Extended EDCA for Delay Guarantees in Wireless Local Area Networks

Gennaro Boggia, Pietro Camarda, Luigi Alfredo Grieco, and Giuseppe Piro

Abstract

Despite of their very broad diffusion, IEEE 802.11 Wireless Local Area Networks are not able to provide service differentiation and to support real-time multimedia applications, due to their channel access methods. To overcome these limitations, the 802.11e working group has proposed the Enhanced Distributed Coordination Access (EDCA) scheme which achieves service differentiation on statistical basis by properly mapping user QoS requirements to channel contention parameters. Such a scheme will be included in the emerging 802.11n standard and in the revision of the 802.11 standard. However, it has been widely demonstrated that, especially at high network loads, EDCA does not provide an effective usage of the channel capacity. In particular, it is unable to provide a bounded delay service to all kind of multimedia flows because flows with lower channel access priorities are starved to advantage only the ones with the highest priority. To fix this undesired behavior and improve wireless LAN performance, this paper proposes the new Extended EDCA (E^{2}DCA) scheme, that is compliant with 802.11e specifications. By exploiting a closed loop control algorithm, E^{2}DCA performs a distributed dynamic bandwidth allocation, providing guarantees on average/absolute delays to real-time media flows, regardless of their priorities. Moreover, an innovative Call Admission Control procedure has been developed. Using the *ns*-2 simulator, the effectiveness of the algorithm has been investigated in realistic network scenarios, involving a mix of audio, video, and FTP flows, at several network loads and with random losses. Results have shown that the proposed scheme is able to provide a bounded delay service to multimedia flows in a wide range of network loads and frame loss ratios.

Index Terms

QoS, Wireless LAN, Real-time Applications, Distributed Bandwidth Allocation.

A short preliminary version of this paper was presented at ACM/IEEE MSWiM'06 Conference.

Authors are with the "Dipartimento di Elettrotecnica ed Elettronica - Politecnico di Bari". Via Orabona, 4 – 70125 BARI, Italy. E-mail: {g.boggia, camarda, a.grieco, g.piro}@poliba.it.

I. INTRODUCTION

IEEE 802.11 Wireless Local Area Networks (WLANs) are widely employed for ensuring ubiquitous networking due to their easy installation, flexibility, and robustness against failures [1]. Despite of its very broad diffusion, the 802.11 Medium Access Control (MAC) cannot support real time applications, characterized by strict constraints on packet delay and jitter [2]-[4]. In fact, the 802.11 MAC employs a mandatory contention-based channel access scheme, known as Distributed Coordination Function (DCF), which, being based on CSMA/CA [5], does not provide guaranteed services.

To overcome this limitation, the 802.11e working group proposed some innovations: the Hybrid Coordination Function (HCF), which is an enhanced access method; specific signaling messages for service request and Quality of Service (QoS) level negotiation; four Access Categories (ACs) with different priorities to map users QoS requirements [6]. In particular, the HCF is in charge of assigning TXOPs (Transmission Opportunities) to each AC in order to satisfy its QoS needs, where TXOP is defined as the time interval during which a station has the right to transmit and it is characterized by a starting time and a maximum duration.

Such modifications to MAC layer are included in the ongoing new revision of the 802.11 standard [7] which has been conceived to incorporate 802.11a [8], 802.11b [9], and 802.11g [10] standards and several other 802.11 amendments (e.g., 802.11e standard [6]).

Moreover, it is worthwhile to note that the importance of HCF scheme is not only restricted to 802.11a/b/g WLANs. In fact, the emerging IEEE 802.11n standard is designed for achieving a data rate up to 100 Mbps by considering a new physical layer and by using just HCF as access function at MAC layer [11].

In HCF, time is divided into repeated periods (i.e., the SuperFrames) and two new channel access methods are introduced: the Enhanced Distributed Coordination Access (EDCA), which is a contention-based access scheme; the HCF Controlled Channel Access (HCCA), which requires a centralized controller, generally located at the Access Point (AP).

EDCA operates as the basic DCF method [1], but using different contention parameters per AC. A queue is associated to each AC at any station with QoS capabilities (i.e., QoS station, QSTA), acting as a virtual station with its own QoS parameters. Each queue within a station contends for a TXOP and defers its transmission until the channel is sensed idle for a time interval, known as Arbitration Interframe Space (AIFS), plus an additional random backoff time, multiple of a slot time by an integer taken from an uniform distribution in the interval from 0 to the *Contention Window* (CW). For each class AC_i, a contention window CW_i and an AIFS_i are defined as shown in Table I[1] [12]. If several backoff timers reach zero within the same station at the same time slot, then the highest priority frame will be transmitted and any lower priority frame will be deferred with a retry procedure, modifying the backoff timer [2], [6].

TABLE I

EDCA CONTENTION PARAMETERS

Designation	AC	Minimum CW	Maximum CW	AIFS
Background	AC_BK	CW_{min}	CW_{max}	7
Best Effort	AC_BE	CW_{min}	CW_{max}	3
Video	AC_VI	$\frac{CW_{min}+1}{2} - 1$	CW_{min}	2
Voice	AC_VO	$\frac{CW_{min}+1}{4} - 1$	$\frac{CW_{min}+1}{2} - 1$	2

A substantial difference with respect to DCF is that a virtual station can transmit more than one frame when it gains the access to the wireless medium. In particular, a virtual station will occupy the channel for a TXOP, whose maximum duration limit is advertised by the AP in the beacon frame at the beginning of each superframe. If this advertised maximum TXOP limit is zero, only one frame can be transmitted during each TXOP.

The EDCA scheme statistically pursues a service differentiation among traffic streams [13], but tuning its parameters to provide prioritization of ACs is a research topic [14]. In [15], a method for setting EDCA parameters in order to provide throughput guarantees has been described. Regarding the goal of providing delay guarantees, several papers have pointed out that the EDCA can provide a real-time service to highest priority flows, at the price of starving flows with lower priority, especially at high network load [16]-[18]. Moreover, it can provide only a relative differentiation among service classes, but not absolute guarantees on throughput/delay performance [13], [19].

[1]The contention window limits CW_{min} and CW_{max} in Table I, are not fixed, as with DCF, but can be variable and are assigned by a management entity or by an AP.

To overcome these limitations, adaptive algorithms that dynamically tune EDCA parameters have been proposed in [20]-[26]. In [27] the Enhanced Distributed Contention Control is proposed, which mitigates the contention level on the channel by exploiting an estimate of channel congestion level; it improves EDCA performance at high loads. In [28] a dynamic mechanism is proposed to adapt data rate and priority of multimedia wireless stations equipped with IEEE 802.11e network cards. In [29] an adaptive transmission probability has been introduced for to ensures the QoS requirements of higher priority classes at high traffic load conditions. In [30] a deterministic priority channel access has been proposed. The proposed scheme uses a busy tone to limit the transmission of lower priority traffic when higher priority traffic has packets to send.

Several schemes have been studied in literature to define TXOP limit over EDCA. In [31] a Surplus TXOP Diverted (STXD) scheme has been proposed to define the TXOP limit for per-flow but not for per-ACs. The fundamental of STXD is to regulate TXOP limit according to per-flow behavior to provide absolute delay guarantee of VBR flows. In [32] a distribution of the multimedia frame size is used off-line to dimension the TXOP limit in order to minimize the packet delays. In [33] a dynamic TXOP scheme has been proposed to adjusts the TXOP limits according to the state of transmission queue length. When the queue length is below a given threshold, the TXOP limit is set at the default one; on the other hand, if the queue length exceeds the threshold, the algorithm sets a new larger TXOP limit value than the default one to better manage a bursty traffic.

It is important to note that the summarized approaches have the main limitations of being not compliant with the standard (such as those proposed in [31]-[33]), or of being proved only using simulations without giving any theoretical bounds on their performance in a general scenario (such as those proposed in [20]-[30]).

This paper proposes the new Extended EDCA (E^{2}DCA) scheme (compliant with 802.11e specifications) which performs a distributed dynamic bandwidth allocation among wireless stations, providing to real-time media flows guarantees on average and absolute delays. In particular, each station evaluates its bandwidth needs by exploiting a closed loop control scheme which uses the transmission queue length as feedback signal. It is worth to note that E^{2}DCA requires that each node of the WLAN runs a linear control algorithm with a very limited complexity; thus, it is well suited for providing QoS in WLAN. Moreover, a Call Admission Control (CAC) algorithm that works jointly to E^{2}DCA has been designed. The main properties of E^{2}DCA have been theoretically investigated and bounds on both average and maximum packet delay as a function of protocol parameters have been derived.

Moreover, the *ns*-2 simulator [34] has been used to analyze the effectiveness of the proposed new scheme in realistic network scenarios, involving a mix of audio, video, and FTP flows, at several network loads and considering a noisy channel with random losses. Simulation results have shown that E^{2}DCA is able to provide bounded delays to multimedia flows in a wide range of network loads and frame loss ratios whereas standard EDCA is not able to provide service differentiation, especially at high network loads.

The rest of the paper is organized as follows: in Section II the E^{2}DCA algorithm is proposed; Section III shows simulation results; Section IV draws the conclusions.

II. The Extended EDCA Scheme

The IEEE 802.11e standard does not specify an effective way to evaluate TXOPs when a QSTA acquires the access to the wireless channel; it just specifies some superior limits advertised by the access point in a system variable, i.e., the *dot11EDCATableTXOPLimit* variable [6]. Instead, with E^{2}DCA, we provide QSTAs with a systematic way to choose TXOP after a successful contention on the wireless channel getting the right to transmit. We underline that the developed algorithm is fully compliant with the 802.11e standard.

In E^{2}DCA, after a successful channel contention, a QSTA can transmit multiple frames during its TXOP. Such a TXOP is computed by taking into account the transmission queue length of the QSTA in order to provide delay guarantee to each real-time flow.

To illustrate the E^{2}DCA scheme, we will refer to a WLAN system made by an access point and a set of QSTAs, each one with up to 4 queues, one for each AC introduced in 802.11e standard. As said before, each queue behaves like a virtual station with its own contention parameter.

A. The System Model

In our system, we suppose that N active traffic flows share a communication channel. Associated to each of these flows, there is a queue, where Protocol Data Units are stored waiting for transmission. E^{2}DCA scheduler evaluates the transmission needs of each queue in terms of TXOPs.

We define the time instant $t_{k,i}$ as the starting time of $TXOP_i(k)$, that is the k-th transmission opportunity for the i-th queue. We will consider such an instant $t_{k,i}$ as the sampling instant in our system, i.e., $\Delta t_{k,i} = t_{k+1,i} - t_{k,i}$ is the sampling interval (see Fig. 1).

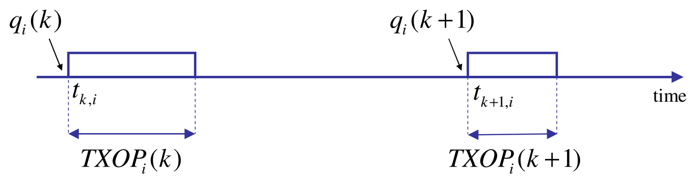


Fig. 1. Transmission opportunities of the i-th queue.

Now, the following equation holds:

$$q_i(k+1) - q_i(k) = d_i(k) - u_i(k) \tag{1}$$

where

- $q_i(k)$ is the i-th queue length at time $t_{k,i}$;
- $q_i(k+1)$ is the i-th queue length at time $t_{k+1,i}$;
- $u_i(k)$ corresponds to the amount of data that is transmitted during the $TXOP_i(k)$;
- $d_i(k)$ is the amount of data that filled the queue during the time interval $[t_{k,i}, t_{k+1,i}]$, i.e., it models the behavior of the data source feeding the i-th queue.

Naturally, $TXOP_i(k)$ can be expressed as a function of $u_i(k)$ and vice-versa, by taking into account protocol overhead, transmission rates, and packet size.

In the following, we will refer to $Q_i(z)$, $D_i(z)$, and $U_i(z)$ as the $\mathcal{Z}$-transforms of the signals $q_i(k)$, $d_i(k)$, and $u_i(k)$, respectively.

B. The control law

The E^{2}DCA scheme is based on a control law that has to compute, at each time instant $t_{k,i}$, the $TXOP_i(k)$ (i.e., the interval duration which can be used by the i-th queue for transmission), or, equivalently, $u_i(k)$. Such a control law should be properly designed in order to provide bounded delays to transmitted packets, assuring, at the same time, BIBO stability [35] to the system defined by eq. (1).

We will assume the following general control law:

$$u_i(k) = h_i(k) * q_i(k) \tag{2}$$

where the '$*$' operator represents the discrete time convolution [35].

Eq. (2) means that the amount of data to be transmitted during the k-th transmission opportunity is obtained by filtering the signal $q_i(k)$ (i.e., the queue level) through a time-invariant linear filter with pulse response $h_i(k)$ or, equivalently, with transfer function $H_i(z) = \mathcal{Z}[h_i(k)]$ [35].

Combining eqs. (1) and (2), we obtain that our scheduling algorithm realizes the control loop shown in Fig. 2, with the set point $q_i^T = 0$. This means that our control algorithm tries to target empty queues using a a linear regulator with transfer function $H_i(z)$. In the following, the pulse response of the system will be referred to as $h_{s_i}(k)$, so that the following equality holds:

$$q_i(k) = h_{s_i}(k) * d_i(k) \tag{3}$$

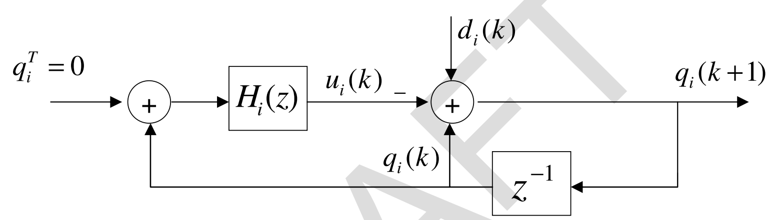


Fig. 2. Control loop of the bandwidth allocation algorithm.

Assuming $q_i(0) = 0$ (i.e., empty queues at the beginning), our design strategy is to find the proper function $H_i(z)$ that ensures the BIBO stability to the system and guaranteed queuing delays. As we will shown below, both of these objectives could be achieved if the closed-loop response to the Kronecker pulse $\delta(k)$ [35] (i.e., the system pulse response) has the following expression:

$$h_{s_i}(k) = \sum_{n=0}^{M_i} c_i(n)\delta(k-n) \tag{4}$$

where $c_i(n)$ are real finite coefficients, i.e., $c_i(n) \in R$,

First of all, if eq. (4) holds, it simple to demonstrate that the system is BIBO stable because [35]:

$$\sum_{k=0}^{+\infty} |h_{s_i}(k)| = \sum_{k=0}^{M_i} |c_i(k)| < +\infty. \tag{5}$$

To clearly explain the system behavior in response to a pulse $d_i(k) = \delta(k)$, the signals $d_i(k)$, $u_i(k)$, and $q_i(k)$ are shown in Fig. 3.

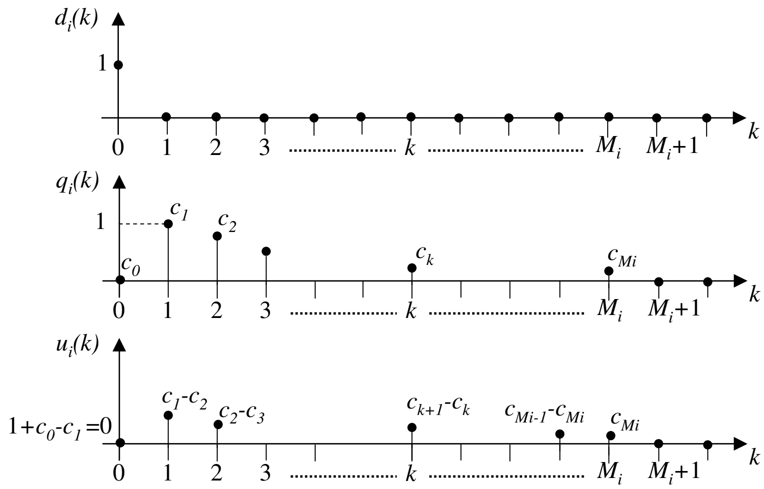


Fig. 3. System response to a pulse of data.

Remark 1: If we consider a Kronecker pulse as input to the queue, obviously the queue response given by eq. (4) cannot be negative. Therefore, it holds that $h_{s_i}(k) \geq 0 \Leftrightarrow c_i(n) \geq 0$. Moreover, the queue cannot contain more data than its input (i.e., a pulse with width equal to 1). It means that $h_{s_i}(k) \leq 1 \Leftrightarrow c_i(n) \leq 1$.

Remark 2: In order to guarantee the system causality, we have to set $c_i(0) = 0$ and $c_i(1) = 1$. In fact, a pulse of data arriving during the first sampling interval $[t_{0,i}, t_{1,i}]$ will be enqueued during that interval. It will be transmitted not before the second sampling interval $[t_{1,i}, t_{2,i}]$. In other words, assuming at time $t = 0$ an empty queue, i.e., $q_i(0) = 0$, and a single data pulse as system input, i.e., $d_i(k) = \delta(k)$, we have to impose that the queue is filled only by the data pulse at time $t = 1$, i.e., $q_i(1) = 1$. This means, equivalently, that it should be $c_i(0) = 0$ and $c_i(1) = 1$ in eq. (4).

Remark 3: With reference to eq. (1), even considering the Kronecker pulse as system input and that $c_i(0) = 0$, we have:

$$u_i(k) = d_i(k) + q_i(k) - q_i(k+1) = \delta(k) + \sum_{n=0}^{M_i} c_i(n)\delta(k-n) - \sum_{n=1}^{M_i} c_i(n)\delta(k+1-n). \quad (6)$$

After a bit of algebra, we have:

$$u_i(k) = c_i(M_i)\delta(k - M_i) + \sum_{n=1}^{M_i-1} [c_i(n) - c_i(n+1)]\delta(k-n). \quad (7)$$

Now, considering that $u_i(k)$ should not be negative, it holds that $c_i(n) \geq c_i(n+1)$ for $n \geq 1$.

To summarize, considering Remarks 1 and 3, we have to impose in eq. (4) the constraints

$$0 \leq c_i(n) \leq 1 \ \forall n; \qquad c_i(n) \geq c_i(n+1), n \geq 1. \tag{8}$$

It will be mathematically demonstrated later that these constraints are able to provide upper bounded queuing delays.

Proposition 1: The eq. (4) for the system pulse response is satisfied when the transfer function of the controller is:

$$H_i(z) = \frac{U_i(z)}{Q_i(z)} = \frac{(1-z)\sum_{n=0}^{M_i} c_i(n) z^{-n} + 1}{\sum_{n=0}^{M_i} c_i(n) z^{-n}} \tag{9}$$

Proof: By definition, the system transfer function $H_{S_i}(z)$ is just the $\mathcal{Z}$-transform of the system pulse response $h_{S_i}(k)$, assuming $q_i(0) = 0$. That is, with reference to Fig. 2:

$$H_{S_i}(z) = \frac{Q_i(z)}{D_i(z)} = \frac{1}{z - 1 + H_i(z)} = \mathcal{Z}[h_{S_i}(k)]. \tag{10}$$

Now, considering eq. (4), we have:

$$\mathcal{Z}[h_{S_i}(k)] = \mathcal{Z}\left[\sum_{n=0}^{M_i} c_i(n)\delta(k-n)\right] = \sum_{n=0}^{M_i} c_i(n) z^{-n}. \tag{11}$$

Solving eq. (10) with respect to $H_i(z)$ the proof is obtained. □

Theorem 1: The queuing delay τ_i of the i-th queue is smaller than $M_i + 1$ sampling intervals.

Proof: The thesis requires that the queue backlog measured in $t_{k+1,i}$ will be transmitted in at most $M_i + 1$ sampling interval. In this way, a generic packet that entered the queue during the time interval $[t_{k,i}, t_{k+1,i}]$ will wait in queue for at most $M_i + 1$ sampling intervals. This can be expressed as:

$$\sum_{n=0}^{M_i} u_i(k+n) \geq q_i(k) \qquad \forall k \geq 0 \tag{12}$$

Now, by considering eq. (1), $u_i(k) = q_i(k) - q_i(k+1) + d_i(k)$; then, eq. (12) can be equivalently rewritten as:

$$\sum_{n=0}^{M_i} d_i(k+n) \geq q_i(k+M_i+1) \qquad \forall k \geq 0 \tag{13}$$

Transforming back to time domain eq. (10), we obtain that:

$$q_i(k) = \sum_{n=0}^{M_i} c_i(n) d_i(k-n) \tag{14}$$

Substituting eq. (14) in (13), we obtain that (13) is equivalent to the following inequality:

$$\sum_{n=0}^{M_i} d_i(k+n) \geq \sum_{n=0}^{M_i} c_i(n) d_i(k+M_i-n+1) \tag{15}$$

Imposing $m = M_i - n + 1$, it becomes:

$$d_i(k) + \sum_{n=1}^{M_i} d_i(k+n) \geq \sum_{m=1}^{M_i} c_i(M_i - m + 1) d_i(k+m) \tag{16}$$

that is

$$d_i(k) + \sum_{n=1}^{M_i} [1 - c_i(M_i - n + 1)] d_i(k+n) \geq 0 \tag{17}$$

Remembering that $d_i(k) \geq 0$ and $0 \leq c_i(n) \leq 1$, the last inequality (17) holds for all k values. This proves the thesis. $\square$

Remark 4: Depending on the arbitration function for channel access, it is possible to find an upper bound T_i^M for the sum of $M_i + 1$ sampling intervals. Thus, the proposed allocation algorithm assures that $\tau_i \leq T_i^M$.

Theorem 2: The average queuing delay τ_{avg} of the i-th queue is equal to $\sum_{n=0}^{M_i} c_i(n)$.

Proof: In the general case of a VBR data source, using the Little law [36], the average delay is given by

$$\tau_{avg} = E[q_i]/E[d_i] \tag{18}$$

that is, the ratio between the expectation of the two processes q_i and d_i.

Assuming the processes as ergodic, the two expectation are equivalent to the time averages. Hence, we have:

$$\tau_{avg} = \frac{\lim\limits_{N\to\infty} \dfrac{1}{N} \sum\limits_{k=0}^{N} q_i(k)}{\lim\limits_{N\to\infty} \dfrac{1}{N} \sum\limits_{k=0}^{N} d_i(k)} = \frac{Q_i(z)|_{z=1}}{D_i(z)|_{z=1}} \tag{19}$$

Now, by considering eq. (11), eq. (19) can be equivalently rewritten as:

$$\tau_{avg} = \frac{Q_i(z)|_{z=1}}{D_i(z)|_{z=1}} = \sum_{n=0}^{M_i} c_i(n) \tag{20}$$

This proves the thesis. □

Remark 5: With reference to eq.(4) , we can set the $c_i(n)$ coefficients as follows:

$$\begin{cases} c_i(0) = 0 \\ c_i(n) = 1 - \frac{n-1}{M_i} \end{cases} \tag{21}$$

for $n = 1, \ldots, M_i$. In this way $H_{S_i}(z)$, described in eq. (11), has a linear pulse shaping.

With this choice for the coefficients, considering eq.(21), the eq.(20) can be rewritten as:

$$\tau_{avg} = \sum_{n=0}^{M_i} c_i(n) = \sum_{n=1}^{M_i} (1 - \frac{n-1}{M_i}) = \sum_{n=1}^{M_i} (\frac{M-n+1}{M_i}) \tag{22}$$

Imposing $m = M - n + 1$, it becomes:

$$\tau_{avg} = \sum_{m=1}^{M_i} (\frac{m}{M_i}) = \frac{M_i(M_i+1)}{2M_i} = \frac{M_i+1}{2}. \tag{23}$$

To summarize the results, if we consider the E²DCA scheme using a controller with the transfer function (9) and the constraint (8), we can ensure BIBO stability to the system and guaranteed delays. A proper choice of the coefficients of the transfer function allow us to estimate an upper bound and the average value for the delay.

C. TXOP assignemnts

After computing the bandwidth, E²DCA defines how each virtual station has to transform $|u_i(n)|$ into a $TXOP_i(k)$ assignment, according to the 802.11e standard.

If the packets in the i^{th} queue are transmitted at rate C_i, the following relation is considered:

$$TXOP_i(k) = \frac{u_i(k)}{C_i} + H(k) \tag{24}$$

where $TXOP_i(k)$ is the TXOP assigned to the i^{th} queue during the k^{th} successful contention, and $H(k)$ is the time overhead due to ACK packets and interframe spaces (see the 802.11e standards [6] for details about these interframe intervals).

D. Call Admission Control

In order to achieve the performance expected from Theorem 1 and Theorem 2, it is necessary to avoid bandwidth saturation episodes. For this purpose, we have developed an innovative CAC test to be used jointly with E^{2}DCA. The CAC test should allow all admitted flows to win at least one contention in each superframe and, at the same time, ensure that the sum of the allocated TXOPs is smaller than the superframe duration, T_{SF}.

We suppose that the system has already admitted N active flows when a new admission request is received, i.e., the $(N+1)$-th flow. To design the CAC algorithm we consider the k-th transmission opportunity to be assigned to each active flow by using eq. (24).

Now, we can model each data source using token bucket parameters [37]. In particular, for the i-th data source, we have to consider the peak data rate p_i, the average rate ρ_i, and the burst size B_i. Note that we have not any constraint on the peak rate, e.g, $p_i = \infty$.

As shown before in Theorem 1, each pulse of data $d_i(k)$ has a transient pulse response with a duration smaller than $M_i + 1$ consecutive sampling intervals. Moreover, as stated in Remark 4, this duration has an upper bound equal to T_i^M depending on the arbitration function, i.e., $T_i^M \geq \sum_{i=0}^{M_i} \Delta t_{k,i}$.

To solve the admission test for each k, we have to find an upper bound u_{max_i} for $u_i(k)$. Such a bound can be estimated considering the worst bursty behavior of the i-th source, i.e., when one single pulse of data with the maximum width is sent regularly each $M_i + 1$ sampling intervals. Considering the token bucket modeling, the width of this pulse is:

$$B_i^{max} = \min\{B_i, \rho_i T_i^M\} \tag{25}$$

that is, it is the minimum between the burst size B_i and the amount of data that can be generated at the average rate ρ_i in the interval T_i^M.

Under these hypothesis, we can compute the maximum value of bandwidth u_{max_i} allocated for the i-th queue. In fact, by eq. (4) if a pulse $d_i(k) = B_i^{max}\delta(k)$ arrives to the i-th queue, the pulse response is $q_i(k) = B_i^{max} \sum_{n=1}^{M_i} c_i(n)\delta(k-n)$. Now, using eq. (1), we can easily obtain:

$$u_{max_i} = B_i^{max}\Delta c_i, \qquad \text{with } \Delta c_i = \max_{n=1,\ldots,M_i}\{c_i(n-1) - c_i(n)\}. \tag{26}$$

Now, we can define our CAC test, that is a new flow is admitted only if the following inequality is satisfied:

$$\sum_{i=1}^{N+1}\left(\frac{B_i^{max}\Delta c_i}{C_i}+H\right)<T_{SF}. \tag{27}$$

In the next section we will compare E^{2}DCA with respect to the EDCA scheme, which allows a single frame to be transmitted for each successful contention. We also compare our algorithm with a Multiple Frame Transmission (MFT) scheme which allows a station to transmit all the frames in its transmission queue after a successful channel contention (obviously still compliant with limits on TXOP duration). So that, herein we need to properly define the CAC behavior for these two other schemes.

When we use the MFT scheduler, for each flow the maximum TXOP duration is set to accommodate the flow burst size. In other words, the CAC scheme used with MFT should ensure that all admitted flows can send at least one burst of data, one time in a superframe. Therefore, the CAC test becomes:

$$\sum_{i=1}^{N+1}\left(\frac{B_i}{C_i}+H\right)<T_{SF}. \tag{28}$$

It is worth to note that TXOPs allocated by E^{2}DCA are smoother than those of MFT due to the low-pass filter $H_i(z)$, which is not exploited by MFT. Thus, we expect a larger number of flow admitted by E^{2}DCA with respect to MFT.

Finally, when the simple EDCA is used, no CAC test is applied as in the standard. Thus, we expect that EDCA admits almost all flows, but it should be unable to provide any QoS guarantees.

This considerations will be proved in the next section using computer simulations.

III. Performance Evaluation

In order to assert the validity of E^{2}DCA in realistic scenarios, computer simulations involving voice, video, and FTP data transfers have been run using the *ns*-2 simulator [34]. We have considered a scenario compliant to 802.11g [10] standard (see Fig. 4) where a number of 3α G.729 voice flows [38], α MPEG-4 encoded video flows [39], α H.263 video flows [40], and α FTP flows are transmitted at a data rate of 54 Mbps (see Fig. 4). Therefore, in such a scenario the traffic load is proportional to the parameter α (i.e., the load factor). Each wireless node hosts a single multimedia or FTP sender. The HC is connected through a wired link with a sink node, hosting receivers. This link has capacity equal to 100 Mbps, propagation delay equal to 20 ms, and a maximum queue size equal to 50 packets.

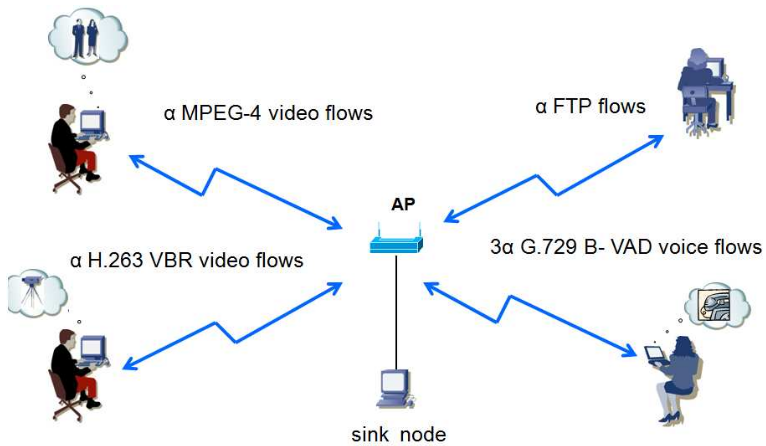


Fig. 4. Scenario with multimedia flows.

G.729 sources have been modeled as Markov ON/OFF sources [41]. The ON period is exponentially distributed with mean 3 s; the OFF period has a truncated exponential pdf with an average value of 2.23 s and an upper limit of 6.9 s [42]. During the ON period, the voice source sends packets of 20 bytes every 20 ms (i.e., the source data rate is 8 kbps; also, we are considering two G.729 frames combined into one packet [43]). By taking into account the overheads of the RTP/UDP/IP protocol stack, the total rate at network layer is 24 kbps, during the ON periods. During the OFF period the rate is set to zero since we assume the presence of a Voice Activity Detector (VAD). For what concerns video flows, we have used traffic traces available from the video trace library [44]. The duration of video flows is deterministic and equal to 200 s, whereas voice flows durations are exponentially distributed with an average value of 120 s. When a multimedia flow terminates, a new stream of the same type is generated after an exponentially distributed random time, with an average value equal to 1 minute. Each simulation lasts 800 s and all simulation results are averaged over 5 simulations.

The main characteristics of the considered multimedia flows are summarized in Table II.

Before data transmission, a multimedia source has to set up a new Traffic Stream. If the reply to the stream admission message is not received within a σ_{TO} timeout interval, the request is repeated up to a maximum number N_{Adm} of times; as in [45], we have chosen $N_{Adm} = 10$

TABLE II

MAIN CHARACTERISTICS OF THE CONSIDERED MULTIMEDIA FLOWS.

Type of flow	Nominal MSDU [byte]	Maximum MSDU [byte]	Burst Size [byte]	Mean Data Rate [kbps]	Peak Data Rate [kbps]
MPEG-4 HQ	1536	2304	16745	770	3300
H.263 VBR	1536	2304	18168	450	3400
G.729 VAD	60	60	60	8.4	24

and $\sigma_{TO} = 1.5$ s. If after N_{Adm} admission tries no reply is received back, then the request is considered lost and a new admission procedure is initiated after an exponential distributed random time σ_{defer} with average value equal to 1 minute. When a flow terminates, it is withdrawn from the polling list by the HC when no more packets of that flow are received for a time equal to the Inactivity Interval, which is equal to 3 s for video flows and 10 s for voice flows.

When E^{2}DCA is used, we have considered $M_i = 3$ or $M_i = 5$. In particular we have set coefficient $c_i(n)$ by using eq. (21).

Contention parameters have been set as suggested in Tab. I for all the arbitration shemes. FTP flows are served using standard EDCA, but with the smallest priority.

According to the 802.11 standard, in our *ns*-2 implementation the T_{SF} is expressed in Time Units (TUs), each one equal to 1024 μs [1]. We have studied scheduler performances with T_{SF} equal to 29 TUs.

We will first assume an ideal wireless channel and we will analyze the impact of the network load on the performance of E^{2}DCA, EDCA, and MFT. Then, we will investigate the effect of noise on the channel, using a frame-level error model [46]. Finally, we will study the interoperability between E^{2}DCA, EDCA, and MFT in a hybrid scenario where the three algorithms are used simultaneously.

A. The impact of the load factor

In order to investigate the impact of the network load on E^{2}DCA, EDCA, and MFT performances, we analyzed the network behavior when the load factor α is varied in the range $[4 \div 20]$.

Fig. 5 shows that E^{2}DCA admits almost all flows as well as EDCA. On the contrary, MFT admits a very limited number of flows.

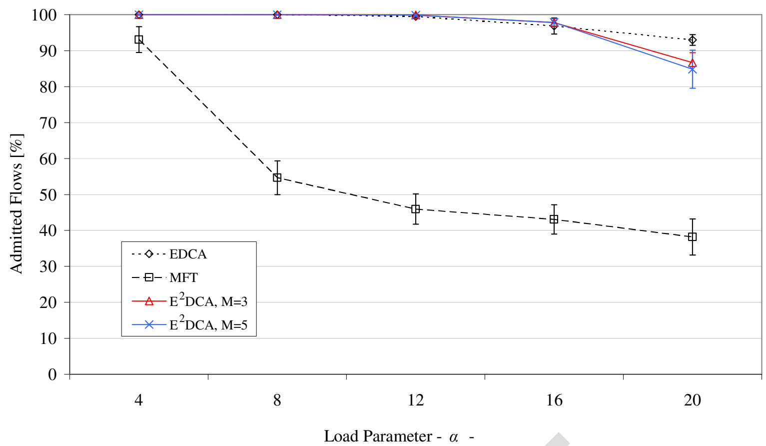


Fig. 5. Percentage of admitted flows.

Fig. 6 shows the average and the 95% percentile of packet delays of multimedia flows. It is clear that E^{2}DCA is able to reduce the packet delay with respect to EDCA under high network load. On the other hand MFT achieves the smallest delays because it admits the smallest quota of flows (see fig. 5). It is worth to note that E^{2}DCA with $M_i = 5$ ($M_i = 3$), also admitting almost all flows, is able to provide, in the scenario with the highest load factor, an upper bound on packet delays equal to 130 ms (100 ms) for video flows and 16 ms (14 ms) for voice flows.

It is important to note that the performance of E^{2}DCA depends on the M_i value. This is another privilege of E^{2}DCA scheduler because changing M_i value we can obtain different priority levels, enforcing service differentiation.

We can observe that G.729 flows exhibit the smallest packet delays with respect to other multimedia flows. The reason is that the voice flows have the highest priority and gain the right to transmit with the highest probability (see Tab. I).

To analyze how the real-time service provided to multimedia flows affects the performance of FTP flows, Fig. 7 reports the goodput achieved by this kind of flows. Goodput of FTP flows is strictly related to the the quota of admitted flows. In fact MFT scheduler achieves the highest goodput with respect to other algorithms. E^{2}DCA and EDCA achieve similar FTP goodputs because they admit similar quota of flows. The reason is that FTP uses at transport layer the TCP protocol, which is able to fully exploit the bandwidth left available by real-time flows.

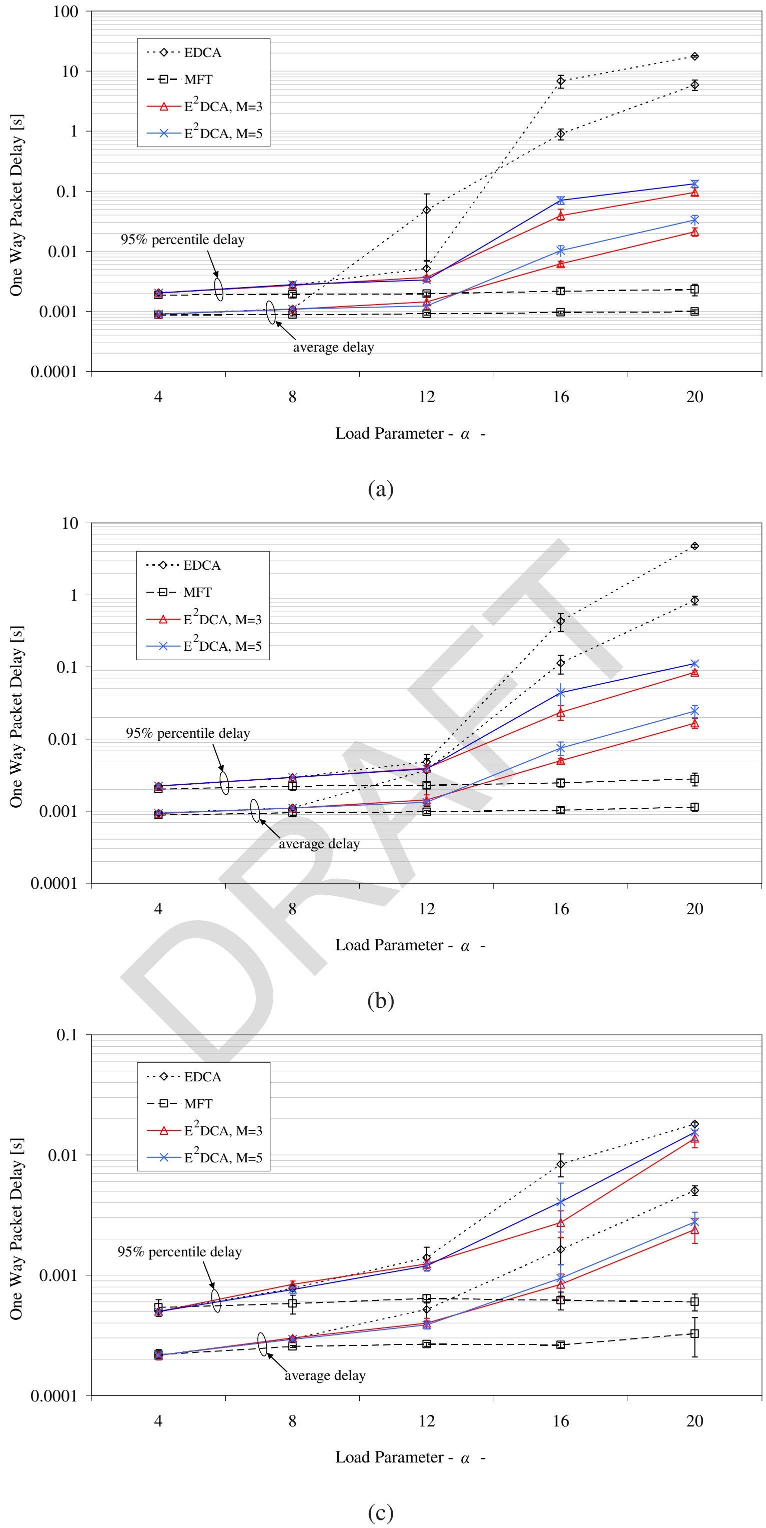


Fig. 6. Average and 95% percentile delays of (a) MPEG4, (b) H263, and (C) G.729 flows.

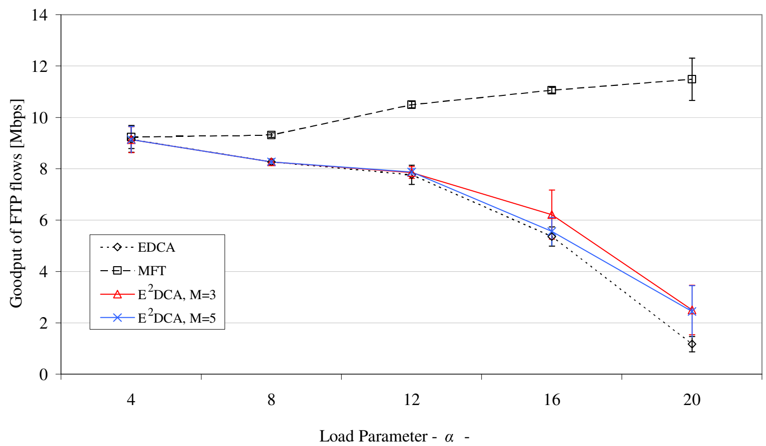


Fig. 7. Goodput of FTP flows.

To provide a further insight, Fig. 8 shows the ratio of collided frames, computed as the ratio between the number of collisions and the total number of transmitted frames at MAC level. Naturally, the ratio of collided frames increases as the load factor α increases. We note that E^{2}DCA provides a collision ratio smaller than EDCA. On the other hand, MFT achieves the lowest ratio of collided frames because it admits the smallest quota of flows.

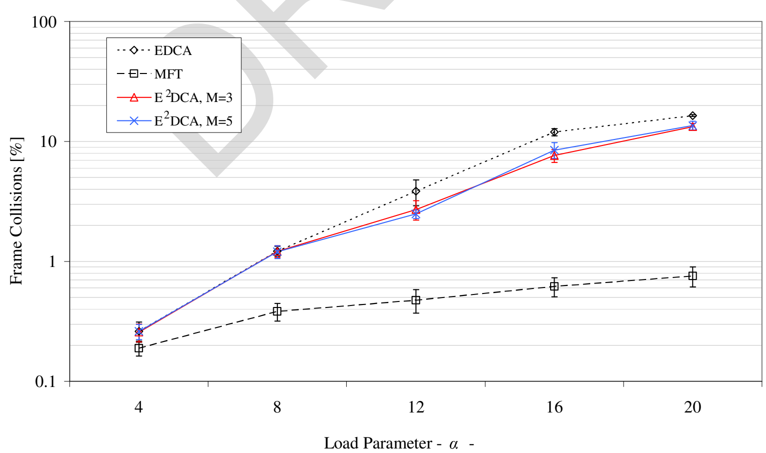


Fig. 8. Percentage of collided frames.

B. The impact of noise

All previous results have been obtained considering ideal channel conditions, i.e., a frame error rate (FER) equal to zero. To verify the effectiveness of E^{2}DCA in a realistic environment, also the impact of noise on the channel has been studied. We have modeled the loss process affecting the WLAN channel with a semi-Markov model, referred to as k-state threshold model [46]. This model describe very well the behavior of a real wireless channel at the data link layer.

When a MAC frame is lost due to interference or channel unreliability, it is retransmitted up to a maximum number of times equal to 7.

From Fig. 9 we note that random packet losses do not appreciabily modify the percentage of admitted flows.

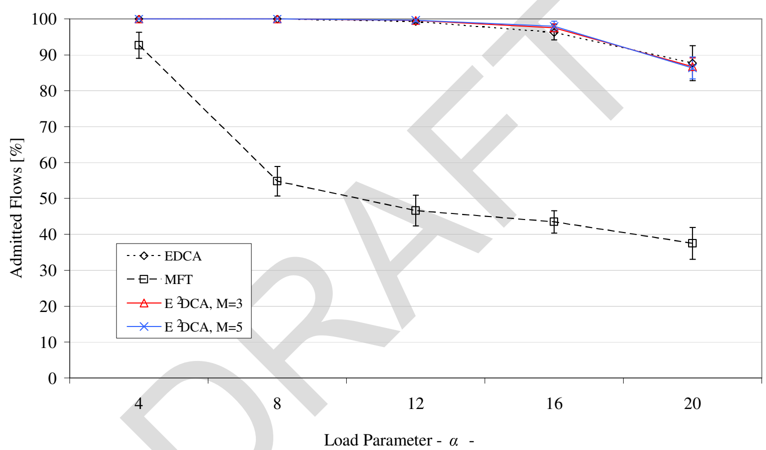
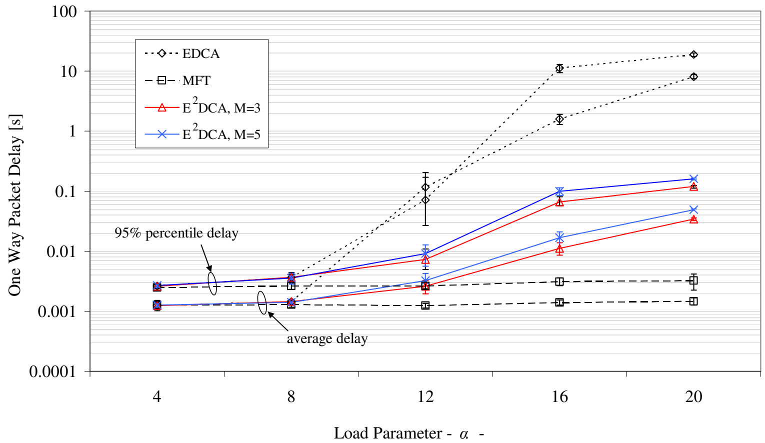


Fig. 9. Percentage of admitted flows with a noisy channel.

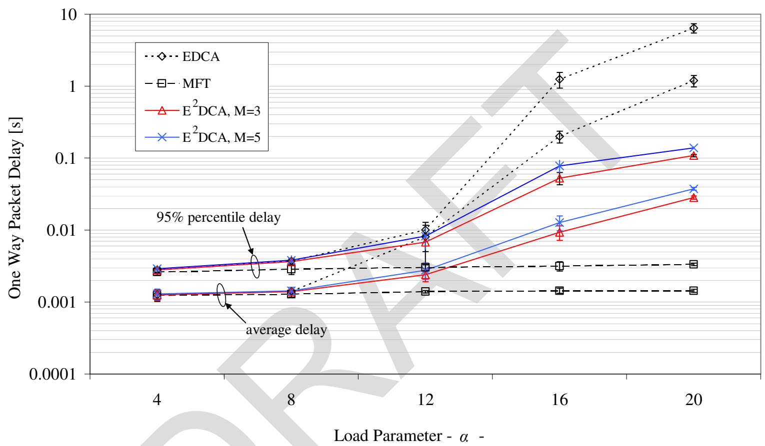
The effect of noise on packet delays is more evident with respect to the case of a loss-free environment (see Fig. 10). In fact, a quota of allocated TXOPs is now used for frame retransmission. It is worth to note that E^{2}DCA with $M_i = 5$ ($M_i = 3$), also admitting almost all flows, is able to provide, in the scenario with the highest load factor, an upper bound on packet delays equal to 160 ms (120 ms) for video flows and 20 ms (20 ms) for voice flows.

Fig. 11 shows that, with a noisy channel, the goodput of FTP flows decreases with respect to the case of a loss-free environment.

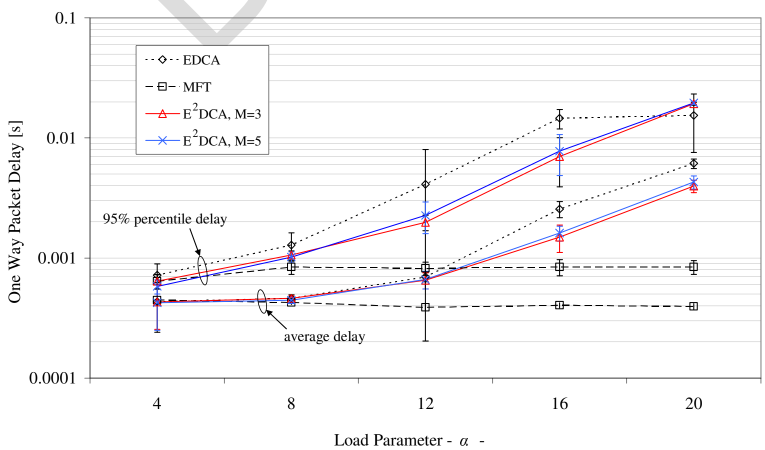
Fig. 12 shows that the ratio of collided frames increases when there is noise on the channel, due to frame retransmission.



(a)



(b)



(c)

Fig. 10. Average and 95% percentile delays of (a) MPEG4, (b) H263, and (C) G729 flows with noise on the channel.

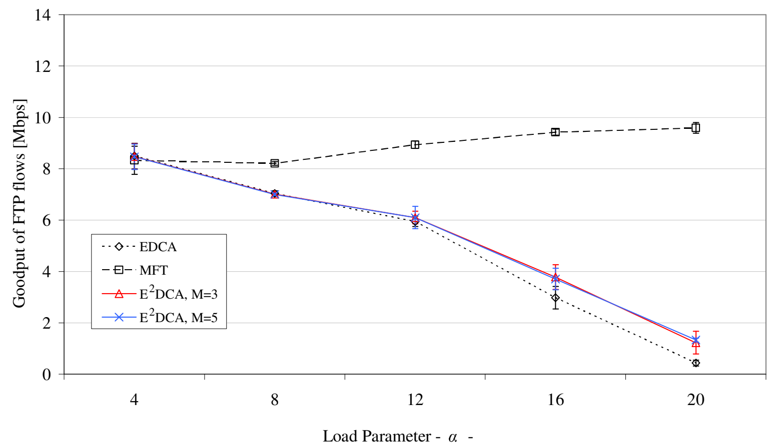


Fig. 11. Goodput of FTP flows with noise on the channel.

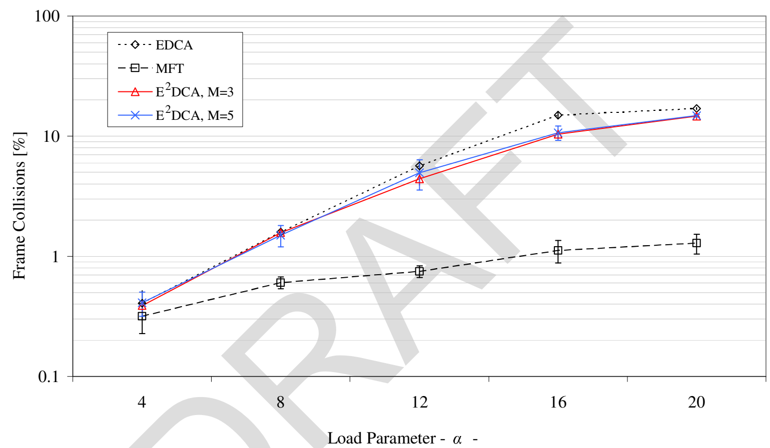


Fig. 12. Percentage of collided frames with noise on the channel.

C. Interoperability between EDCA, MFT, and E^{2}DCA

In a real life scenario we can find stations that operate with different bandwidth algorithms. Therefore it is very important to study the interoperability between E^{2}DCA, EDCA, and MFT. We have considered a scenario similar to the one described in the Fig. 4, but where there are:

- $\alpha/4$ MPEG-4 flows, $\alpha/4$ H.263 flows and $3\alpha/4$ voice flows that use E^{2}DCA scheme,
- $\alpha/4$ MPEG-4 flows, $\alpha/4$ H.263 flows and $3\alpha/4$ voice flows that use EDCA scheme,
- $\alpha/4$ MPEG-4 flows, $\alpha/4$ H.263 flows and $3\alpha/4$ voice flows that use MFT schemer.

We assigned to α values that are multiple of 4 in the range $[4 \div 20]$. As in the previous section, we have analyzed the performance of the considered algorithms both of an ideal and a real wireless channel. To ensure interoperability, a more general CAC scheme has been used.

At the end of k-th superframe, the AP measures the quota of T_{SF} used by each scheduler in the current superframe, referred as follows:

- $T_{E^2DCA}(k)$ is the quota of T_{SF} used by E^{2}DCA scheme;
- $T_{MFT}(k)$ is the quota of T_{SF} used by MFT scheme;
- $T_{EDCA}(k)$ is the quota of T_{SF} used by EDCA scheme.

Let $T_{sched}(k)$ be the the quota of T_{SF} used by the generic scheduler. The AP estimates its weighted moving average value, referred as $\hat{T}_{sched}(k)$, using the following equation:

$$\hat{T}_{sched}(k) = 0.9\hat{T}_{sched}(k-1) + 0.1T_{sched}(k). \tag{29}$$

The CAC test is implemented if a new admission request is received by the AP during the $(k+1)$-th superframe.

Let s and r be the number of active flows that use E^{2}DCA scheme and the number of active flows that use MFT scheme, respectively.

For the E^{2}DCA scheduler, the CAC test is given by:

$$\sum_{i=1}^{s+1} \left(\frac{B_i^{max} \Delta c_i}{C_i} + H \right) < T_{SF} - \left(\hat{T}_{EDCA}(k) + \hat{T}_{MFT}(k) \right). \tag{30}$$

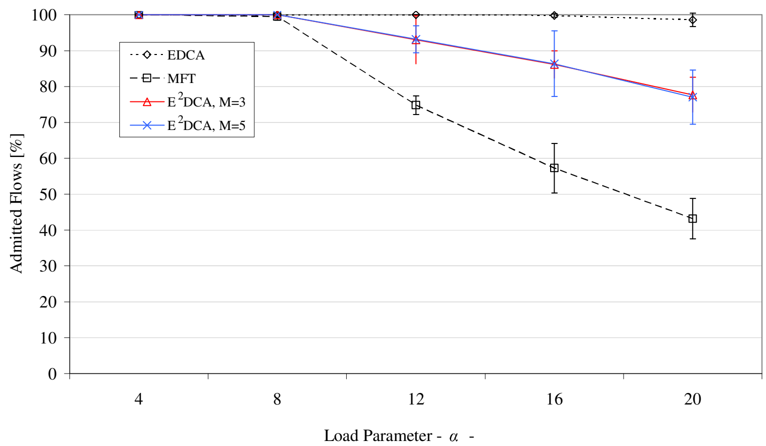
Whereas, for the MFT scheduler the CAC test is given by:

$$\sum_{i=1}^{r+1} \left(\frac{B_i}{C_i} + H \right) < T_{SF} - \left(\hat{T}_{EDCA}(k) + \hat{T}_{E^2DCA}(k) \right). \tag{31}$$

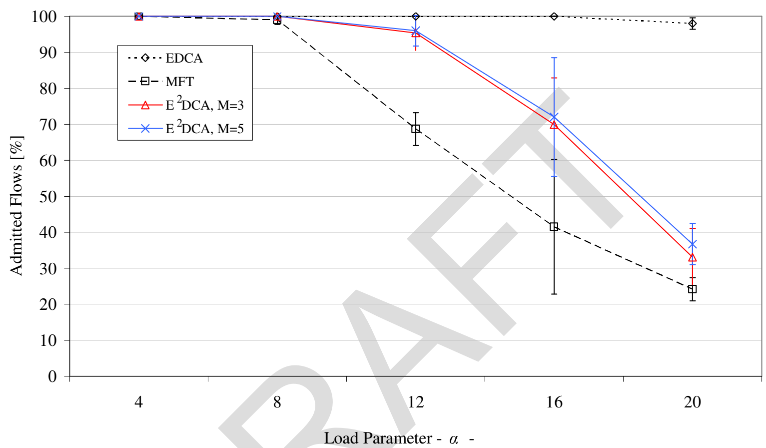
When a new flow arrives, it is admitted only if one of the last equations is satisfied.

When EDCA is used, all received admission requests are accepted.

Figs. 13 show the percentage of admitted flows. We note that as the load factor increases the percentages of flows admitted by E^{2}DCA and by MFT decrease. Nevertheless, E^{2}DCA always admits more flows than MFT as in the homogeneous scenario. Obviously, EDCA exhibits a ratio of admitted flows very closed to 100% because its CAC admits all requests. It is worth to note that E^{2}DCA admits the smallest flows with respect to the homogeneous scenario because in this case a portion of bandwidth is used by other scheme. It is important to note that, when frame



(a)



(b)

Fig. 13. Percentage of admitted flows in a hybrid scenario (a) without noise, (b) with noise.

error model has been used, E^{2}DCA and MFT admit a smaller number of flows with respect to the homogeneous scenario.

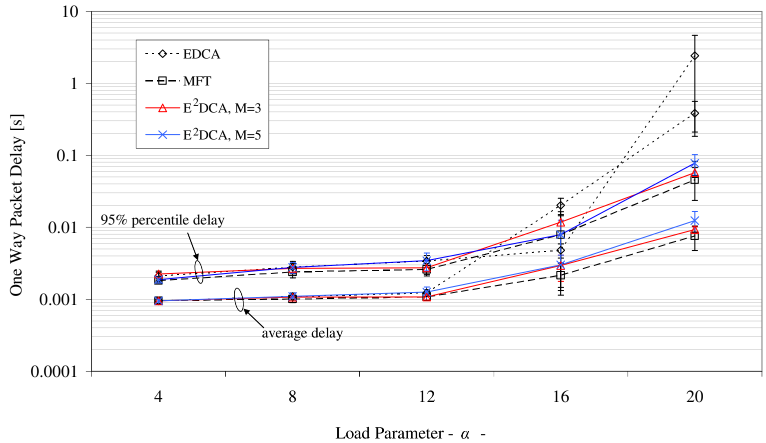
Figs. 14 and 15 show the average and the 95% percentile of packet delays of multimedia flows. The main result is that in the heterogeneous scenario E^{2}DCA and MFT achieve similar performance in terms of packet delays. EDCA, instead, obtains highest delays of video flows with respect to other schedulers when load factor increases. This effect is more evident when frame error model has been used.

It is worth to note that E^{2}DCA with $M_i = 5$ ($M_i = 3$), also admitting almost all flows, is able to provide, in the scenario with the highest load factor, an upper bound on packet delays equal to 80 ms (60 ms) for video flows and 7 ms (7 ms) for voice flows. Similarly, when a noisy channel has been considered, we note that E^{2}DCA with $M_i = 5$ ($M_i = 3$) is able to provide, in the scenario with the highest load factor, an upper bound on packet delays equal to 140ms (100 ms) for video flows and 16 ms (16 ms) for voice flows.

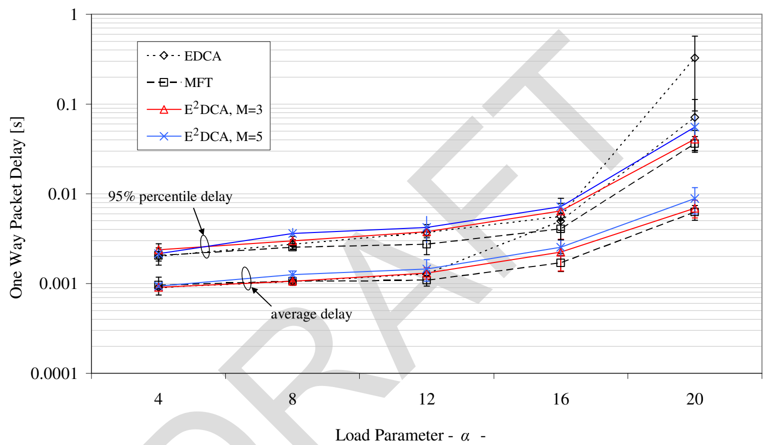
We can observe that G.729 flows exhibit smaller packet delays, independently on the scheduler, with respect to other multimedia flows, because voice flows have a higher priority than video flows. So that, voice flows gain the right to transmit with a higher probability.

Figs. 16-17 show that, with a noisy channel, the goodput of FTP flows decreases and frame collisions increase.

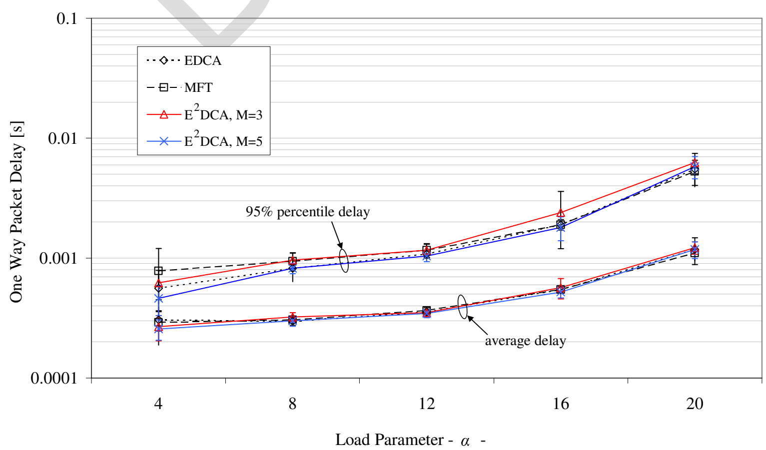
Finally, we emphasize that E^{2}DCA, MFT, and EDCA are compatible in the same WLAN scenario. On the other hand, in a heterogeneous scenario E^{2}DCA achieves the best trade-off between percentage of admitted flows and packet delays with respect to the other bandwidth algorithms.



(a)

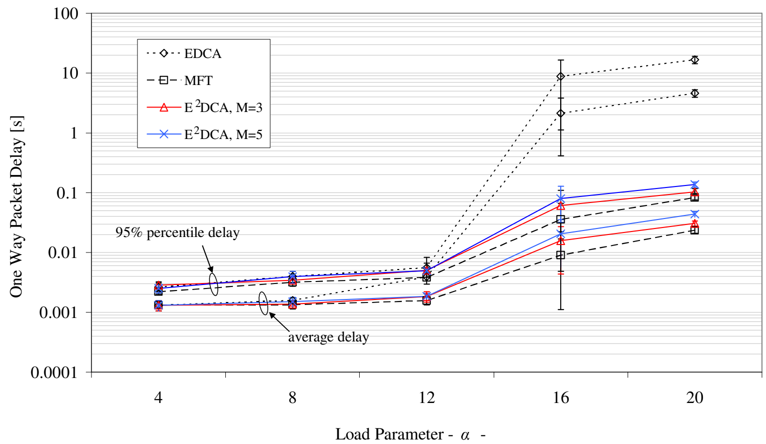


(b)

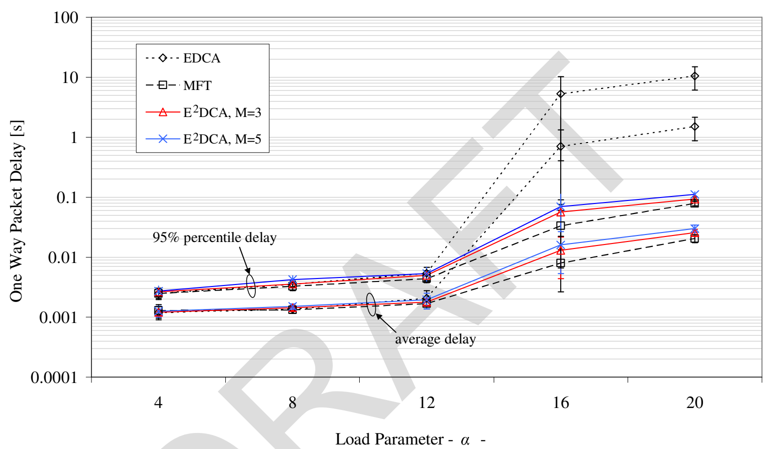


(c)

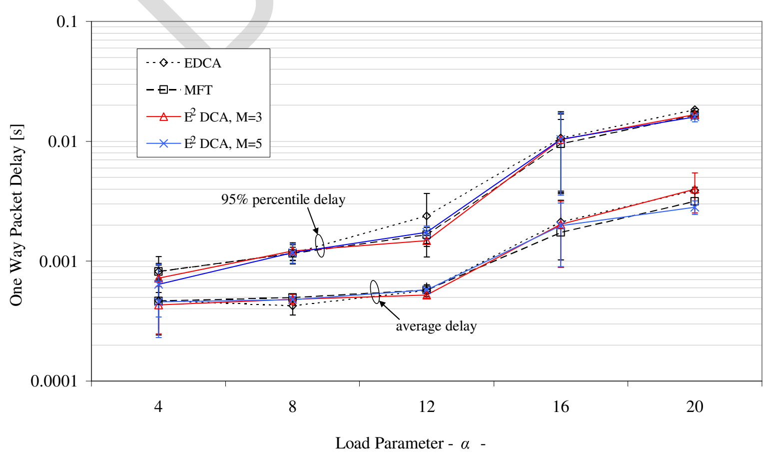
Fig. 14. Average and 95% percentile delays of (a) MPEG4, (b) H263, and (C) G729 flows in a hybrid scenario.



(a)



(b)



(c)

Fig. 15. Average and 95% percentile delays of (a) MPEG4, (b) H263, and (C) G729 flows in a hybrid scenario with frame-level error model.

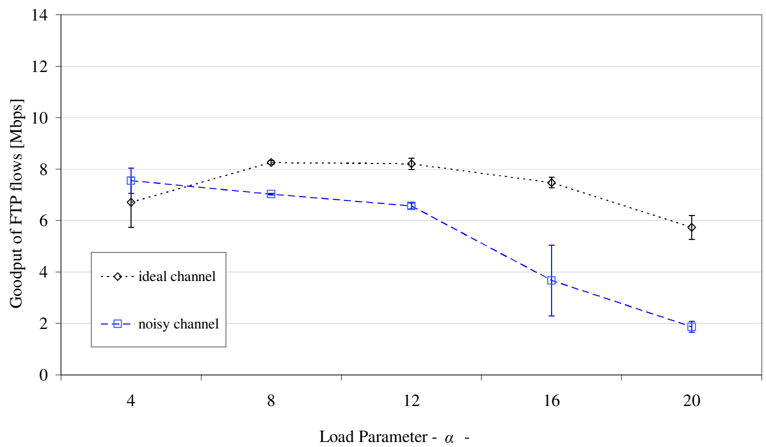


Fig. 16. Goodput of FTP flows in a hybrid scenario.

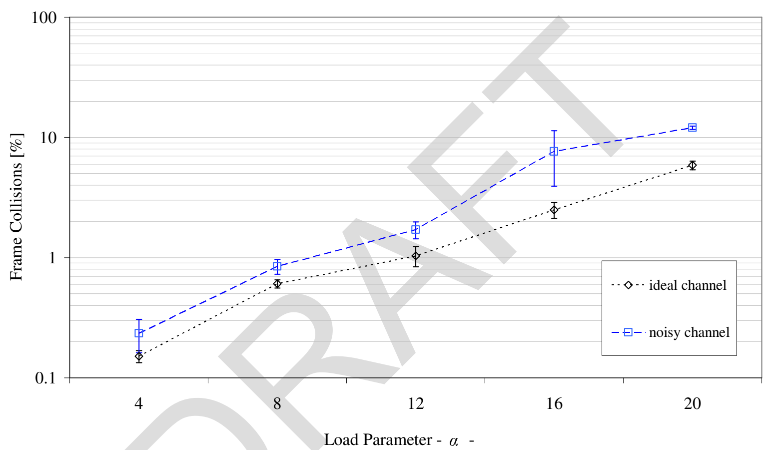


Fig. 17. Percentage of collided frames in a hybrid scenario.

IV. CONCLUSION

The new E^{2}DCA scheme has been proposed in this paper, compliant with IEEE 802.11e specifications. It allows a service differentiation in WLANs by means of a dynamic bandwidth allocation on distributed basis. The proposed scheme has been designed using closed loop control theory. Algorithm performances have been investigated, using the *ns*-2 simulator, in realistic network scenarios, involving a mix of audio, video, and FTP flows, at several network loads and with random losses. Simulation results have shown that E^{2}DCA is able to provide bounded delays to multimedia flows in a wide range of network loads and frame loss ratios. On the contrary, EDCA is not able to provide service differentiation in WLAN, especially at high network loads.

REFERENCES

[1] IEEE 802.11, *Information Technology - Telecommun. and Information Exchange between Systems. Local and Metropolitan Area Networks. Specific Requirements. Part 11: Wireless LAN MAC and PHY Specifications*, 1st ed., ANSI/IEEE Std. 802.11, ISO/IEC 8802-11, 1999.

[2] S. Mangold, S. Choi, P. May, O. Klein, G. Hiertz, and L. Stibor, "IEEE 802.11e Wireless LAN for Quality of Service," in *European Wireless Conference 2002*, Florence, Italy, Feb. 2002.

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

[4] A. Koepsel, J. P. Ebert, and A. Wolisz, "A Performance Comparison of Point and Distributed Coordination Function of an IEEE 802.11 WLAN in the Presence of Real-Time Requirements," in *Proc. of 7th Intl. Workshop on Mobile Multimedia Communications (MoMuC2000)*, Tokio,Japan, October 2000.

[5] B. H. Walke, S. Mangold, and L. Berlemann, *IEEE 802 Wireless Systems*. John Wiley and Sons, 2006.

[6] IEEE 802.11 WG, *Draft Amendment to Standard for Information Technology. LAN/MAN Specific Requirements - Part 11: Wireless MAC and PHY Specifications: MAC Quality of Service (QoS) Enhancements*, IEEE 802.11e/D13.0, Jan. 2005.

[7] IEEE 802.11, *Information Technology - Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE 802.11-REVma/D9.0, Revision of Std. 802.11-1999, 2006.

[8] ——, *Supplement to IEEE Standard for Information Technology Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 5 GHz Band*, IEEE Std 802.11a, ISO/IEC 8802-11:1999/Amd 1:2000(E), 1999.

[9] ——, *Supplement to IEEE Standard for Information Technology Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band*, IEEE Std 802.11b, 1999.

[10] ——, *Supplement to IEEE Standard for Telecommunications and Information Exchange Between Systems-LAN/MAN Specific Requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band*, IEEE Std 802.11g, 2003.

[11] E. Perahia and R. Stacey, *Next Generation Wireless LANs: Throughput, Robustness, and Reliability In 802. 11n*. Cambridge University Press, 2008.

[12] D. Gu and J. Zhang, "QoS enhancement in IEEE802.11 wireless local area networks," *IEEE Commun. Mag.*, pp. 120–124, June 2003.

[13] G. Bianchi and I. Tinniriello, "Analysis of priority mechanisms based on differentiated inter frame spacing in CSMA-CA," in *IEEE 58th Veichular Technology Conference (VTC03), fall*, Orlando, October 2003.

[14] W. Pattara-Atikom and P. Krishnamurty, "Distributed Mechanisms for Quality of Service in Wireless LANs," *IEEE Wireless Communications*, pp. 1536–1284, June 2003.

[15] A. Banchs, X. Perez-Costa, and D. Qiao, "Providing throughput guarantees in IEEE 802.11e Wireless LANs," in *Providing Quality of Service in Heterogeneous Environments, ITC 2003*, Berlin, Germany, August 2003, pp. 1001–1010.

[16] A. Lindgren, A. Almquist, and O. Schelén, "Quality of service schemes for IEEE 802.11 wireless LANs - an evaluation," *Mobile Networks and Applications*, vol. 8, no. 3, pp. 223–235, June 2003.

[17] Z. Kong, D. H. K. Tsang, B. Bensaou, and D. Gao, "Performance Analysis of IEEE 802.11e Contention-Based Channel Access," *IEEE Journal of Selected Areas in Communications*, vol. 22, no. 10, pp. 2095–2106, December 2004.

[18] G. Boggia, P. Camarda, L. A. Grieco, and S. Mascolo, "Feedback-based control for providing real-time services with the 802.11e MAC," *IEEE/ACM Trans. on Networking*, vol. 15, no. 2, pp. 323–333, Apr. 2007.

[19] Q. Ni, L. Romdhani, and T. Turletti, "A survey of QoS enhancements for IEEE 802.11 wireless LAN," *Wireless Communications and Mobile Computing*, vol. 4, pp. 547–566, 2004.

[20] L. Romdhami, Q. Ni, and T. Turletti, "Adaptive EDCF: Enhanced service differentiation for IEEE 802.11 wireless Ad-Hoc networks," in *Proc. of IEEE Wireless Commun. and Networking Conf. (WCNC)*, New Orleans, Louisiana, USA, Mar. 2003.

[21] Y. Xiao, H. Li, and S. Choi, "Protection and guarantee for voice and video traffic in IEEE 802.11e Wireless LANs," in *IEEE Infocom*, Hong Kong, March 2004.

[22] K. J. Noh, W. Y. Choi, and S. K. Lee, "Adaptive and Dynamic Tuning of the Operation Parameter Value for QoS and Fairness in Wireless LAN," in *Proceedings of IEEE VTC 2004 Fall*, Los Angeles, U.S.A., September 2004.

[23] J. Naoum-Sawaya, B. Ghaddar, S. Khawam, H. Safa, H. Artail, and Z. Dawy, "Adaptive Approach for QoS Support in IEEE 802.11 Wireless LAN," in *Proceedings of IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communications (WiMob2005)*, Montreal, Canada, August 2005.

[24] Y. W. Lan, J. Yeh, J. C. Chen, and Z. T. Chou, "Performance Enhancement of IEEE 802.11e EDCA by Contention Adaption," in *Proceedings of IEEE VTC 2005 Spring*, Stockholm, Sweden, May 2005.

[25] J. F. Lee, W. Liao, and M. C. Chen, "A per-class QoS service model in IEEE 802.11e WLANs," in *Proc. of 2nd IEEE Int. Conf. on Quality of Service in Heterogeneous Wired/Wireless Networks, QShine 05*, Orlando, Florida, Aug. 2005.

[26] T. Nilsson and J. Farooq, "A novel mac scheme for solving the qos parameter adjustment problem in ieee 802.11e edca," *International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2008)*, pp. 1 – 9, June 2008.

[27] H. Kyung, S. Seo, and S. C. Park, "QoS (Quality-of-Service) improvement in IEEE 802.11e enhanced distributed channel access (EDCA)," in *Proceedings of the IEEE International Symposium on Communications and Information Technology (ISCIT 2004)*, Sapporo, Japan, October 2004.

[28] A. Iera, A. Molinaro, G. Ruggeri, and D. Tripodi, "Dynamic prioritization of multimedia flows for improving QoS and throughput in IEEE 802.11e WLANs," in *Proceedings of the IEEE International Conference on Communications (ICC 2005)*, Seoul, Korea, May 2005.

[29] C. Yanhong, Z. Li, and B. Li, "Adaptive transmission probability for ieee 802.11e mac enhancements," *International Conference on Wireless Communications, Networking and Mobile Computing (WiCom 2007)*, pp. 2169 – 2172, Sept. 2007.

[30] S. Kim, R. Huang, and Y. Fang, "Deterministic priority channel access scheme for qos support in ieee 802.11e wireless lans," *Vehicular Technology, IEEE Transactions on*, vol. 58, pp. 855 – 864, Feb. 2009.

[31] D. Qianhua and C. Anni, "A txop-based scheduling algorithm for video transmission in ieee 802.11e networks," *6th International Conference on ITS Telecommunications Proceedings*, pp. 573 – 576, June 2006.

[32] N. Cranley and M. Davis, "An experimental investigation of ieee 802.11e txop facility for real-time video streaming," *IEEE Global Telecommunications Conference (GLOBECOM '07)*, pp. 2075 – 2080, Nov 2007.

[33] M. Geyong, H. Jia, and M. Woodward, "A dynamic ieee 802.11e txop scheme in wlans under self-similar traffic: Performance enhancement and analysis," *IEEE International Conference on Communications (ICC '08)*, pp. 2632 – 2636, May 2008.

[34] Ns-2, "Network simulator," http://www.isi.edu/nsnam/ns, 2009.

[35] K. J. Astrom and B. Wittenmark, *Computer controlled systems: theory and design*, 3rd ed. Prentice Hall, 1995.

[36] L. Kleinrock, *Queueing systems, Volume I: Theory*. Wiley & Sons, 1975.

[37] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach (4th Edition)*. Addison-Wesley Longman Publishing Co.,, 2007.

[38] International Telecommunication Union (ITU), *A silence compression scheme for G.729 optimized for terminals conforming to Recommendation V.70*, ITU-T Recommendation G.729 Annex B, Nov. 1996.

[39] MPEG-4 Video Group, "MPEG-4 Overview," available at http://mpeg.telecomitalialab.com/, Mar. 2002.

[40] International Telecommunication Union (ITU), *Video coding for low bit rate communication*, ITU-T Recommendation H.263, Feb. 1998.

[41] C. Chuah and R. H. Katz, "Characterizing Packet Audio Streams from Internet Multimedia Applications," in *Proc. of International Communications Conference (ICC 2002)*, New York, NY, April 2002.

[42] Y. Xu and R. Guerin, "On evaluating loss performance deviation: a simple tool and its pratical implications," in *Quality of Service in multiservice IP networks (QoSIP2003)*, Milan, Italy, Feb. 2003, pp. 1–18.

[43] W. Wang, S. C. Liew, and V. O. K. li, "Solutions to performance problems in VoIP over a 802.11 wireless LAN," *IEEE Trans. Veh. Technol.*, vol. 54, no. 1, pp. 366–384, Jan. 2005.

[44] P. Seeling, F. H. Fitzek, and M. Reisslein, *Video Traces for Network Performance Evaluation - A Comprehensive Overview and Guide on Video Traces and Their Utilization in Networking Research*. Springer, November 2006.

[45] G. Boggia, P. Camarda, L. A. Grieco, and S. Mascolo, "Feedback-based control for providing real-time services with the 802.11e mac," *IEEE/ACM Trans. on Networking*, vol. 15, no. 2, pp. 323–333, Apr. 2007.

[46] G. Boggia, P. Camarda, and A. D'Alconzo, "Performance of markov models for frame-level errors in ieee 802.11 wireless lans," *to appear on Int. J. of Commun. Systems.*, 2008.