-[(i-p)^{k+1}+(k+1)p(1-p)^k]}{1-(1-p)^{k+1}} \ , \tag{I.2}$$

and

$$P\{Succ|n_{tr}\geq 1\}_k = \frac{(k+1)p(1-p)^k]}{1-(1-p)^{k+1}} \ . \tag{I.3}$$

From (I.2) and (I.3) we derive the distribution of the number of collisions in a $T_s(k)$ as $P\{n_c = i|n_{tr} \geq 1\}_k = [P\{Coll|n_{tr}\geq 1\}_k]^i \cdot P\{Succ|n_{tr}\geq 1\}_k$, $i\!=\!0, 1, \ldots$. From this distribution, with standard algebraic manipulations, formula (15b) is obtained.

■

- $E[Coll|Coll^{AP}]_k$ **and** $E[Coll|Coll^{S}]_k$ **computation**

Since no collision detection is implemented in the IEEE 802.11b MAC protocol, the collision length $Coll$ depends on the duration of the colliding packets. To correctly compute $E[Coll|Coll^{AP}]_k$ and $E[Coll|Coll^{STA}]_k$, we have to distinguish between AP's and STAs' transmissions. First, we will derive the expression of $E[Coll|Coll^{AP}]_k$, that can be written as

$$E[Coll|Coll^{AP}]_k = t_H + \sum_{l=1}^{l_{Max}} t_B \cdot l \cdot \sum_{i=2}^{k+1} P\{Coll\!=\!l|I_{AP}\!=\!1, n_{tr}\!=\!i\}_k \cdot P\{n_{tr}\!=\!i|I^{AP}\!=\!1, n_{tr}\geq 2\}_k \ . \tag{I.4}$$

Assuming that the lengths of the colliding frames are i.i.d. random variables it follows that

$$P\{Coll\!=\!l|I_{AP}\!=\!1, n_{tr}\!=\!i\}_k =$$

$$Pr\{MAX\{l^{AP}, l^{S_1}, \ldots, l^{S_{i-1}}\} = l\}_k = F_k^{AP}(l)[F_k^S(l)]^{i-1} - F_k^{AP}(l-1)[F_k^S(l-1)]^{i-1} \ . \tag{I.5}$$

Finally, considering the p-persistent MAC protocol, it holds that

$$P\{n_{tr}\!=\!i|I^{AP}\!=\!1, n_{tr}\geq 2\}_k = \frac{\binom{k+1}{i} p^i (1-p)^{k-i+1}}{1-[(1-p)^{k+1}+(k+1)p(1-p)^k]} \ . \tag{I.6}$$

By substituting (I.5) and (I.6) in (I.4) we obtain (15c).

Similarly, to derive the expression of $E[Coll|Coll^{STA}]_k$ we start from

$$E[Coll|Coll^S]_k = t_H + \sum_{l=1}^{l_{Max}} t_B \cdot l \cdot \sum_{i=2}^{k+1} P\{Coll = l | I_{AP} = 0, n_{tr} = i\}_k \cdot P\{n_{tr} = i | I^{AP} = 0, n_{tr} \geq 2\}_k \; . \tag{I.7}$$

Under the same assumption used to derive (I.5), it follows that

$$P\{Coll = l | I_{AP} = 0, n_{tr} = i\}_k = Pr\{MAX\{l^{S_1}, l^{S_2}, \ldots, l^{S_i}\} = l\}_k = [F_k^S(l)]^i - [F_k^S(l-1)]^i \; . \tag{I.8}$$

Finally, considering the p-persistent MAC protocol, it holds that

$$P\{n_{tr} = i | I^{AP} = 0, n_{tr} \geq 2\}_k = \frac{\binom{k}{i} p^i (1-p)^{k-i}}{1 - [(1-p)^k + kp(1-p)^{k-1}]} \; . \tag{I.9}$$

By substituting (I.8) and (I.9) in (I.7), after some standard algebraic manipulations, we obtain (15d).

■

- **$P\{Coll^{AP}|Coll\}_k$ and $P\{Coll^S|Coll\}_k$ computation**

The probabilities that a collision either involves or doesn't involve the AP, can be computed as

$$P\{Coll^{AP}|Coll\}_k = \sum_{i=2}^{k+1} P\{I_{AP} = 1 | n_{tr} = i\}_k P\{n_{tr} = i | n_{tr} \geq 2\}_k \; , \tag{I.10}$$

and

$$P\{Coll^S|Coll\}_k = \sum_{i=2}^{k+1} P\{I_{AP} = 0 | n_{tr} = i\}_k P\{n_{tr} = i | n_{tr} \geq 2\}_k \; . \tag{I.11}$$

To derive the probability that the AP is transmitting, given that i stations are transmitting, we have to count how many ways exist to select the i transmitting nodes among $k+1$ nodes (the k STAs plus the AP). In particular it holds that

$$P\{I_{AP} = 1 | n_{tr} = i\}_k = \binom{k}{i-1} \Big/ \binom{k+1}{i} \; , \tag{I.12}$$

and

$$P\{I_{AP} = 0 | n_{tr} = i\}_k = \binom{k}{i} \Big/ \binom{k+1}{i} \; , \tag{I.13}$$

Finally, considering the p-persistent MAC protocol, it holds that

$$P\{n_{tr} = i | n_{tr} \geq 2\}_k = \frac{\binom{k+1}{i} p^i (1-p)^{k-i+1}}{1 - [(1-p)^{k+1} + (k+1)p(1-p)^k]} \; . \tag{I.14}$$

By substituting (I.12) and (I.14) in (I.10), and (I.13) and (I.14) in (I.11), we obtain formula (15e) and formula (15f), respectively.

■

- $E[Succ^{AP}]_k$ **and** $E[Succ^{S}]_k$ **computation**

To derive the $E[Succ^{AP}]_k$ and $E[Succ^{S}]_k$ expressions, we need to compute the average frame transmission time for the AP's and STAs' successful transmissions, say $E[l^{AP}]_k$ and $E[l^{S}]_k$, respectively. To this end, it is useful to compute the probability that an AP's successful transmission is destined to either a S^D or S^U wireless station. Considering that the AP transmits to a randomly selected STA, it follows that the probability that the AP's successful transmission is destined to a S^D station is n_D/n, and to a S_U station is n_U/n. Similarly, the probability that a STA's successful transmission is carried out by a S^D station is equal to n_D/n, and by a S_U station is n_U/n. Finally, considering that S^D stations receive TCP data packets and send TCP acknowledgment packets, while S^U stations receive TCP acknowledgment packets and send TCP data packets, taking into account the MAC protocol overheads, it holds that:

$$E[Succ^{AP}]_k = 2\tau + SIFS + t_{ACK} + DIFS + t_H + t_B\left[\frac{n_D}{n}l_T + \frac{n_U}{n}l_A\right] , \tag{I.15a}$$

$$E[Succ^{S}]_k = 2\tau + SIFS + t_{ACK} + DIFS + t_H + t_B\left[\frac{n_D}{n}l_A + \frac{n_U}{n}l_T\right] . \tag{I.15b}$$

■

Appendix II
Estimation of the p value

In [10], Calì *et al.* defined an iterative algorithm to calculate the average MAC contention window size of the standard protocol by focusing on a tagged station. In our study we leverage on this approach, by extending the algorithm proposed [10] to take into account the variable number of active STAs in the network. Specifically, we have developed an iterative algorithm that constructs the sequence $E[CW^{(j)}]$, with $j = 0, 1, 2, \ldots$, which is the average congestion window after j successful transmissions. The $E[CW]$ is the limiting value of this sequence, which is approximated by the value $E[CW^{(z)}]$, where z is the first index such that $E[CW^{(z)}] - E[CW^{(z-1)}] < \epsilon$. The starting value of the sought sequence is the minimal congestion value, say CW_{min}. It is useful to remind that the standard 802.11 MAC protocol adopts a binary exponential backoff. Specifically, a station that experiences i collisions, is in the i^{th} stage of the backoff process. At the i^{th} stage the congestion window is $CW_i = CW_{min}2^i$ if $i < m$, and $CW_i = CW_{min}2^m$ if $i \geq m$.

Our algorithm is based on the function $\Psi()$, such that $E[CW^{(j+1)}] = \Psi(E[CW^{(j)}])$. The $E[CW^{(j+1)}]$ is the average contention window computed by assuming that the tagged station is using as transmission probability $p^{(j)} = 2/(E[CW^{(j)}]+1)$. Consequently, $E[CW^{(j+1)}] = \Psi(p^{(j)})$. To derive the $\Psi()$ function we initially derive the $E[CW^{(j+1)}]_k$ value, that is the average contention conditioned to having $n_b = k$ active stations, by focusing on a tagged station. When the tagged station transmits, it collides if at least one other station tries to transmit as well. Thus, the probability of this collision at the $(j+1)^{th}$ iteration is

$$P^{coll}_{(j+1),k} = 1 - (1 - p^{(j)})^k . \tag{II.1}$$

Let us consider the process of transmitting a frame during the $(j+1)^{th}$ iteration. During this process, $h=0,1,2,\dots$ collisions may occur. In [10], it is has been proved that the probability of a station being at the stage h of the backoff process, can be computed as

$$P\{CW^{(j+1)}=CW_h\}_k=\begin{cases} P^{coll}_{(j+1),k}[1-P^{coll}_{(j+1),k}]^h & h<m \\ [P^{coll}_{(j+1),k}]^h & h=m \end{cases} \tag{II.2}$$

From (II.2), it immediately follows that

$$E[CW^{(j+1)}]_k=\sum_{h=0}^{m} CW_h\cdot P\{CW^{(j+1)}=CW_h\}_k \ . \tag{II.3}$$

Finally, we have that

$$E[CW^{(j+1)}]=\sum_{k=1}^{n} E[CW^{(j+1)}]_k\cdot \Pi_k \ . \tag{II.4}$$

Relationships (II.3) and (II.4) define the sought $\Psi()$ function.