ARTICLE IN PRESS

Computer Networks xxx (2010) xxx–xxx

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

On scaling the IEEE 802.11 to facilitate scalable wireless networks

Fragkiskos Papadopoulos *

Department of Electrical and Computer Engineering, University of Cyprus, Kallipoleos 75, Nicosia 1678, Cyprus

ARTICLE INFO

Article history:
Received 11 February 2009
Received in revised form 13 January 2010
Accepted 8 February 2010
Available online xxxx
Responsible Editor: A. Abouzeid

Keywords:
IEEE 802.11
Wireless local area networks (WLANs)
Scaling properties
Sustaining user performance

ABSTRACT

The IEEE 802.11 MAC protocol has gained widespread popularity and has been adopted as the de-facto layer 2 protocol for wireless local area networks (WLANs). However, it is well known that as the number of competing stations increases, the performance of the protocol degrades dramatically. Given the explosive growth in WLANs' usage, the question of how to sustain each user's perceived performance when a large number of competing stations are present, is an important and challenging open research problem.

Motivated by this, in this paper we analyze the behavior of 802.11-based WLANs as the number of competing stations increases, and attempt to provide concrete answers to the following fundamental questions: (i) is there a set of system and protocol parameters that we can scale in order to sustain each individual user's perceived performance, and (ii) what is the minimum scaling factor?

Using theoretical analysis coupled with extensive simulations we show that such a set of parameters exists, and that the minimum scaling factor is equal to the factor by which the number of users increases. Our results reveal several important scaling properties that exist in today's 802.11-based wireless networks, and set guidelines for designing future versions of such networks that can efficiently support a very large number of users.

1. Introduction

Due to its simple deployment and low cost the IEEE 802.11 Medium Access Control (MAC) protocol has been adopted as the standard layer 2 protocol for wireless local area networks (WLANs) [19]. In 2006 the number of worldwide IEEE 802.11 hotspots (public places where users can find wireless access to the Internet) has surpassed the 100,000 milestone, while the total number of hotspot users around the world is expected to reach 500 million by year 2009 [14].

Because of the popularity and usage of the 802.11 WLAN there has been a large body of work focusing on its analytical modeling, *e.g.* [6,39,40,11,7], simulation study, *e.g.* [18,29], and measurement-based performance evaluation, *e.g.* [4,3,27]. However, despite the large body of work on the system, there are still many problems related to it. Perhaps the most important one, from an end user's perspective, is that its perceived performance, in terms of packet delays and throughput, degrades *dramatically* with only a small increase in the total number of users/stations sharing the wireless channel, *e.g.* [6,39]. For the 802.11 WLAN to continue to thrive and evolve as a viable wireless access to the Internet, understanding how user performance depends on the number of competing stations, and how it can be sustained as this number increases, is an important and challenging open research problem.

Motivated by this, in this paper we attempt to answer the following fundamental questions: consider an 802.11 WLAN shared by αN users, where $\alpha \geqslant 1$ is a scaling factor. These users are randomly distributed around the base-station/access point of the WLAN and generate traffic destined to it according to some arbitrary arrival process. (i) Is there a set of system and protocol parameters that we can scale in order to sustain each individual user's perceived performance as the scaling factor α (and hence the

* Tel.: +357 22892245
E-mail address: fragkis@ucy.ac.cy

1389-1286/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2010.02.005

total number of users αN) increases? And, (ii) what is the minimum scaling factor?

Our theoretical analysis coupled with extensive ns-2 [22] simulations give concrete answers to both of the above questions. Interestingly enough, we find that such a set of parameters exists, and that the minimum factor by which one should scale these parameters is equal to the factor α by which the total number of users increases. In summary, the set of these parameters comprises the 802.11 MAC protocol timeouts, the transmission speeds of nodes, and two specific parameters of the 802.11 MAC protocol, the *minimum and maximum contention window size*, which regulate the transmission probability for each user. We show that if all these parameters are scaled by the factor α,[1] the perceived performance of each individual user remains virtually invariant as the total number of users (αN) increases, and quickly becomes independent of α.

The system scaling we study in this paper does not require any modification of the operations of the IEEE 802.11 MAC standard, which is desirable given its widespread adoption. Our results reveal several important scaling properties of today's 802.11-based WLANs, and set guidelines for designing future versions of such networks that can efficiently support a very large number of users.

The rest of the paper is organized as follows. In Section 2, we give a detailed description of the IEEE 802.11 MAC protocol, illustrate the main problem with it, and discuss related work. In Section 3, we analyze the behavior of 802.11-based WLANs under the proposed scaling. In Section 4, we verify our theoretical arguments using extensive ns-2 simulations. A discussion follows in Section 5, and we conclude in Section 6.

2. Preliminaries

In this section we first give a detailed description of the IEEE 802.11 MAC protocol. We then illustrate the main problem with it and explain how our approach could be used to solve it. Finally, we discuss related work.

2.1. Overview of the IEEE 802.11 MAC

The IEEE 802.11 MAC layer [19] is responsible for channel access and contains two methods, the distributed coordination function (DCF) and the point coordination function (PCF). In this paper we consider the DCF, which is specified as the fundamental access method and supported by all current wireless cards. Below we summarize its main functionality. For a more complete and detailed presentation the reader is referred to [19].

The DCF is based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol, which is designed to reduce the collisions due to multiple stations transmitting simultaneously on a shared channel. According to the DCF, time is slotted with the duration of each slot equal to a constant value, which we denote by σ. The value of σ is set equal to the time needed at any station to detect the transmission of a packet from any other station. A station with a packet to transmit shall ensure that the medium is idle before attempting to transmit. It performs a *backoff procedure*, with the *backoff timer* uniformly distributed over the interval $[0, CW)$, where CW is called *current contention window*. Initially $CW = CW_{min}$, where CW_{min} is called *minimum contention window*. The backoff timer is decremented by one at each time slot if the channel is sensed idle. If the channel is sensed busy (either by a successful transmission or collision among the other stations) the timer is stopped (i.e., freezes), and the decrementing process is restarted when the channel becomes idle again for an interval equal to the Distributed Inter-Frame Space (DIFS). When the backoff timer reaches zero and the channel is sensed to be idle for a DIFS time, the station transmits its data packet. Since the backoff interval is chosen randomly, the probability that two or more stations choose the same backoff slot to transmit is low, at least as long as the total number of competing stations is not large. If the receiver successfully receives the packet it waits for a brief period, called the Short Inter-Frame Space (SIFS), and acknowledges the packet by sending an acknowledgment (ACK). If no ACK is received within a specified period $ACK_{timeout}$, the packet is considered lost. A packet is lost either due to collisions at the receiver with transmissions from other stations, or when the transmission fails due to non-ideal channel conditions (*e.g.* fading). When a packet is lost, the transmitter will double the size of CW, choose a new backoff timer, and start the above process again. The value of CW can reach a maximum upper limit, called *maximum contention window* CW_{max}, where it remains there until it is reset. When the transmission of a packet fails for a maximum number of times k_{max} (whose typical values according to the standard are $k_{max} = 5, \ldots, 7$ [19]), the packet is dropped and CW is reset to CW_{min}.

To (try to) avoid collisions of long packets and the "hidden terminal" problem [16], the short request-to-send/clear-to-send (RTS/CTS) packets can be employed. A station that wishes to transmit a frame first sends an RTS packet to the destination in order to reserve the channel. The transmission of the actual data packet starts when the station receives a CTS packet from the destination. Notice that under this scheme, the collisions involve RTS packets and not actual data packets, since, it is the former that content for the channel. We refer to this method as the *RTS/CTS access method*, and to the method where RTS/CTS packets are not used as the *basic access method*.

Let L_{data} be the length of a data packet, L_{ack} be the length of an ACK packet, and C be the station transmission speed. Then, $T_{data} = \frac{L_{data}}{C}$ and $T_{ack} = \frac{L_{ack}}{C}$, are respectively the time needed to transmit a data packet and an ACK packet. Under the basic access method the total duration of a successful transmission T_{suc}, and of a collision T_{col}, are (*e.g.* [38,39]):

$$\begin{aligned} T_{suc} &= \mathrm{DIFS} + T_{data} + \mathrm{SIFS} + T_{ack}, \\ T_{col} &= \mathrm{DIFS} + T^{*}_{data} + ACK_{timeout}, \end{aligned} \tag{1}$$

where $ACK_{timeout} = \mathrm{SIFS} + T_{ack}$, and $T^{*}_{data} = \frac{L^{*}_{data}}{C}$, where L^{*}_{data} is the maximum length among collided packets. And, under the RTS/CTS access method these are (*e.g.* [38,39]):

[1] More precisely, the protocol timeouts divided by α, and the transmission speed of the stations and the minimum and maximum contention windows multiplied by α.

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

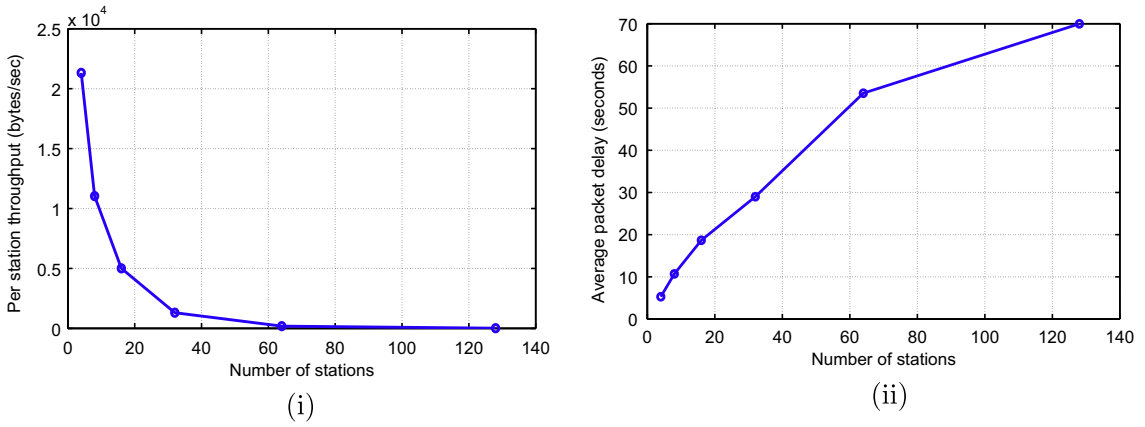

Fig. 1. (i) Per station throughput, and (ii) average packet delay, as a function of the number of competing stations.

$$T_{suc} = \text{DIFS} + T_{rts} + T_{cts} + T_{data} + T_{ack} + 3\text{SIFS},$$
$$T_{col} = \text{DIFS} + T_{rts} + ACK_{timeout}, \quad (2)$$

where $T_{rts} = \frac{L_{rts}}{C}$, $T_{cts} = \frac{L_{cts}}{C}$, are respectively the duration of an RTS and a CTS packet, with corresponding lengths L_{rts} and L_{cts}, and $ACK_{timeout} = \text{SIFS} + T_{cts}$.

In this paper we refer to the set of parameters $\{\sigma, \text{DIFS}, \text{SIFS}\}$ as the *protocol timeouts*. According to the standard: $\text{DIFS} = \text{SIFS} + 2\sigma$. Note that the values of these parameters, as well as of the station transmission speed C, depend on the physical layer specifications [19]. Different amendments of the original IEEE 802.11 standard use different values for these parameters, coupled with different radio signal transmission (*e.g.* modulation) techniques at the physical layer, in order to provide higher data rates. For example, $\sigma = 20$ μs, $\text{SIFS} = 10$ μs, and $C = 11$ Mbps in IEEE 802.11b [20], whereas $\sigma = 9$ μs, $\text{SIFS} = 5$ μs, and $C = 54$ Mbps in IEEE 802.11g [21]. Today, the single most modern 802.11 document available that contains all amendments is [2].

2.2. Problem and motivation

We now illustrate the main problem with the 802.11 MAC protocol via ns-2 [22] simulations. We consider stations uniformly distributed around a base-station/access-point that generate traffic destined to it. As we can see from Fig. 1(i), the throughput for each station decreases exponentially as the number of competing stations increases, while at the same time, the average packet delay increases significantly.[2] These general observations hold irrespectively of the exact access method used (basic or RTS/CTS access method), and of the exact traffic arrival process and load at the stations. They are expected since more competing stations result in a higher packet collision probability (and hence more packet drops), and more frequent backoffs. This problem becomes critical in wireless hotspots, which are characterized by a high concentration of users in small geographical areas, such as hotel lounges, airport lobbies, university campuses, conference areas, etc. In such cases the number of wireless stations associated with an access point usually exceeds 100 [5].

Motivated by this problem, in this paper we attempt to answer two important questions, which are formally stated as follows. Consider an IEEE 802.11 WLAN consisting of one access point and shared by αN users, where $\alpha \geqslant 1$ is a scaling factor: (i) is there a set of system and protocol parameters that we can scale in order to sustain each individual user's perceived performance, as the scaling factor α (and hence the total number of users αN) increases? And, (ii) what is the minimum (parameter) scaling factor?

Interestingly enough, we show that such a set of parameters exists and that the minimum parameter scaling factor equals the factor α by which the number of stations increases. In particular, we show that if the protocol timeouts $\{\sigma, \text{DIFS}, \text{SIFS}\}$ are divided by the factor α, and the transmission speed of nodes C and the minimum contention window CW_{min} and the maximum contention window CW_{max} are multiplied by the factor α, then the perceived performance of each user remains virtually invariant and quickly becomes independent of α. This holds under any conditions, *e.g.*, irrespectively of the network load, traffic arrival process, channel access method used, and so on. And, the factor α is the *minimum* scaling factor that guarantees this. For example, in Fig. 1, the perceived performance of a user when their total number is 128 will be approximately the same with the performance of a user when their total number is 4, as long as the aforementioned parameters are scaled by $\alpha = \frac{128}{4} = 32$.

2.3. Related work

The literature dealing with the IEEE 802.11 MAC protocol is abundant. There are primarily two main threads of prior research relevant to this paper: performance modeling/analysis and performance enhancement. Below, we briefly review some of the most representative results.

2.3.1. Performance analysis

The first thread has focused on deriving analytical models that characterize and predict the performance of the 802.11. The simple, but accurate, analytical model intro-

[2] By packet delay in this paper we refer to the time interval from the moment that a packet is generated at a station until it is successfully received by the access point.

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

duced by Bianchi [6], has become a common method for studying the throughput of the IEEE 802.11 protocol in saturated scenarios, where each station has always a packet available for transmission. The model was later refined to capture further details of the protocol's operations, such as the freezing of backoff counters, packet errors, impact of hidden terminals, and so on, *e.g.* see [39,40,11,7,30,35]. Further, under different approximations to ease analysis, several studies have also derived accurate throughput models for the 802.11 in non-saturated scenarios, as well as models for the channel access and packet queueing delay distributions, *e.g.* [39,32,34,13].

2.3.2. Performance enhancement

The second thread of research has focused on enhancing the 802.11 performance. In [9], Cali et al. analytically derived the average size of the contention window that maximizes throughput, and proposed in [8] the replacement of the exponential backoff mechanism with an adaptive one. Kim and Hou developed a model-based frame scheduling algorithm to improve the protocol capacity of 802.11 [15]. In [17] a fast collision resolution scheme was proposed, which dynamically adjusts the backoff timers and contention window sizes to avoid collisions. To provide service differentiation, Ada and Castelluccia [1] proposed to scale the contention window and use a different interframe spacing or maximum frame length for services of different priorities. In [33], the authors proposed an adaptive optimization algorithm that dynamically adjusts the 802.11 backoff parameters to maximize throughput based on estimations on the number of competing stations. Other interesting studies that propose adjustments to the backoff algorithm for enhancing 802.11 performance include [10,37]. Notice that the studies in this thread of research involve solutions that significantly modify the operations of the IEEE 802.11 standard.

Our work in this paper is significantly different from all prior studies of the IEEE 802.11 MAC protocol. First, our analysis studies how IEEE 802.11 WLANs behave under the scaling we described earlier and it is inspired by a set of earlier performance modeling studies. In particular, it is inspired by the studies in [39,40,32,34], which are of the most general ones. Second, we do not propose a new enhancement/modification to the operations of the existing IEEE 802.11 standard. Instead, we identify a set of (existing) system and protocol parameters that if appropriately scaled, leave the performance of each user virtually invariant as their total number increases.

The idea of scaling a network in a manner that performance is preserved has been extensively studied for the case of *wireline* networks that resemble the Internet, by Psounis and co-workers [23] and Papadopoulos et al. [26]. In the context of *wireless* networks, to our best knowledge, the only relevant to this work is the recent study by Papadopoulos and Psounis [24], where it has been shown that it is possible to predict the full behavior of an arbitrary mobile ad hoc network deployed in an outdoor environment at one spatial scale, by a suitably scaled replica consisting of the *same* number of nodes but deployed in an outdoor environment at another spatial scale. This was later experimentally verified in [31]. In this work, we investigate whether performance in IEEE 802.11 WLANs can be sustained while the number of nodes in the network *changes*.

A preliminary version of the material in this paper has appeared in [25], where we have primarily considered homogeneous stations in saturation, and briefly discussed the non-saturated case. Here, we extend and complement our work by rigorously considering the more realistic case of non-saturated stations, which can also be non-homogeneous.

3. Sustaining performance in IEEE 802.11 WLANs

In this section, we start by giving a detailed description of the network model we study and clearly stating our assumptions. Then, we derive a set of equations that characterize the network's performance and use them to theoretically support our arguments.

3.1. Network model and assumptions

Let αN denote the total number of competing stations in an IEEE 802.11 WLAN with a single access point, where $\alpha \geqslant 1$ is a scaling factor and N is some constant. The stations are randomly distributed around the access point and generate traffic destined to it according to some process. Further, let the protocol timeouts of this system be $\{\frac{\sigma}{\alpha}, \frac{\mathrm{DIFS}}{\alpha}, \frac{\mathrm{SIFS}}{\alpha}\}$, for some values of σ, DIFS, and SIFS, the transmission speed of each station be αC, for some value of C, and the minimum and maximum contention windows be respectively αCW_{min} and αCW_{max}, for some values of CW_{min} and CW_{max}. We call this system an *α-scaled system*.

We analyze the performance of α-scaled systems. Keeping the rest of the system parameters fixed, we show that as the factor α (and hence the total number of users αN) increases, the performance of each individual user never gets worse and becomes independent of α. For ease of exposition, and to stay focused on the impact of the 802.11 MAC protocol on performance, we make the following assumptions: (i) we assume ideal channel conditions with no capture, where packet loses at the physical layer are due to collisions only and not to any other factors, *e.g.* such as fading, and (ii) we ignore the hidden terminal problem [16], as in a typical WLAN environment every station can sense all the others stations' transmissions, although it may not be able to correctly receive packets from every one of them [38]. These assumptions have been also made in several other studies, including [6,39,40,11,7,30,38]. However, note that our findings in this paper hold even if these assumptions are not made as we explain in Section 5.

We model each competing station as a queueing system, which can be characterized by the packet arrival process and the service time distribution. The service time, is the MAC-layer service time, i.e., the time interval between the time instant a packet starts to contend for transmission and the time instant the packet is either acknowledged for correct reception by the access point or is dropped by the station. To incorporate user inhomogeneity we assume that there are $G \geqslant 1$ groups of users. Within each group $j \in \{1, \ldots, G\}$ the packet arrival process for each user is

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

the same and has an average rate denoted by λ_j packets/s. The packet arrival process and/or arrival rate can differ between users of different groups. For ease of exposition we assume a common packet size distribution for all users in the network. Our results hold even if this is not the case. In an α-scaled system the number of users belonging to group j is αn_j for some value of n_j, so that $\sum_{j=1}^{G} \alpha n_j = \alpha N$.[3] Next, we derive a system of equations that characterize the performance of α-scaled systems.

3.2. Performance analysis

Consider an α-scaled system and some station that belongs to some group of this system, say group j. We analyze the perceived performance of this station. In particular, we derive expressions for the following three interdependent quantities that govern the station's perceived performance: (i) the offered load at the station, (ii) the transmission and collision probabilities as seen by the station, and (iii) the packet service time distribution at the station.

3.2.1. Offered load

Let T_j be the random variable representing the packet service time at the station in terms of seconds/packet. The station's offered load is defined as $\lambda_j \overline{T}_j$.[4] In the analysis that follows we first assume that each station has an infinite buffer size, in which case the probability that the station has a packet contenting for transmission (i.e., that its queueing system is non-empty [36]) is:

$$\rho_j = \min\{1, \lambda_j \overline{T}_j\}. \tag{3}$$

3.2.2. Transmission and collision probabilities

Recall that in an α-scaled system the minimum contention window of a station is αCW_{min}, the maximum contention window is αCW_{max}, and let $m = \log_2\left(\frac{\alpha CW_{max}}{\alpha CW_{min}}\right) = \log_2\left(\frac{CW_{max}}{CW_{min}}\right)$. (Observe that m is independent of α.) Further, recall that the number of stations belonging to a group j is αn_j. We are interested in the following four probabilities: (i) τ_j, which is the conditional probability that the station transmits in a randomly chosen time slot given that it has a packet to transmit, (ii) p_j, which is the probability that there is a collision of the packet transmitted by the station, (iii) q_j, which is the conditional probability that there is one successful transmission among the other stations in a randomly chosen time slot given that the station under study does not transmit, and (iv) p_b, which is the probability that the channel is busy.

Given that a station has a packet to transmit, a relation for the station's transmission probability was first derived in [6] and later refined to capture further details of the 802.11 MAC operations. Among the refinements, a simple and general one is due to Ziouva and Antonakopoulos [40], which also captures the freezing of the backoff counters when the channel is sensed busy by the station. From Eq. (6) in [40]:

$$\tau_j \approx \frac{1}{\alpha CW_{min}} f(p_j, p_b, m), \tag{4}$$

where $f(p_j, p_b, m) = \frac{2(1-p_b)(1-2p_j)}{(p_b+p_j-p_jp_b)(1-p_j-p_j(2p_j)^m)}$. Clearly, the unconditional transmission probability, which is the probability that the station transmits in a randomly chosen time slot is $\rho_j\tau_j$.

The collision probability seen by the station when transmitting, is just the probability that there is at least one packet transmission in the medium from the other stations:

$$p_j = 1 - (1-\rho_j\tau_j)^{\alpha n_j - 1} \prod_{i=1, i\neq j}^{G} (1-\rho_i\tau_i)^{\alpha n_i}. \tag{5}$$

Further, the probability that one other station successfully transmits in some time slot, given that the station under study does not transmit, is:

$$\begin{aligned} q_j = {} & (\alpha n_j - 1)\rho_j\tau_j(1-\rho_j\tau_j)^{\alpha n_j - 2} \prod_{i=1, i\neq j}^{G} (1-\rho_i\tau_i)^{\alpha n_i} \\ & + \sum_{i=1, i\neq j}^{G} \alpha n_i \rho_i\tau_i (1-\rho_i\tau_i)^{\alpha n_i - 1} \\ & \times \prod_{k=1, k\neq i, k\neq j}^{G} (1-\rho_k\tau_k)^{\alpha n_k} (1-\rho_j\tau_j)^{\alpha n_j - 1}. \end{aligned} \tag{6}$$

The first term in Eq. (6) is the probability that the successful transmission is from a station that belongs to the same group as the station under study, and the second term is the corresponding probability for a station that belongs to some other group.

Notice that given p_j and q_j, one can also compute the probability that there is a collision in a time slot among the other stations, given that the station under study does not transmit. Denoting this probability by w_j, it is easy to see that $w_j = p_j - q_j$. As we will see shortly, all three probabilities p_j, q_j, w_j, are important for determining the packet service time seen by the station and the station's throughput.

Finally, the probability that the channel is busy at a time slot is just the probability that at least one station transmits (including the station under study):

$$p_b = 1 - \prod_{i=1}^{G} (1-\rho_i\tau_i)^{\alpha n_i}. \tag{7}$$

3.2.3. Packet service time

We now study how the packet service time T_j behaves and derive the expression for its probability distribution function. As we can deduce from the description of the MAC protocol in Section 2, there are four components contributing to the service time of a packet: (i) the total number of backoff slots the station has to wait before its packet is served (i.e., until its packet is either successfully transmitted or eventually dropped), (ii) the total amount of time the backoff counter at the station is kept frozen because of successful packet transmissions and/or collisions among the other stations that contend for the channel, (iii) the

[3] Observe that the case $G = 1$ corresponds to a homogeneous scenario where all users are identical.

[4] In this paper if X is a random variable, $\overline{X}$ is its average value. Further, when we write that two random variables are equal, we mean *equal in distribution*.

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

total amount of time lost due to collisions of the station's packet, and (iv) the time needed by the station to transmit its packet. We proceed by first analyzing each one of these components separately.

Denote by U a uniformly distributed random variable in $[0,1]$, and let $BO(\xi)$ be the random variable that denotes the value of the backoff counter at the station after its packet has collided ξ times. According to the standard [19], the value of $BO(\xi)$ is computed as follows:

$$BO(\xi) = \lfloor U \min(\alpha CW_{max}, 2^{\xi}\alpha CW_{min}) \rfloor = \lfloor \alpha CW_{min} U \min(2^{m}, 2^{\xi}) \rfloor. \quad (8)$$

Now, recall that the time slot in an α-scaled system has duration $\frac{\sigma}{\alpha}$, and that k_{max} is the maximum number of unsuccessful transmission attempts before a packet is dropped by a station. If the packet of the station has experienced a number $0 \leqslant k \leqslant k_{max}-1$ collisions before being served (i.e., it is served at the $k+1$ transmission attempt), the amount of time the station spent in decrementing its backoff counter, denoted by $T_j^d(k)$, is:

$$T_j^d(k) = \frac{\sigma}{\alpha} \sum_{\xi=0}^{k} BO(\xi). \quad (9)$$

Note that since the protocol timeouts are divided by α and the station transmission speed multiplied by α, a packet collision in an α-scaled system takes a time period of $\frac{T_{col}}{\alpha}$ and a successful packet transmission takes a time period of $\frac{T_{suc}}{\alpha}$, where T_{col}, T_{suc}, as given by Eqs. (1) or (2) depending on the access method used. T_{col} and T_{suc} can be random variables, whose distribution depends on the packet size distribution.

On each time slot the backoff counter at the station may freeze for a duration of $\frac{T_{suc}}{\alpha}$ due to a successful transmission from the other stations, an event which occurs with probability q_j (Eq. (6)). And, with probability $w_j = p_j - q_j$ (p_j given by Eq. (5)) the backoff counter freezes for a duration of $\frac{T_{col}}{\alpha}$ due to a collision among the other stations. Therefore, if the station's packet experienced $0 \leqslant k \leqslant k_{max}-1$ collisions before being served, the amount of time its backoff counter remained frozen, denoted by $T_j^f(k)$, is:

$$T_j^f(k) = \sum_{\xi=0}^{k} \sum_{i=1}^{BO(\xi)} \left(q_j \frac{T_{suc}(i)}{\alpha} + (p_j - q_j) \frac{T_{col}(i)}{\alpha} \right), \quad (10)$$

where $T_{suc}(i)$, $T_{col}(i)$ are respectively the values of the random variables T_{suc}, T_{col} on the ith successful transmission or collision among the other stations.

When the station's packet has experienced k collisions before being served, the total amount of time lost because of these collisions is $k\frac{T_{col}}{\alpha}$. The packet is served on its $k^{th}+1$ transmission attempt, which means that it is either successfully transmitted, requiring an additional $\frac{T_{suc}}{\alpha}$ amount of time, or dropped. Clearly, the packet is dropped only when $k = k_{max}-1$ and the next transmission attempt results in a collision, which requires $\frac{T_{col}}{\alpha}$ amount of time.

Let $T_j(k)$ be the packet service time at the station when its packet is successfully transmitted on the $1 \leqslant k+1 \leqslant k_{max}$ transmission attempt. Considering all the above four components that contribute to the service time of a packet, it is easy to see that:

$$T_j(k) = T_j^d(k) + T_j^f(k) + k\frac{T_{col}}{\alpha} + \frac{T_{suc}}{\alpha}. \quad (11)$$

Note that if a collision occurs at the packet's k_{max} transmission attempt, the packet is dropped and the service time denoted by $T'_j(k_{max}-1)$ is given by Eq. (11) after replacing $\frac{T_{suc}}{\alpha}$ with $\frac{T_{col}}{\alpha}$.

Now, $P(T_j(k) \leqslant t)$ is the probability distribution function of the random variable $T_j(k)$, and $P(T'_j(k_{max}-1) \leqslant t)$ is the probability distribution function of the random variable $T'_j(k_{max}-1)$. Removing the condition on the number of collisions k, we get the probability distribution function of the packet service time T_j:

$$P(T_j \leqslant t) = \sum_{k=0}^{k_{max}-1} P(T_j(k) \leqslant t)(p_j)^k(1-p_j) + (p_j)^{k_{max}} P(T'_j(k_{max}-1) \leqslant t). \quad (12)$$

And, clearly, the average packet service time $\overline{T_j}$ (used in Eq. (3)), is:

$$\overline{T_j} = \sum_{k=0}^{k_{max}-1} \overline{T_j(k)}(p_j)^k(1-p_j) + (p_j)^{k_{max}} \overline{T'_j(k_{max}-1)}. \quad (13)$$

3.3. Scaling properties of α-scaled systems

We are now ready to formally prove the following important properties about the scaling behavior of α-scaled systems: (i) the perceived performance of each user does not degrade as the scaling factor α (and hence the total number of users αN) increases, and (ii) the factor α by which the number of users increases is the minimum factor by which the system parameters should be scaled, in order not to degrade each user's perceived performance.

Before proceeding, we first derive approximate simplified expressions for Eqs. (5)–(7) and Eqs. (9) and (10), which we use. Our approximations for Eqs. (5)–(7) are based on the following two facts. First, on the exponential approximation $(1-\rho_j\tau_j)^{\alpha n_j} \approx e^{-\rho_j\tau_j\alpha n_j}$, $\forall j \in \{1,\ldots,G\}$, which holds even for relatively small values of the exponent αn_j, given that $\rho_j\tau_j \ll 1$ [28]. It is not hard to see that this last relation holds as long as CW_{min} and $m = \log_2\left(\frac{CW_{max}}{CW_{min}}\right)$ are not too small. One can also verify this by setting, for example, $CW_{min} = 32$ and $m = 5$ (i.e., the default values in the IEEE 802.11 standard [19]) and solving the system of Eqs. (4), (5) and (7) numerically. No matter what the rest of the system parameter values (α, G, n_j, ρ_j, $j \in \{1,\ldots,G\}$) are, the product $\rho_j\tau_j$ never exceeds 0.12, $\forall j \in \{1,\ldots,G\}$. And, the second fact that we use is that $n_j - \frac{2}{\alpha} \approx n_j - \frac{1}{\alpha} \approx n_j$, $\forall j \in \{1,\ldots,G\}$, which, of course, holds better as α increases.

3.3.1. Approximate expressions for the system behavior

From Eqs. (4)–(7) and the above approximations we can deduce the following:

$$p_j \approx 1 - \prod_{i=1}^{G} e^{-\frac{n_i\rho_i f(p_i,p_b,m)}{CW_{min}}}, \quad (14)$$

$$q_j \approx \sum_{i=1}^{G} \frac{n_i\rho_i f(p_i,p_b,m)}{CW_{min}} \prod_{k=1}^{G} e^{-\frac{n_k\rho_k f(p_k,p_b,m)}{CW_{min}}}, \quad (15)$$

$$p_b \approx p_j. \quad (16)$$

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

Further, from Eqs. (8)–(10) and the fact that $BO(\xi) = \lfloor \alpha CW_{min} U \min(2^m, 2^\xi) \rfloor \approx \alpha \lfloor CW_{min} U \min(2^m, 2^\xi) \rfloor$, it is easy to see that:

$$T_j^d(k) \approx \sigma \sum_{\xi=0}^{k} \lfloor CW_{min} U \min(2^m, 2^\xi) \rfloor, \tag{17}$$

$$\begin{aligned} T_j^f(k) &\approx \sum_{\xi=0}^{k} \sum_{i=1}^{\alpha \lfloor CW_{min} U \min(2^m, 2^\xi) \rfloor} \frac{1}{\alpha} \left(q_j T_{suc}(i) + (p_j - q_j) T_{col}(i) \right) \\ &= \sum_{\xi=0}^{k} \sum_{i=1}^{\lfloor CW_{min} U \min(2^m, 2^\xi) \rfloor} \sum_{l=\alpha i - \alpha + 1}^{\alpha i} \frac{1}{\alpha} \left(q_j T_{suc}(l) + (p_j - q_j) T_{col}(l) \right) \\ &= \sum_{\xi=0}^{k} \sum_{i=0}^{\lfloor CW_{min} U \min(2^m, 2^\xi) \rfloor - 1} \tau^\alpha(i), \end{aligned}$$

$$\text{where: } \tau^\alpha(i) = \frac{1}{\alpha} \sum_{l=\alpha i+1}^{\alpha i+\alpha} \left(q_j T_{suc}(l) + (p_j - q_j) T_{col}(l) \right). \tag{18}$$

Note that by substituting $T_j^d(k)$ from Eq. (17) and $T_j^f(k)$ from Eq. (18) into Eq. (11), we get the corresponding approximate expression for $T_j(k)$ (as well as for $T_j'(k_{max}-1)$ after replacing $\frac{T_{suc}}{\alpha}$ with $\frac{T_{col}}{\alpha}$ as before).

Now, observe that $T_j^d(k)$ is independent of α. Further, since $\overline{\tau^\alpha(i)} = q_j \overline{T_{suc}} + (p_j - q_j)\overline{T_{col}}$, $\forall i, \alpha$, the variable $\overline{T_j^f(k)}$ does not explicitly depend on α, but only implicitly, through the probabilities p_j and q_j. Therefore, we can write the following:

$$\begin{aligned} &\sum_{k=0}^{k_{max}-1} \left(\overline{T_j^d(k)} + \overline{T_j^f(k)} \right) (p_j)^k (1 - p_j) \\ &\quad + (p_j)^{k_{max}} \left(\overline{T_j^d(k_{max}-1)} + \overline{T_j^f(k_{max}-1)} \right) \\ &\quad = \psi(p_j, q_j, \sigma, \overline{T_{suc}}, \overline{T_{col}}, m, CW_{min}, k_{max}), \end{aligned} \tag{19}$$

where $\psi(\cdot)$ is a function of the variables p_j, q_j, σ, $\overline{T_{suc}}$, $\overline{T_{col}}$, m, CW_{min}, and k_{max}. Further, we can also write:

$$\begin{aligned} &\sum_{k=0}^{k_{max}-1} (k\overline{T_{col}} + \overline{T_{suc}})(p_j)^k (1 - p_j) + (p_j)^{k_{max}} k_{max} \overline{T_{col}} \\ &\quad = \omega(p_j, \overline{T_{suc}}, \overline{T_{col}}, k_{max}), \end{aligned} \tag{20}$$

where $\omega(\cdot)$ is a function of p_j, $\overline{T_{suc}}$, $\overline{T_{col}}$, and k_{max}.

From Eqs. (13), (11), (19) and (20), the average packet service time $\overline{T_j}$ can be written as[5]:

$$\overline{T_j} \approx \psi(p_j, q_j, \sigma, \overline{T_{suc}}, \overline{T_{col}}, m, CW_{min}, k_{max}) + \frac{1}{\alpha} \left(\omega(p_j, \overline{T_{suc}}, \overline{T_{col}}, k_{max}) \right). \tag{21}$$

Remark 1. The set of Eqs. (3), (14)–(16) and (21), implies that each user, irrespectively of the group j that he/she belongs to, sees approximately the same probabilities $p_j = p$, $q_j = q$ (for some p and q) and experiences approximately the same average packet service time $\overline{T_j} = T$ (for some T). What distinguishes users of different groups is their perceived packet throughput and queueing delay, since they may have different packet arrival processes and rates λ_j's ($j \in \{1, \ldots, G\}$). Recall that we have assumed a common packet size distribution for all users in the network. A similar analysis can be carried out for the case of different packet size distributions among different groups of users. The main difference in this case is that the packet service time among different groups can also be different. This does not affect the arguments that follow.

[5] It is a matter of simple algebra to write the expressions for the functions $\psi(\cdot)$ and $\omega(\cdot)$. We omit these calculations for brevity since, as it will become apparent shortly, they are not needed for proving our arguments.

3.3.2. Asymptotic behavior

We first study how the perceived performance of a station behaves as the scaling factor α (and hence the total number of competing stations αN) becomes large, i.e., technically, as $\alpha \to \infty$.

Lemma 1. *As the scaling factor α increases, the probabilities p_j, q_j, p_b, and the average packet service time $\overline{T_j}$ become independent of α, $\forall j \in \{1, \ldots, G\}$.*

Proof. From Eq. (21), as $\alpha \to \infty$, $\overline{T_j} \to \psi(p_j, q_j, \sigma, \overline{T_{suc}}, \overline{T_{col}}, m, CW_{min}, k_{max})$. Given this, and the set of Eqs. (3) and (14)–(16), we can deduce that p_j, q_j, p_b do not depend on α, and, in turn, so does $\overline{T_j}$. □

Corollary 1. *As the scaling factor α increases, the distribution of the service time T_j becomes independent of α, $\forall j \in \{1, \ldots, G\}$.*

Proof. From Eq. (11), as $\alpha \to \infty$, $T_j(k) \to T_j^d(k) + T_j^f(k)$, $\forall k$. The random variable $T_j^d(k)$ does not depend on α (Eq. (17)). Further, since in Eq. (18), $\lim_{\alpha \to \infty} \tau^\alpha(i) \to q_j \overline{T_{suc}} + (p_j - q_j)\overline{T_{col}}$, $\forall i$, by the Law of Large Numbers, and since p_j and q_j become independent of α from Lemma 1, so does the random variable $T_j^f(k)$. Therefore, the distribution of $T_j(k)$ becomes independent of α, $\forall k$, and so does the distribution of T_j (Eq. (12)). □

Theorem 1. *As the scaling factor α increases, the perceived performance (delay distribution and throughput) of a user becomes independent of α, and therefore, of the total number of users sharing the wireless channel (αN).*

Proof. Each station is a queueing system. The packet arrival process at the queue of this system (*e.g.* from upper layer protocols) remains unaltered as α increases, and the packet service time distribution becomes independent of α by Corollary 1. Therefore, the queue occupancy and queueing delay distributions also become independent of α, and hence, so does the user perceived delay distribution. Further, since the collision probability also becomes independent of α, and the maximum number of allowed collisions k_{max} before a packet is dropped remains unaltered, the user perceived throughput becomes independent of α. □

Remark 2. So far we have been assuming an infinite buffer size at the queue of each station. However, it is easy to see that Theorem 1 still holds even if this is not the case. In the infinite buffer case the probability that the queue occupancy exceeds some level, say b, becomes independent of the scaling factor α. If there is a finite buffer of size b, any excess workload over this level is lost. Hence, the

probability that a packet is lost due to a buffer overflow becomes independent of α, and therefore, the user perceived throughput does not depend on α here either. The same argument holds for the perceived delay as the queue occupancy remains independent of α.[6]

3.3.3. Pre-asymptotic behavior

We now turn our attention to the pre-asymptotic behavior of the system. First, observe that the packet delay fluctuations due to the term $\tau^{\alpha}(i)$ in Eq. (18) decrease with increasing α, $\forall i$, since the term converges by the Law of Large Numbers as explained in the proof of Corollary 1. Further, from Eqs. (3), (13), (11), (17), (18) and (14)–(16), we can see that the other difference as α increases is that the duration of a collision and of a successful transmission of a packet become smaller by the factor α. We can therefore state the following Theorem, whose proof follows immediately:

Theorem 2. *As the scaling factor α increases, the perceived performance (delay distribution and throughput) of a user does not degrade.*

Remark 3. As we will see in Section 4, the perceived performance of a user at higher loads remains virtually invariant. This is expected since at higher loads there is a large number of collisions and stations spent most of their time in backoff. As can be seen by Eq. (17) the one portion of the backoff time $T^{d}(k)$ is independent of α, $\forall k$. Further, the other portion $T^{f}(k)$ given by Eq. (18) becomes quickly independent of α, $\forall k$, as the term $\tau^{\alpha}(i)$ converges fast. This is because the events of packet collisions or successful transmissions on different time-slots are loosely correlated due the protocol's backoff mechanism. Since the time spent in backoff quickly becomes independent of α and dominates the packet service time at higher loads, user perceived performance remains virtually invariant. As we are moving to lower loads, user performance improves with increasing α as packet transmissions require less time.

Summarizing, the main intuition behind the scaling we perform is the following: while the number of competing stations increases by a factor α, the probability that each station transmits at some arbitrary slot decreases by the factor α, thus leaving the collision probability almost unaltered. This is accomplished via the scaling we perform to the minimum and maximum contention window sizes CW_{min} and CW_{max}, which regulate the transmission probability. However, while the transmission probability at each station decreases by α, we speed-up the system by scaling by the same factor α the protocol timeouts $\{\sigma, \mathrm{DIFS}, \mathrm{SIFS}\}$ and node transmission speeds C. This ensures that the actual time duration until a station successfully transmits its packet does not increase.

3.3.4. Minimum parameter scaling factor

An interesting question is what is the *minimum* parameter scaling factor. In our analysis above we have been scaling the system parameters by the factor α by which the number of stations in the WLAN increases. We now proceed to formally show that this is the *minimum* factor by which these parameters could be scaled in order to ensure that each user's perceived performance is never degraded.

First, let's consider the minimum and maximum contention window sizes, and suppose that these are scaled by a factor $\beta < \alpha$, that is, they become βCW_{min} and βCW_{max}. The following lemma states that the collision probability can increase.

Lemma 2. *If the number of users increases by α, but the minimum and maximum contention window sizes increase by $\beta < \alpha$, then the collision probability can increase.*

Proof. Suppose that we have only one group of stations, i.e., $G = 1$, and that each station is saturated, i.e., it always has a packet available for transmission. This means that the offered load for each station is $\rho_1 = 1$. Further, as before, $m = \log_2\left(\frac{\beta CW_{max}}{\beta CW_{min}}\right) = \log_2\left(\frac{CW_{max}}{CW_{min}}\right)$. From Eqs. (14) and (16) we can deduce that $\frac{\ln(1-p_1)}{f(p_1,m)} = -\frac{\alpha}{\beta}\frac{n_1}{CW_{min}}$. The left hand side of this relation decreases as p_1 increases. Increasing the ratio $\frac{\alpha}{\beta}$ decreases the right hand side of the relation, which therefore means that p_1 increases. □

What about the protocol timeouts and station transmission speeds? Suppose that the minimum and maximum contention window size become αCW_{min} and αCW_{max} in order not to increase the collision probability, but the protocol timeouts and station transmission speeds are scaled by some $\gamma < \alpha$, i.e., they become $\left\{\frac{\sigma}{\gamma}, \frac{\mathrm{DIFS}}{\gamma}, \frac{\mathrm{SIFS}}{\gamma}\right\}$, and γC. Then, the following lemma states that the backoff time, and in turn, the total packet service time at a station can increase.

Lemma 3. *If the number of users and the minimum and maximum contention window sizes increase by α, but the protocol timeouts and station transmission speeds are scaled by $\gamma < \alpha$, then the packet service time can increase.*

Proof. Suppose that the duration of packet transmissions is small (*e.g.* the packet sizes are small). Then, the time spent by a station in decrementing its backoff counter can dominate the total packet service time. Eq. (17), which gives the time duration for decrementing the backoff counter, will be multiplied by the factor $\frac{\alpha}{\gamma} > 1$, $\forall k$, and hence will increase. □

We can now state the following theorem, whose proof follows immediately from the above arguments:

Theorem 3. *If the number of users in an IEEE 802.11 WLAN increases by a factor $\alpha > 1$, then the minimum factor by which the system parameters should be scaled in order to ensure that each user's perceived performance is not degraded is equal to α.*

4. Simulations

In this section, we perform experiments with the ns-2 simulator [22] in order to verify our theoretical arguments.

[6] Note that the probability that a packet is lost due to a buffer overflow in a system with a buffer of size b, is not in general equal to the probability that the queue occupancy exceeds b in an infinite buffer system and we do *not* make such an assumption here.

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

Table 1
System parameters.

Transmission rate (C)	1 Mbps
DIFS	50 μs
SIFS	10 μs
Slot time (σ)	20 μs
CW_{min}	31
CW_{max}	1023

The ns-2 simulator provides one of the most accurate IEEE 802.11 MAC-layer implementations [22], and it is perhaps the most popular simulator for wireless network performance evaluation.

We consider four groups of stations, grp_1, grp_2, grp_3, and grp_4. Stations within each group are uniformly distributed around a base-station/access-point, and generate traffic destined to it according to a Poisson process. The corresponding packet arrival/generation rates for stations of each group are: $\lambda_1 = \frac{1}{0.25T_p}$, $\lambda_2 = \frac{1}{0.5T_p}$, $\lambda_3 = \frac{1}{1.5T_p}$, and $\lambda_4 = \frac{1}{2T_p}$, all expressed in packets/sec. T_p is a parameter that we vary from 8 ms (high network load) to 30 ms (low network load). Further, we set the packet size (in bytes) for $grp_1, \ldots, grp_4$ as $L_1 = 125$, $L_2 = 250$, $L_3 = 750$, and $L_4 = 1000$, respectively. The buffer size at the interface transmission queue of each station can hold 250 packets. The initial number of stations in each group $j \in \{1, \ldots, 4\}$ is $n_j = 1$, and we present results for scenarios where this number scales by $\alpha = 1$, 4, 16, and 32, i.e., when the total number of stations in the system is $N = 4$, 16, 64, and 128. We scale the system parameters as described earlier. The initial values for these parameters, i.e., before performing any scaling, are the ones used by default in the ns-2 simulator, which correspond to the default values in the IEEE 802.11b standard [20] and shown in Table 1. We present results for both the basic and the RTS/CTS access methods.

Fig. 2 shows how the average station throughput in an α-scaled system behaves as we vary the parameter T_p, i.e., as we vary the network load (from high to low). In Fig. 3, we provide a finer-grain view of the throughput behavior, by presenting the throughput for stations that belong to grp_1, grp_2, grp_3, and grp_4.

We observe that the user perceived throughput remains virtually invariant as the parameter α (and hence the total number of competing stations) increases, even for small values of α's, and on a per-group basis. This is in accordance to our theoretical arguments in Section 3.3.3. Recall that at high loads the packet service time is dominated by the backoff time, which quickly becomes independent of α. This means that both the queue length distribution and packet drops (that are either due to collisions or to buffer overflows) also become quickly independent of α. At low loads, where there are very few drops (primarily due to collisions) or no drops at all, the packet service time is dominated by the packet transmission time, which decreases by α. In both cases the user perceived throughput should remain approximately the same as α increases, as shown in the plots.[7]

[7] The reason that different groups have a different throughput behavior is because their packet arrival rates and sizes are different.

Fig. 4 shows how the packet drop ratio behaves. The packet drop ratio is the percentage of packets that are dropped in the network, either at the MAC layer due to collisions, or because of buffer overflows. In Fig. 5, we provide a more detailed view of the packet drop ratio behavior, by presenting the percentage of dropped packets for stations that belong to grp_1, grp_2, grp_3, and grp_4. For the same reasons as before, the packet drop ratio remains virtually invariant as α increases, and this also holds on a per-group basis.

Fig. 6 shows how the average packet delay in an α-scaled system behaves, which includes *both* queueing delay and packet service time, across all successfully transmitted packets in the network. In Fig. 7, we also see that similar results hold for the average delay of packets for stations of different groups. We observe that the delay remains almost invariant as α increases. At low network loads it becomes slightly better, since as mentioned earlier, what dominates the packet service time is the packet transmission time, which decreases by the factor α. For example, in the basic access method, when $T_p = 26$ ms, the average packet delays (across all successfully transmitted packets in the network) for $\alpha = 1$, 4, 16, and 32, are respectively 0.24, 0.15, 0.13, and 0.08 ms. For the RTS/CTS method, when $T_p = 30$ ms, the average packet delays for $\alpha = 1$, 4, 16, and 32, are respectively 0.23, 0.12, 0.10, and 0.08 ms. At higher network loads the average packet delay is approximately the same for all values of α.

Finally, in Figs. 8 and 9 we present packet delay distributions (across all successfully transmitted packets in the network) for the two access methods at two different load values. Similar results hold for all other loads, and support again our theoretical arguments. At higher loads the delay distribution is approximately the same for all values of α. At lower loads it never gets worse, but instead improves with increasing α, having a converging behavior.

Summarizing, all of our experiments are in agreement with our theoretical arguments of Section 3. In particular, the perceived performance of each station never gets worse as the scaling factor α (and hence the total number of users) increases. The throughput and packet drop ratio remain almost invariant. The delays also remain invariant, especially for higher network loads, whereas they improve with increasing α at lower network loads.

5. Discussion

5.1. Impact of non-ideal channel

Recall that in our analysis in Section 3 it is assumed that the channel is perfect. However, when the channel is not perfect, *e.g.* when fading is figured in, packet losses are no longer due to collisions only, but they may well be caused by channel fading. The 802.11 responds in the same way when a packet is lost, no matter whether this is due to collision or channel fading. Practically, it is extremely difficult to distinguish these two causes. However, we can incorporate the packet error probability into the collision probability as the study in [12] did, and all our analytical

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

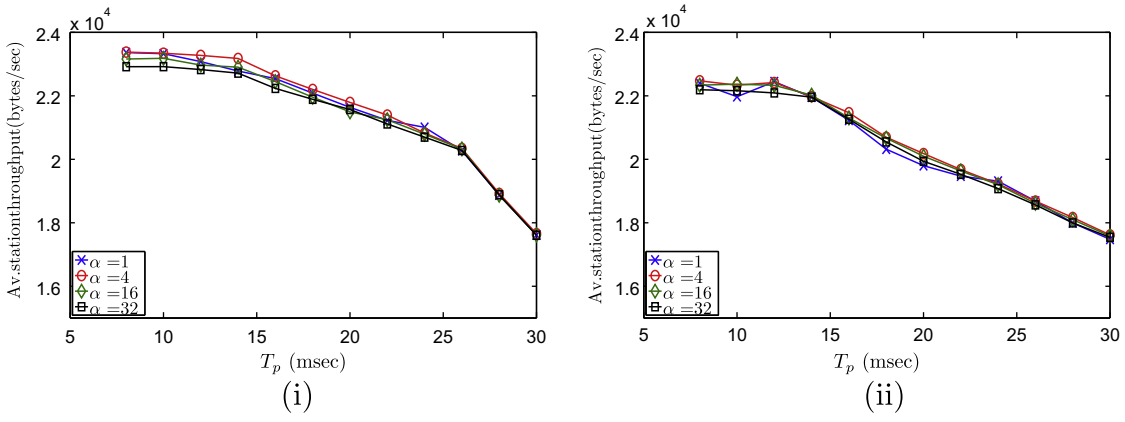

Fig. 2. Average station throughput for different scaling factors α: (i) basic access method, and (ii) RTS/CTS access method.

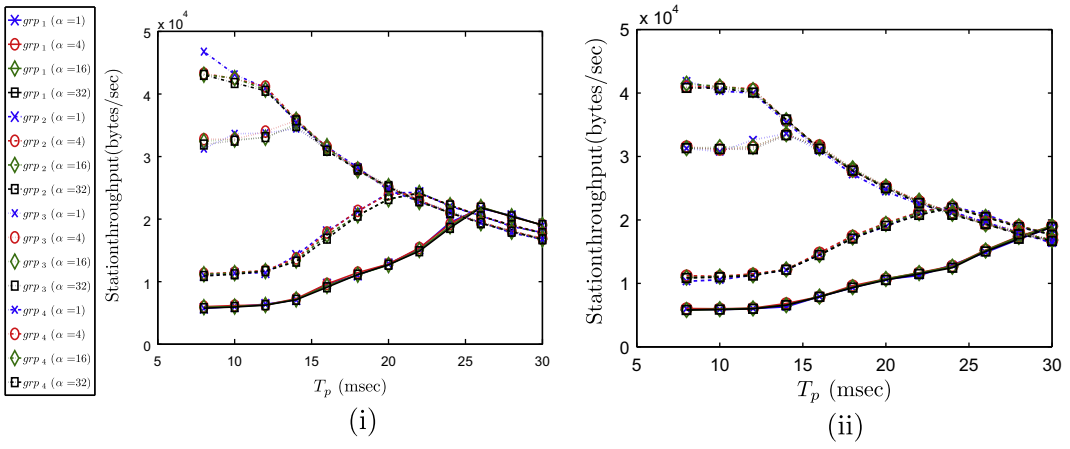

Fig. 3. Throughput for stations of different groups for different scaling factors α: (i) basic access method, and (ii) RTS/CTS access method.

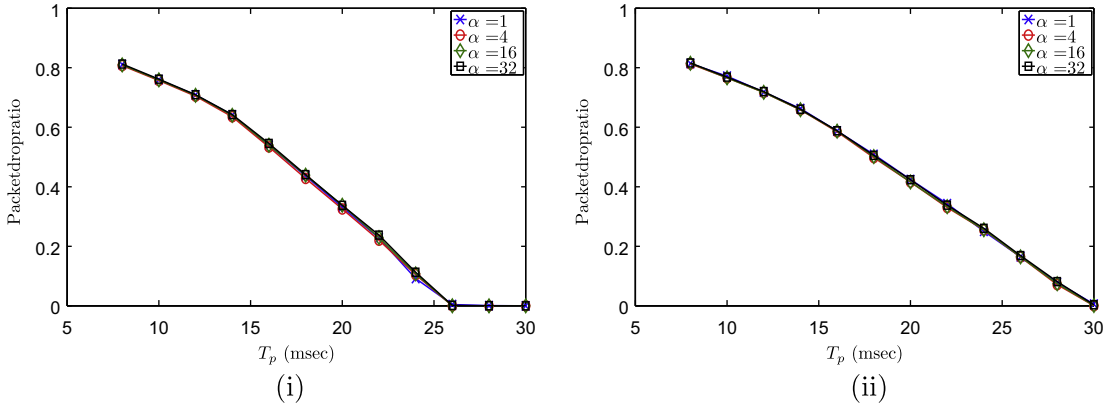

Fig. 4. Packet drop ratio for different scaling factors α: (i) basic access method, and (ii) RTS/CTS access method.

results and arguments still hold. But note that normally WLANs feature low node mobility and relatively stable channels, and packet losses due to errors are not a serious problem anyway.

5.2. Impact of hidden terminals

Recall that we have also ignored the hidden terminal problem [16], as in a typical WLAN environment every sta-

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

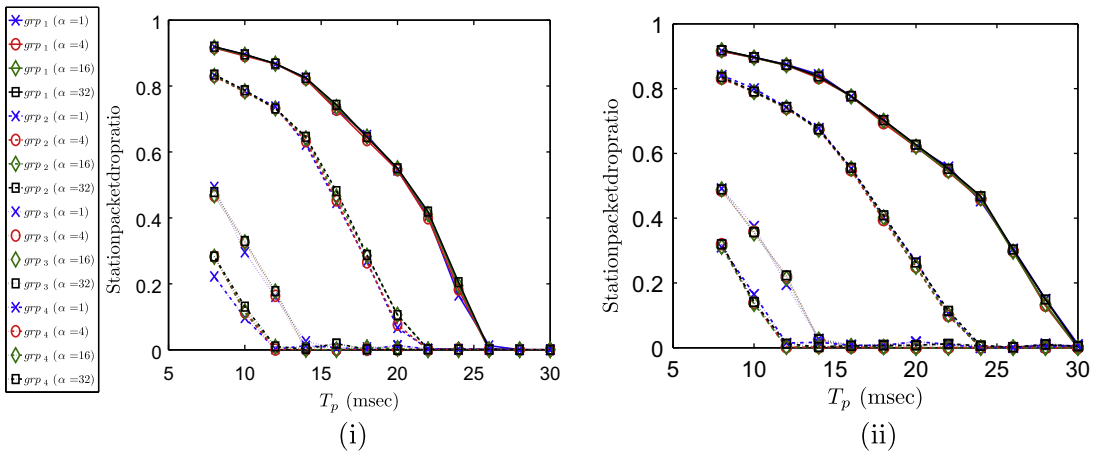

Fig. 5. Packet drop ratio for stations of different groups for different scaling factors α: (i) basic access method, and (ii) RTS/CTS access method.

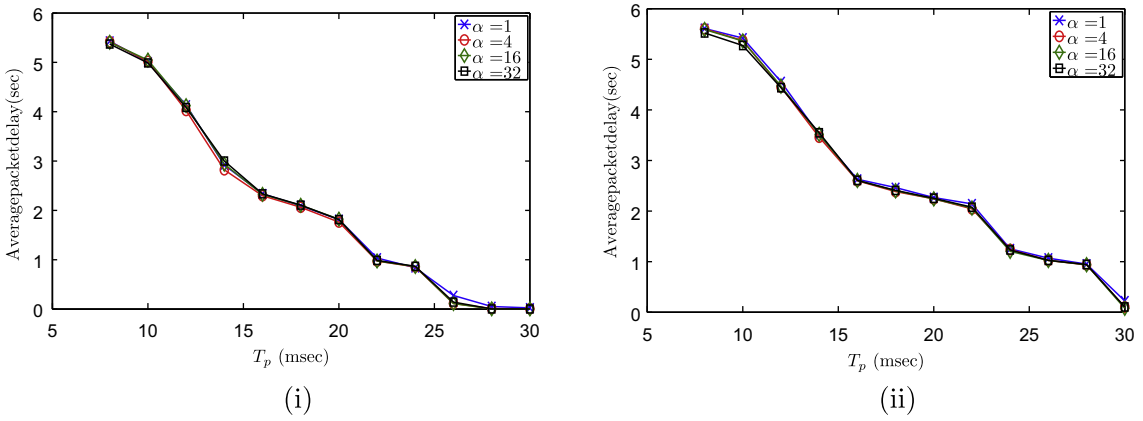

Fig. 6. Average packet delay for different scaling factors α: (i) basic access method, and (ii) RTS/CTS access method.

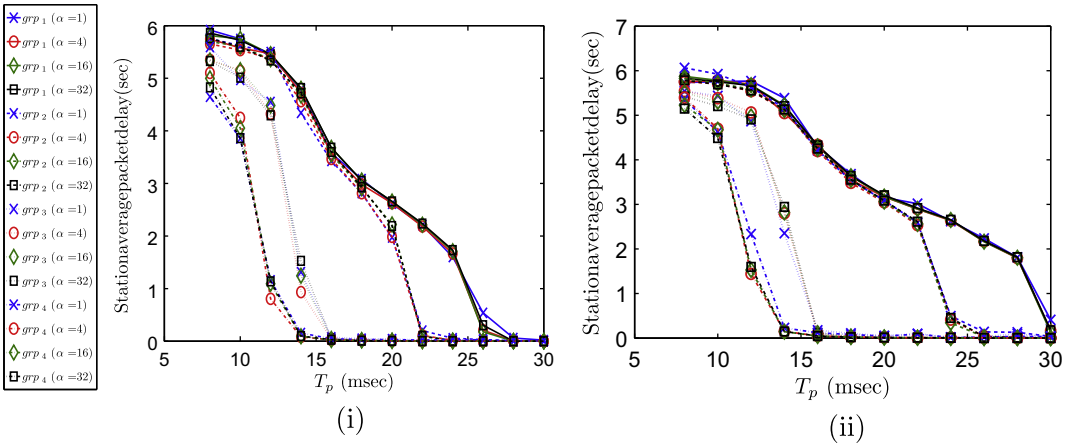

Fig. 7. Average packet delay for stations of different groups for different scaling factors α: (i) basic access method, and (ii) RTS/CTS access method.

tion can sense all the other stations' transmissions [38], especially when the RTS/CTS access method is used [16]. Our results however hold even if this is not the case, i.e., even if there is a large number of hidden terminals such that their impact to network performance cannot be ignored. The analysis however becomes significantly more

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

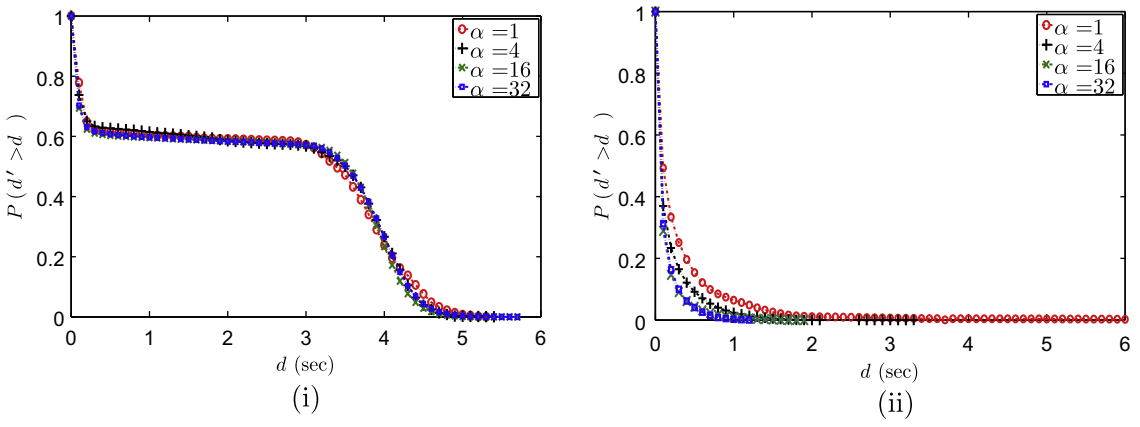

Fig. 8. Packet delay distribution for different scaling factors α: (i) $T_p = 16$ ms (high load), and (ii) $T_p = 26$ ms (low load) (basic access method).

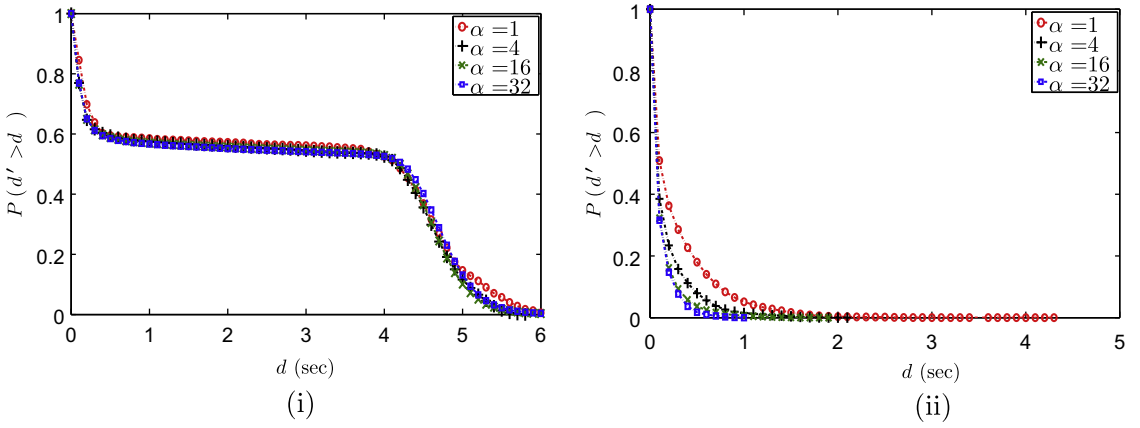

Fig. 9. Packet delay distribution for different scaling factors α: (i) $T_p = 16$ ms (high load), and (ii) $T_p = 30$ ms (low load) (RTS/CTS access method).

involved and we leave it for future publication. For example, when the basic access method is used, hidden stations do not sense the transmission of a station to the access point until they sense the corresponding ACK from the access point to the station. So they will sense the channel as idle during this time period, and if any of these stations completes its backoff procedure it will send another packet to the access point. This packet will collide with the packet from the transmitting station. An extended analysis should also incorporate possible collisions within this *vulnerable* time period due to hidden stations.

5.3. Accelerating simulations

Notice that in this paper we were starting from smaller networks and moving to larger networks, i.e., we have been studying the network behavior as the number of users increases, that is, when the scaling factor is some $\alpha > 1$. One can also move the opposite direction, i.e., downscale large networks, by setting $\alpha < 1$. It is easy to see that as long as the factor α is not too small, one can accurately predict the performance of larger networks from scaled-down replicas that consist of fewer stations/nodes. This is important for simulations and experiments with testbeds where one could experiment with network miniatures, which are much easier to manage, and have much lower computational requirements and costs. For example, Fig. 10 shows the time needed to complete a simulation experiment as a function of the number of stations. We see that this time grows between linear and exponential with the number of nodes. One can therefore simulate fewer nodes to significantly expedite simulations, by scaling the system as described in this paper, using $\alpha < 1$. The performance of the smaller and larger network will be virtually the same, especially for congested scenarios, as demonstrated in Section 4. How small the (down)scaling factor α can get, while keeping the accuracy in performance prediction high, is an interesting open question.

5.4. Practical applications and considerations

Our findings in this paper can also help in designing improved versions of 802.11-based WLANs that can support a large number of users. As discussed at the end of Section 2.1, scaling the protocol timeouts $\{\sigma, \text{DIFS}, \text{SIFS}\}$ and station transmission speeds C has been the trend in recent

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

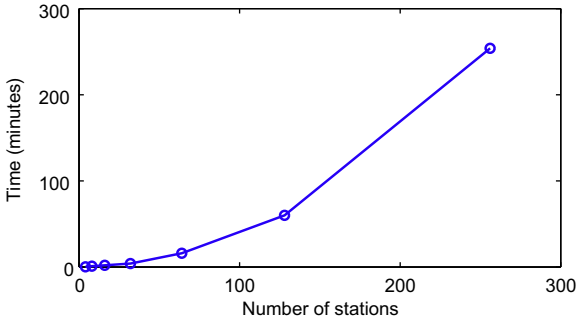

Fig. 10. Simulation time vs. number of stations.

versions of the IEEE 802.11 standard, in an attempt to provide higher data rates and improve performance [2]. In this paper, we have rigorously established the connection between the scaling factor of these parameters and the increase in the total number of users, so that per user perceived performance does not degrade. In addition to the above parameters, we have been also scaling the minimum and maximum contention window sizes CW_{min} and CW_{max} by the same factor. Therefore, given trends in the increase of the population of users, *e.g.* in wireless hotspots, our results give clear guidelines of how the IEEE 802.11 protocol should be scaled so that user performance does not get worse. Given that we cannot arbitrarily reduce the protocol timeouts and increase node transmission speeds, as this is constraint by the technology currently available, our results are of special interest, as they identify the minimum required amount for scaling.

Finally, one has to keep in mind that while the minimum and maximum contention window sizes can be easily scaled, as mentioned, the station transmission speed as well as the protocol timeouts depend in practice on lower network layer functionalities, *e.g.*, modulation techniques at the physical layer, MAC-layer hardware processing times, etc., and different amendments of the original IEEE 802.11 standard use different techniques to allow improvement of these parameters [2]. For example, the slot time duration σ should be in practice larger than the sum of the MAC-layer processing time and the air propagation time (<1 μs). Therefore, to support smaller slot time durations, techniques to achieve faster processing times are needed. Further, as the slot time duration cannot get arbitrarily small, i.e., smaller than the air propagation time, the maximum scaling factor α that we could ever have in practice is bounded.

6. Conclusion and future work

In this paper, we have studied some important scaling properties of today's 802.11-based WLANs. In particular, we have identified a set of protocol and system parameters that if scaled as the number of users sharing the wireless channel increases, ensures that the perceived performance of each individual user is not degraded. We have also established the exact minimum amount of scaling that it is required for these parameters in order to accomplish this. Interestingly enough, we have found that a scaling factor which is equal to the factor by which the number of users increases is sufficient to preserve performance. Our results set guidelines for designing future versions of 802.11-based WLANs that can efficiently support a very large number of users. Our findings can also have other applications, such as accelerating simulations and experiments with testbeds by downscaling the original larger networks.

One of the most interesting, yet challenging, future work directions is to investigate whether similar scaling properties hold for multi-hop wireless networks, which can be either static or mobile. Changing the number of nodes in such networks changes the graph connectivity structure, and in turn, the way the traffic flows into the network.

Acknowledgement

I thank Konstantinos Psounis for useful discussions and suggestions.

References

[1] I. Ada, C. Castelluccia, Differentiation mechanisms for IEEE 802.11, in: Proceedings of the IEEE INFOCOM, 2001.
[2] IEEE 802.11-2007: Standard for LAN/MAN – Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. <http://www.standards.ieee.org/getieee802/802.11.html>.
[3] G. Anastasi, E. Borgia, M. Conti, E. Gregory, Wi-fi in ad hoc mode: a measurement study, in: Proceedings of the IEEE PerCom, 2004.
[4] G. Anastasi, E. Borgia, M. Conti, E. Gregory, IEEE 802.11b ad hoc networks: performance measurements, Journal of Cluster Computing 8 (2–3) (2005).
[5] A. Balachandran, G.M. Voelker, P. Bahl, P.V. Rangan, Characterizing user behavior and network performance in a public wireless LAN, in: Proceedings of the ACM SIGMETRICS, 2002.
[6] G. Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function, IEEE Journal on Selected Areas in Communications 18 (3) (2000).
[7] G. Bianchi, I. Tinnirello, Remarks on IEEE 802.11 DCF performance analysis, IEEE Communications Letters 9 (8) (2005).
[8] F. Cali, M. Conti, IEEE 802.11 protocol: design and performance evaluation of an adaptive back-off mechanism, IEEE Journal on Selected Areas in Communications 18 (9) (2000).
[9] F. Cali, M. Conti, E. Gregori, P. Aleph, Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit, IEEE/ACM Transactions on Networking 9 (6) (2000).
[10] P. Chatzimisios, V. Vitsas, A.C. Boucouvalas, M. Tsoulfa, Achieving performance enhancement in IEEE 802.11 WLANs by using the DIDD backoff mechanism, International Journal of Communication Systems 20 (1) (2007).
[11] C.H. Foh, J. Tantra, Comments on IEEE 802.11 saturation throughput analysis with freezing of backoff counters, IEEE Communications Letters 9 (2) (2005).
[12] Z. Hadzi-Velkov, B. Spasenovski, Saturation throughput-delay analysis of IEEE 802.11 DCF in fading channel, in: Proceedings of the IEEE ICC, 2003.
[13] F. Hung, I. Marsic, Access delay analysis of IEEE 802.11 DCF in the presence of hidden stations, in: Proceedings of the IEEE GLOBECOM, 2007.
[14] JIWIRE. <http://www.jiwire.com/press-100k-hotspots.htm> (accessed September 2008).
[15] H. Kim, J.C. Hou, Improving protocol capacity with model-based frame scheduling in IEEE 802.11-operated WLANs, in: Proceedings of the ACM MOBICOM, 2003.
[16] J.F. Kurose, K. Ross, Computer Networking: A Top-Down Approach Featuring the Internet, Addison-Wesley Longman Publishing Co. Inc., Boston, MA, USA, 2002.
[17] Y. Kwon, Y. Fang, H. Latchman, A novel MAC protocol with fast collision resolution for wireless LANs, in: Proceedings of the IEEE INFOCOM, 2003.

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005

[18] A. Lindgren, A. Almquist, Quality of Service Schemes for IEEE 802.11 – A Simulation Study, Master's Thesis, Lulea University of Technology, May 2001.
[19] IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ISO/IEC 8802-11: 1999(e), August 1999.
[20] Supplement to 802.11-1999, Wireless LAN MAC and PHY Specifications: Higher Speed Physical Layer (PHY) Extension in the 2.4 GHz Band. <http://www.standards.ieee.org>.
[21] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications – Amendment 4: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band. <http://www.standards.ieee.org/>.
[22] Network Simulator. <http://www.isi.edu/nsnam/ns>.
[23] R. Pan, B. Prabhakar, K. Psounis, D. Wischik, SHRiNK: enabling scaleable performance prediction and efficient simulation of networks, IEEE/ACM Transactions on Networking 13 (5) (2005).
[24] F. Papadopoulos, K. Psounis, Predicting the performance of mobile ad hoc networks using scaled-down replicas, in: Proceedings of the IEEE ICC, 2007.
[25] F. Papadopoulos, K. Psounis, Scaling properties of IEEE 802.11 wireless networks, in: Proceedings of the WiOpt, 2008.
[26] F. Papadopoulos, K. Psounis, R. Govindan, Performance preserving topological downscaling of internet-like networks, IEEE Journal on Selected Areas in Communications 24 (12) (2006).
[27] E. Pelletta, H. Velayos, Performance measurements of the saturation throughput in IEEE 802.11 access points, in: Proceedings of the WiOpt, 2005.
[28] W. Rudin, Principles of Mathematical Analysis, McGraw-Hill, 1976.
[29] L. Scalia, I. Tinnirello, A low-level simulation study of prioritization in IEEE 802.11e contention-based networks, in: Proceedings of the IEEE COMSWARE, 2006.
[30] G. Sharma, A. Ganesh, P. Key, Performance analysis of contention based medium access control protocols, in: Proceedings of the IEEE INFOCOM, 2006.
[31] Y. Su, T. Gross, Validation of a miniaturized wireless network testbed, in: Proceedings of the ACM WiNTECH, 2008.
[32] O. Tickoo, B. Sikdar, Queueing analysis and delay mitigation in IEEE 802.11 random access MAC based wireless networks, in: Proceedings of the IEEE INFOCOM, 2004.
[33] A.L. Toledo, T. Vercauteren, X. Wang, Adaptive optimization of IEEE 802.11 DCF based on bayesian estimation of the number of competing terminals, IEEE Transactions in Mobile Computing 5 (9) (2006).
[34] H.L. Vu, T. Sakurai, Accurate delay distribution for IEEE 802.11 DCF, IEEE Communications Letters 10 (4) (2006).
[35] H. Wu, F. Zhu, Q. Zhang, Z. Niu, Analysis of IEEE 802.11 DCF with hidden terminals, in: Proceedings of the IEEE GLOBECOM, 2006.
[36] R.W. Wolff, Stochastic Modeling and the Theory of Queues, Prentice Hall, 1989.
[37] H. Wu, S. Cheng, Y. Peng, J.M.K. Long, IEEE 802.11 distributed coordination function (DCF): analysis and enhancement, in: Proceedings of the IEEE ICC, 2002.
[38] H. Zhai, X. Chen, Y. Fang, How well can the IEEE 802.11 wireless LAN support quality of service?, IEEE Transactions on Wireless Communications 4 (6) (2005)
[39] H. Zhai, Y. Kwon, Y. Fang, Performance analysis of IEEE 802.11 MAC protocols in wireless LANs, Wireless Communications and Mobile Computing 4 (8) (2004).
[40] E. Ziouva, T. Antonakopoulos, CSMA/CA performance under high traffic conditions: throughput and delay analysis, Computer Communications 25 (3) (2002).

Fragkiskos Papadopoulos is a visiting lecturer of the Electrical and Computer Engineering department at the University of Cyprus. He received the Diploma in Electrical and Computer Engineering from the National Technical University of Athens, Greece, in 2002. In 2004 and 2007, he received respectively the M.S. and Ph.D. degrees in Electrical Engineering from the University of Southern California, LA. During 2007–2009, he was a postdoctoral research scholar at the Cooperative Association for Internet Data Analysis (CAIDA), at the University of California–San Diego. He models and analyzes the performance of computer networks, and designs scalable methods and algorithms to solve problems related to such systems.

Please cite this article in press as: F. Papadopoulos, On scaling the IEEE 802.11 to facilitate scalable wireless networks, Comput. Netw. (2010), doi:10.1016/j.comnet.2010.02.005