Real-time applications in 802.11 WLAN using feedback-based bandwidth allocation

A. Barbuzzi, G. Binetti, G. Boggia, P. Camarda, L. A. Grieco, and S. Mascolo

DEE - Politecnico di Bari, Via E. Orabona, 4 - 70125 Bari Italy
{a.barbuzzi, g.binetti, g.boggia, camarda, a.grieco, mascolo}@poliba.it

Abstract. The Hybrid Coordination Function (HCF) has been recently proposed by the 802.11e working group in order to provide real-time services in Wireless Local Area Networks (WLANs). HCF is made of a contention-based channel access, known as Enhanced Distributed Coordination Access (EDCA), and of a HCF Controlled Channel Access (HCCA), which requires a centralized controller, called Hybrid Coordinator (HC). This paper proposes two feedback-based bandwidth allocation algorithms exploiting HCCA for dynamically assigning the WLAN channel bandwidth to mobile stations hosting real-time traffic streams. Proposed algorithms, which have been referred to as Feedback Based Dynamic Scheduler (FBDS) and Proportional Integral (PI)-FBDS, have been designed using classic discrete-time feedback control theory. Simulation results, obtained using the *ns*-2 simulator, have shown that, unlike the simple scheduler proposed by the 802.11e working group, both FBDS and PI-FBDS provide a real-time service regardless of the network load. Moreover, when the PI-FBDS is used, the best trade-off between one-way packet delays and network utilization is achieved.
Keywords. Quality of service, Wireless LAN, Dynamic bandwidth allocation, Feedback control.

1 Introduction

Although 802.11 Wireless Local Area Networks (WLANs) [1] are nowadays very broadly diffused, 802.11 Medium Access Control (MAC) is not well suited for providing real-time services [2]. Recently, to overcome this limitation, the 802.11e working group has proposed: the Hybrid Coordination Function (HCF), which is an enhanced access method; a Call Admission Control (CAC) algorithm; specific signaling messages for service request and Quality of Service (QoS) level negotiation; four Access Categories (ACs) with different priorities to map QoS users' requirements [3].

The HCF is made of a contention-based channel access, known as the Enhanced Distributed Coordination Access (EDCA), and of a HCF Controlled Channel Access (HCCA), which requires a centralized controller, called the Hybrid Coordinator (HC), which is generally located at the access point. EDCA operates as the basic DCF access method [1], but using different contention parameters per AC. In this way, a service differentiation among ACs is statistically pursued . EDCA parameters have to be properly set to provide prioritization of ACs. Tuning them in order to meet specific QoS needs is a current research topic [2]. In particular, 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, but it starves flows with lower priority, especially at high network load [4, 5]. For that purpose, adaptive algorithms that dynamically tune EDCA parameters have been recently proposed in [2, 6]; however, the effectiveness of these heuristic schemes have been proved only using simulations and no theoretical bounds on their performance in a general scenario has been derived.

On the other hand, with the HCCA, the HC is responsible for assigning the right to transmit at nodes hosting applications with QoS requirements, i.e., to perform dynamic bandwidth allocation within the WLAN. However, the 802.11e draft does not specify an effective bandwidth allocation algorithm for providing the QoS required by real-time flows; it only suggests a simple scheduler that uses static values declared by data sources for providing a Constant Bit Rate (CBR) service. As a consequence, this scheduler is not well suited for bursty media flows [7]. An adaptive version of the simple scheduler, which is based on the Delay-Earliest Due-Date algorithm, has been proposed in [7]. However, this scheduler does not exploit any feedback information from mobile stations, but implements a trial and error procedure to discover the optimal amount of resources to assign to each AC. The Fair Scheduling scheme proposed in [8] allocates the WLAN channel bandwidth to wireless nodes in order to fully deplete transmit queues, which are estimated by taking into account the delayed feedbacks from the wireless nodes.

This paper proposes two feedback-based bandwidth allocation algorithms exploiting HCCA. Proposed algorithms, which have been referred to as Feedback Based Dynamic Scheduler (FBDS) and Proportional Integral (PI)-FBDS, have been designed using classic discrete-time feedback control theory. Simulation results, obtained using the *ns*-2 simulator [9], have shown that, unlike the simple scheduler proposed by the 802.11e working group, both FBDS and PI-FBDS provide a real-time service regardless of the network load. Moreover, when the PI-FBDS is used, the best trade-off between one-way packet delays and network utilization is achieved.

The rest of the paper is organized as follows: Section 2 gives an overview of the HCCA method; in Section 3 FBDS and PI-FBDS algorithms are proposed; Section 4 shows simulation results; finally, the last Section draws the conclusions.

2 Overview of the HCCA Method

The core of the 802.11e proposal is the HCF, which is responsible for assigning TXOPs (Transmission Opportunities) to each AC in order to satisfy its QoS needs. TXOP is defined as the time interval during which a station has the right to transmit and is characterized by a starting time and a maximum duration. The contiguous time during which TXOPs are granted to the same station with

QoS capabilities (i.e., a QoS station, QSTA) is called Service Period (SP). The interval T_{SI} between two successive SPs is called Service Interval [3].

HCCA method combines some EDCA characteristics with some features of the Point Coordination Function (PCF) scheme, which is an optional contention-free access method defined by the 802.11 standard [1]. The time is divided into repeated periods, called *SuperFrames* (SFs). Each superframe starts with a beacon frame after which, for legacy purpose, there could be a contention free period (CFP) for PCF access. The remaining part of the superframe forms the Contention Period (CP), during which QSTAs contend to access the radio channel using the EDCA mechanism (see Fig. 1). During the CP or the CFP, the HC can

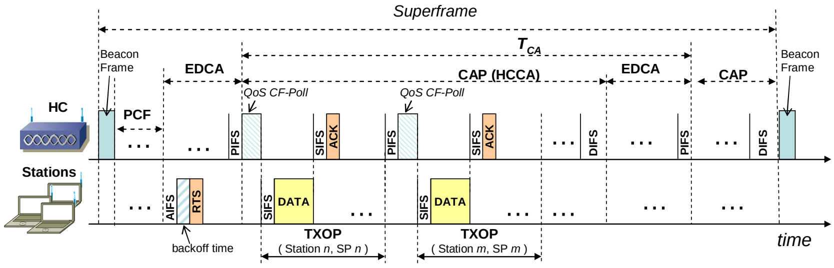

Fig. 1. Scheme of a superframe using the HCF controlled access method.

start a Contention Access Phase (CAP). During the CAP, only QSTAs, polled and granted with the *QoS CF-Poll frame*, are allowed to transmit during their TXOPs. Thus, the HC implements a prioritized medium access control.

The number of CAPs and their locations in each superframe are chosen by the HC in order to satisfy QoS needs of each station. Moreover, at least one CP interval, long enough to transmit a data frame with the maximum size at the minimum rate, must be contained in each superframe; this CP interval can be used for management tasks, such as associations of new stations, new traffic negotiations, and so on. CAP length cannot exceed the value of the system variable *dot11CAPLimit*, which is advertised by the HC in the Beacon frame when each superframe starts [3].

The simple scheduler designed in the draft [3] states that the $TXOP_i$ assigned to the i^{th} queue should be computed as follows:

$$TXOP_i = max\left\{\frac{N_i \cdot L_i}{C_i} + O, \frac{M}{C_i} + O\right\} \tag{1}$$

where, with respect to the i^{th} queue, L_i is the nominal size of MAC Service Data Units (MSDUs); C_i is the rate at which the data are transmitted over the WLAN; O is the protocol overhead; M is the maximum MSDU size; and $N_i = \left\lceil \frac{T_{SI} \cdot \rho_i}{L_i} \right\rceil$, where ρ_i is the Mean Data Rate associated with the queue and T_{SI} is the Service Interval.

According to IEEE 802.11e specifications, each QSTA can feed back queue length of each AC to the HC in the frames' headers. As will be shown in this paper, this information can be fruitfully exploited to design novel HCCA-based dynamic bandwidth allocation algorithms using feedback control theory [10].

2.1 QoS signalling

In the 802.11e proposal [3] each Traffic Stream (TS), i.e., a data flow with QoS needs, is described by a *Traffic SPECification* (TSPEC), which indicates the main characteristics of the stream [3]. Specific signalling has been introduced to manage new TS requests and QoS provisioning. In particular, when a new TS has to be started, the QSTA issues a setup phase by generating a message, which is known as *Mac Layer Management Entity (MLME)-ADDTS request* and contains the TSPEC of the stream.

This request message is sent to the HC which decides whether to admit the stream with the specified TSPEC, or to refuse it, or to not admit it suggesting an alternative TSPEC. The decision of the HC is transmitted with the *ADDTS response* message.

The QSTA receives this management frame and sends a *MLME-ADDTS confirm* message specifying whether the HC response meets its needs or not; if not, the whole process can be repeated [3].

2.2 Call Admission Control

In a IEEE 802.11 network the HC is used as admission control unit. Since the QoS facility supports two access mechanisms, there are two distinct admission control schemes: one for the contention-based access and the other for the controlled-access. Herein, we will focus on the latter mechanism. Details regarding the contention-based admission control can be found in [3].

Let m be the number of admitted flows. When in the presence of a new TS admission request, the admission control unit in the HC calculates the TXOP duration needed by the stream ($TXOP_{m+1}$) as imposed by the simple scheduler (see Eq. (1)). The stream is admitted if the following inequality is satisfied:

$$\frac{TXOP_{m+1}}{T_{SI}} + \sum_{i=1}^{m} \frac{TXOP_i}{T_{SI}} \leq \frac{T - T_{CP}}{T} \tag{2}$$

where T indicates the superframe duration, and T_{CP} is the time during which EDCA is used for frame transmission in the superframe.

3 FBDS and PI-FBDS Algorithms

In this section, FBDS and PI-FBDS algorithms will be designed using feedback-based control theory. We will assume that both algorithms, running at the HC, allocate the WLAN channel bandwidth to wireless stations hosting real-time applications, using HCCA functionalities. These allow the HC to assign TXOPs to ACs by taking into account their specific time constraints and transmission queue levels [5].

We will refer to a WLAN system made of an Access Point and a set of quality of service enabled mobile stations (QSTAs). Each QSTA has up to 4 queues, one for each AC in the 802.11e proposal.

Let T_{CA} be the time interval between two successive CAPs (see Fig. 1). Every time interval T_{CA}, assumed constant, the HC has to allocate the bandwidth that will drain each queue during the next CAP. We assume that at the beginning of each CAP, the HC is aware of all the queue levels $q_i,\ i = 1, \dots, M$ at the beginning of the previous CAP, where M is the total number of traffic queues in the WLAN system. [1]

The following discrete time linear model describes the dynamics of the i^{th} queue:

$$q_i(n+1) = q_i(n) + d_i(n) \cdot T_{CA} + u_i(n) \cdot T_{CA}, \qquad i = 1, \dots, M, \tag{3}$$

where, with respect to the i^{th} queue, $q_i(n) \geq 0$ is the queue level at the beginning of the n^{th} CAP; $u_i(n) \leq 0$ is the average depletion rate (i.e., the bandwidth assigned to drain the queue); $d_i(n) = d_i^s(n) - d_i^{CP}(n)$ is the difference between $d_i^s(n) \geq 0$, which is the average input rate at the queue during the n^{th} T_{CA} interval, and $d_i^{CP}(n) \geq 0$, which is the amount of data transmitted by the queue during the n^{th} T_{CA} interval, using EDCA, divided by T_{CA}.

The signal $d_i(n)$ is unpredictable since it depends on the behavior of the source that feeds the i^{th} queue and on the number of packets transmitted using EDCA. Thus, from a control theoretic perspective, $d_i(n)$ can be modelled as a disturbance. Without loss of generality, the following piece-wise constant model for the disturbance $d_i(n)$ can be assumed [11]:

$$d_i(n) = \sum_{j=0}^{+\infty} d_{0j} \cdot 1(n - t_j) \tag{4}$$

where $1(n)$ is the unitary step function, $d_{0j} \in \mathbb{R}$, and t_j is a time lag.

Due to the assumption (4), the linearity of the system (3), and the superposition principle that holds for linear systems, we will design the feedback control law by considering a step disturbance: $d_i(n) = d_0 \cdot 1(n)$.

3.1 The closed loop control scheme

Our goal is to design a control law that drives the queuing delay τ_i, experienced by each frame going through the i^{th} queue, to a desired target value τ_i^T that represents the QoS requirement of the AC associated to the queue.

[1] This is a worst case assumption, in fact, queue levels are fed back using frame headers as described in Sec. 2; as a consequence, if the i^{th} queue length has been fed at the beginning of the previous CAP, then the feedback signal might be delayed up to T_{CA} seconds.

We will consider the closed loop control system shown in Fig. 2, where the set point q_i^T has been set equal to 0, i. e., we would ideally obtain empty queues. Regarding the controller transfer function $G_i(z)$, we will focus the attention on two possible kinds of controller: a proportional (P) controller, obtained by setting $G_i(z) = k_{p_i}$, and a proportional and integral (PI) controller, obtained as $G_i(z) = k_{p_i}\left(1 + \dfrac{z}{z-1} \cdot \dfrac{1}{T_{I_i}}\right)$. The corresponding bandwidth allocation algorithms will be referred to as Feedback Based Dynamic Scheduler (FBDS), and PI-FBDS.

Using a proportional controller. By considering the control scheme in Fig. 2 where $G_i(z) = k_{p_i}$ it is straightforward to compute the $\mathcal{Z}$-transform of $q_i(n)$ and $u_i(n)$:

$$Q_i(z) = \frac{z \cdot T_{CA}}{z^2 - z + k_{pi} \cdot T_{CA}} \cdot D_i(z); \quad U_i(z) = -\frac{k_{pi} \cdot T_{CA}}{z^2 - z + k_{pi} \cdot T_{CA}} \cdot D_i(z) \quad (5)$$

with $D_i(z) = \mathcal{Z}[d_i(n)]$.

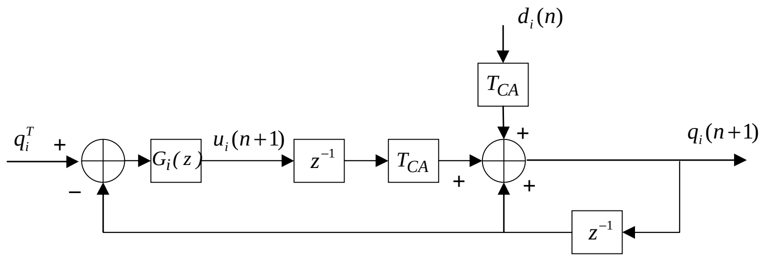

Fig. 2. Closed-loop control scheme based on HCCA.

From Eqs. (5) the system poles are $z_p = \frac{1 \pm \sqrt{1-4k_{pi} \cdot T_{CA}}}{2}$; thus, the system is asymptotically stable if and only if $|z_p| < 1$, that is:

$$0 < k_{pi} < 1/T_{CA}. \quad (6)$$

To investigate the steady-state behaviour of the control system, we apply the final value theorem to Eqs. (5).By considering that the $\mathcal{Z}$-transform of the step function $d_i(n) = d_0 \cdot 1(n)$ is $D_i(z) = d_0 \cdot \frac{z}{z-1}$, the following results turn out:

$$u_i(+\infty) = \lim_{n \to +\infty} u_i(n) = \lim_{z \to 1} (z-1) U_i(z) = -d_0; \qquad q_i(+\infty) = d_0/k_{pi},$$

which implies that the steady state queueing delay is:

$$\tau_i(+\infty) = |q_i(+\infty)/u_i(+\infty)| = 1/k_{pi}. \quad (7)$$

It is worth to note that $q(+\infty) > 0$ even if $q_i^T = 0$, which means that the proportional controller is not able to fully reject the step disturbance $d_0 \cdot 1(n)$.

From Eq. (7), it turns out that the following inequality has to be satisfied in order to achieve a steady-state delay smaller than τ_i^T:

$$k_{pi} \geq 1/\tau_i^T. \tag{8}$$

By considering inequalities (6) and (8),the T_{CA} parameter has to fulfill the following constraint:

$$T_{CA} < \min_{i=1..M} \tau_i^T. \tag{9}$$

From these results we conclude that, when a proportional controller is used, the gain k_{pi} can vary in the range $\left[\frac{1}{\tau_i^T}, \frac{1}{T_{CA}}\right[$. We will set $k_{pi} = 1/\tau_i^T$, i.e., at its lowest admissible value. This choice allows, for a given feedback queue level $q_i(n)$, the lowest bandwidth assignment ensuring bounded delays, so that, a cautios usage of the WLAN channel bandwidth is achieved.

Using a PI controller. Similarly to the case of the proportional controller, by considering Fig. 2 where $G_i(z) = k_{p_i}\left(1 + \frac{z}{z-1} \cdot \frac{1}{T_{I_i}}\right)$ and $q_i^T = 0$, after a little algebra, the following $\mathcal{Z}$-transforms of $q_i(n)$ and $u_i(n)$ can be obtained:

$$Q_i(z) = \frac{T_{CA}T_{I_i}z(z-1)}{T_{I_i}z^3 - 2T_{I_i}z^2 + (T_{I_i} + T_{CA}k_{p_i}T_{I_i} + T_{CA}k_{p_i})z - T_{CA}k_{p_i}T_{I_i}} D_i(z); \tag{10}$$

$$U_i(z) = \frac{T_{CA}k_{p_i}T_{I_i} - T_{CA}k_{p_i}T_{I_i}z - T_{CA}k_{p_i}z}{T_{I_i}z^3 - 2T_{I_i}z^2 + (T_{I_i} + T_{CA}k_{p_i}T_{I_i} + T_{CA}k_{p_i})z - T_{CA}k_{p_i}T_{I_i}} D_i(z). \tag{11}$$

By applying the Jury criterion [10], the system in Fig. 2 is asymptotically stable if the following inequalities are satisfied:

$$k_{p_i} < 1/T_{CA}; \qquad T_{I_i} > 1/\left(1 - T_{CA}k_{p_i}\right) . \tag{12}$$

Also in this case, to investigate the steady-state behaviour of the control system, we apply the final value theorem to Eq. (10), thus obtaining: $q_i(+\infty) = 0$, which implies that the steady state queuing delay is zero.

This result is due to the integral action of the controller, which is able to fully reject the step disturbance at steady state. In this way, the parameter set of the PI regulator is only subject to the stability constraints (12). As a consequence we have more degrees of freedom in the choice of k_{p_i} and T_{I_i} with respect to the case of the proportional controller.

When the PI controller is used, it might happen that the depletion rate computed by the controlled $|u_i(n+1)|$ is larger than $q_i(n)/T_{CA}$, which is the amount of bandwidth required to fully deplete the i^{th} queue during the $(n+1)^{th}$ CAP. This assignement would obviously waste WLAN resources. To overcome this drawback, we will employ the following shortcut:

$$u_i(n+1) \leftarrow max\left\{u_i(n+1), -q_i(n)/T_{CA}\right\}. \tag{13}$$

where, taking into account that $u_i \leq 0$, the term $-q_i(n)/T_{CA}$ is the depletion rate needed to fully empty the queue.

The rationale of eq. (13) is to correct the bandwidth allocation that would be performed by the PI regulator by not allowing a bandwidth assignment larger than the one than would fully deplete the i^{th} queue.

3.2 TXOP assignment

We have seen in Sec. 2 that, every time interval T_{CA}, the HC allocates TXOPs to mobile stations in order to meet the QoS constraints. Herein, we shows how to transform the bandwidth u_i into a $TXOP_i$ assignment. In particular, if the i^{th} queue is drained at rate C_i, the following relation holds:

$$TXOP_i(n) = \frac{|u_i(n) \cdot T_{CA}|}{C_i} + O \tag{14}$$

where $TXOP_i(n)$ is the TXOP assigned to the i^{th} queue during the n^{th} CAP and O is the protocol time overhead. The extra quota of TXOP due to the overhead O depends on the number of MSDUs corresponding to the amount of data $|u_i(n) \cdot T_{CA}|$ to be transmitted. O could be estimated by assuming that all MSDUs have the same nominal size specified into the TSPEC. Moreover, when $|u_i(k) \cdot T_{CA}|$ does not correspond to a multiple of MSDUs, the TXOP assignment will be rounded in excess in order to guarantee a queuing delay equal or smaller than the target value τ_i^T.

3.3 Channel saturation

The above bandwidth allocation algorithm is based on the implicit assumption that the sum of the TXOPs assigned to each traffic stream is smaller than the maximum CAP duration, which is defined by the system variable *dot11CAPLimit*; this value can be violated when the network is saturated. In order to avoid heavy channel saturations, we will adopt a CAC scheme obtained by improving the one proposed by the 802.11e working group. However, since transient overloads cannot be avoided due to the burstiness of the multimedia flows, when for a given n_0 we have that: $\sum_{i=1}^{M} TXOP_i(n_0) > dot11CAPLimit$, each computed $TXOP_i(n_0)$ is decreased by an amount $\Delta TXOP_i(n_0)$, so that the following capacity constraints is satisfied:

$$\sum_{i=1}^{M} [TXOP_i(n_0) - \Delta TXOP_i(n_0)] = dot11CAPLimit\ . \tag{15}$$

In particular, the generic amount $\Delta TXOP_i(n_0)$ is evaluated as a fraction of the total amount $\Delta = \sum_{j=1}^{M} TXOP_i(n_0) - dot11CAPLimit$, as follows:

$$\Delta TXOP_i(n) = \frac{TXOP_i(n)C_i}{\sum_{j=1}^{M} [TXOP_j(n)C_j]} \Delta . \tag{16}$$

Notice that Eq. (16) provides a $\Delta TXOP_i(n_0)$, which is proportional to $TXOP_i(n_0)C_i$; in this way, connections transmitting at low rates are not too much penalized.

3.4 Call Admission Control

When the number of multimedia flows sharing the WLAN increases, the channel saturates and delay bounds cannot be guaranteed [6]. Under these conditions, a Call Admission Control scheme is required to provide QoS. Herein, we describe the proposed CAC scheme, which exploits the 802.11e CAC proposal (see Sec. 2).

In particular, starting from the TXOPs allocated to the active traffic streams in each CAP, a new flow request is admitted if

$$\frac{TXOP_{m+1}}{T_{CA}} + \sum_{i=1}^{m} \frac{TXOP_i}{T_{CA}} \leq \frac{T - T_{CP}}{T} \tag{17}$$

where m is the number of admitted flows, T is the superframe duration, and T_{CP} is the time used by EDCA during the superframe.

Notice that the proposed CAC scheme given by eq. (17) has been obtained from the CAC suggested in the draft (see sec. 2.2), by replacing the constant TXOPs used by the simple scheduler with the time-varying ones allocated by the proposed bandwidth allocation algorithms. In this way, our suggested CAC takes into account the bandwidth actually used by the flows and not just the sum of the average source rates declared in the TSPECs.

4 Performance Evaluation

In order to assert the validity of the proposed bandwidth allocation algorithms in realistic scenarios, computer simulations involving voice, video and FTP data transfers have been run. We have used the *ns*-2 simulator [9] considering a scenario where a 802.11a wireless channel with data rate of 54Mbps is shared by a mix of 3α voice flows encoded with the G.729 standard [12], α MPEG-4 encoded video flows [13], α H.263 video flows [14], and α FTP best effort flows. Each flow is hosted by a wireless node. Therefore, in such a scenario the traffic load is directly related to the parameter α.

For video flows, we have used traffic traces available from the video trace library [15]. For voice flows, we have modeled the G.729 sources using Markov ON/OFF sources [16]. The ON period is exponentially distributed with mean 3 s and the OFF period has a truncated exponential pdf with an upper limit of 6.9 s and an average value of 3 s [17]. During the ON period, the source sends packets of 20 bytes every 20 ms (i.e., the source data rate is 8 kbps and we are considering two G.729 frames combined into one packet [18]). By considering the overheads of the RTP/UDP/IP protocol stack, during the ON periods the total rate over the wireless channel becomes 24 kbps. During the OFF period the rate is approximated by zero assuming the presence of a Voice Activity Detector (VAD).

During CAPs, stations access the channel with HCCA method, otherwise they use EDCA. In simulations, EDCA parameters have been set as suggested in [3].

The target delay τ_i^T has been set equal to 30 ms for the voice flow and 40 ms for the video flows. According to the 802.11 standard, in our *ns*-2 implementation the T_{CA} is expressed in Time Unit (TU), which is equal to 1024 μs [1]; we assume a T_{CA} of 29 TU in order to satisfy inequality (9). The value of the system variable *dot11CAPlimit* has been set in order to allow the transmission of at least 10 MSDUs of maximum size using EDCA, between two successive CAPs.

When FBDS is used, the proportional gain k_{p_i} is set equal to $1/\tau_i^T$ (see Sec. 2). With the PI-FBDS scheme, we consider the following parameter sets: $(k_{p_i} = 22s^{-1}, T_{I_i} = 10)$, $(k_{p_i} = 18s^{-1}, T_{I_i} = 7)$, and $(k_{p_i} = 14s^{-1}, T_{I_i} = 4)$.

The main characteristics of the considered multimedia flows are summarized in Table 1.

Table 1. Main features of the considered multimedia flows.

Type of flow	*Nominal (Maximum) MSDU Size* [Byte]	*Mean (Maximum) Data Rate* [Byte]	*Inactivity Interval* [s]
MPEG-4	1536 (2304)	770 (3300)	3
H.263 VBR	1536 (2304)	450 (3400)	3
G.729 VAD	60 (60)	12 (24)	10

Before starting data transmission, a multimedia source has to set up a new Traffic Stream as specified in Sec. 2.1. If the reply to the admission message for the new stream is not received within a Δ_{TO} timeout interval, the request is repeated up to a maximum number of times, N_{Adm}; in our simulations, we have chosen $N_{Adm} = 10$ and $\Delta_{TO} = 1.5$ s. If after the 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 Δ_{defer} with average value equal to 1 min. The duration of video flows is deterministic and equal to 10 min, whereas voice flows durations are exponentially distributed with 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 min. Each terminated flow is withdrawn from the polling list by the HC after that no more packets from that flow are received for a time equal to the Inactivity Interval reported in Table 1. Each simulation lasts 1 hour.

In the sequel, we will first compare the performances of the PI-FBDS, the FBDS, and the Simple scheduler, then, we will investigate the impact of the parameters of the PI-FBDS (i.e., k_{p_i} and T_{I_i}) on the WLAN behavior.

Tab. 2 reports the ratio of admitted flows for various values of the network load parameter α when PI-FBDS or FBDS or the simple scheduler are used. By looking at this table, it is straightforward to note that FBDS admits the smallest ratio of flows. On the other hand, almost all the flows are admitted when either the PI-FBDS or the Simple scheduler are used. The reason is that the Simple scheduler allocates TXOPs by taking into account only the declared average source rates, which can be much smaller than the actual source rates; so that, the CAC test realized with eq. (2) is always satisfied. When FBDS or PI-FBDS are used, the CAC test given by eq. (17) takes into account the actual load. The substantial difference between FBDS and PI-FBDS is that, as discussed in Sec. 2, the PI regulator gives the opportunity to choice its parameter set in a broader space of values with respect to the case of a proportional controller. In other words, we can select the parameter set of the PI regulator to filter out high frequency components of the traffic load better than in the case of the P regulator. As a consequence, PI-FBDS allows the CAC test of eq. (17) to consider only the low frequency components of the network load, and to admit a larger number of flows. In other words, PI-FBDS is able to smooth the peaks of the traffic load better than FBDS. As a consequence, when PI-FBDS is used the probability that a new starting flow finds a fully occupied wireless channel is smaller.

Table 2. Ratios of admitted flows.

α	PI-FBDS ($k_{p_i} = 14s^{-1}$, $T_{I_i} = 4$)	FBDS	Simple Scheduler
5	100%	100%	100%
10	100%	89%	100%
12	99%	59%	100%

At this point, due to the smaller number of admitted traffic streams, we would expect lower one-way packet delays when FBDS is employed with respect to the one-way packet delay values obtained with the Simple scheduler or PI-FBDS. This is shown in Figs. 3, 4, and 5 where the average one-way packet delays experienced by, respectively, the MPEG, H.263, and G.729 flows are reported. By looking at those figures, we can note that FBDS obtains the smallest delays due the smallest quota of admitted flows. The surprising result is that, with high load (i.e., $\alpha = 12$) although the Simple scheduler and PI-FBDS allow almost the same ratio of admitted flows, delays obtained with PI-FBDS are one order of magnitude smaller than those obtained with the Simple scheduler. The latter observation highlights that if the scheduling strategy works fine, i.e., the network resources are carefully administered, a good trade-off between the number of admitted flows and the one-way packet delay can be achieved, also with high load.

Table 3 reports the average and peak superframe utilization in HCCA mode. It shows that the Simple scheduler requires the highest average quota of WLAN

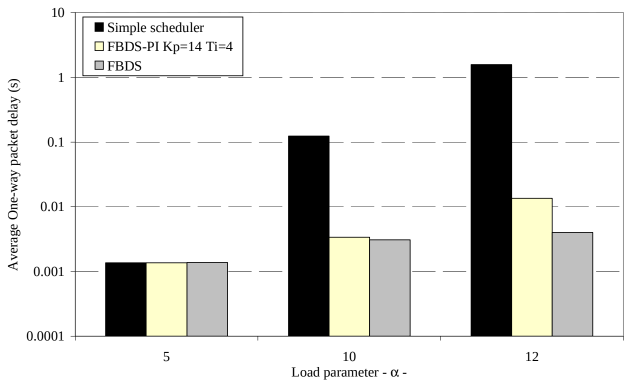

Fig. 3. Average one-way packet delay of the MPEG4 flows.

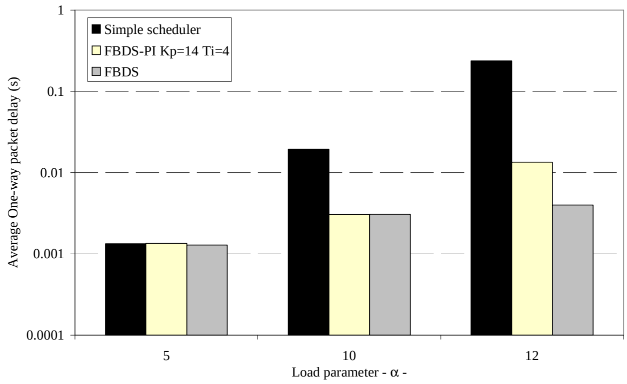

Fig. 4. Average one-way packet delay of the H.263 flows.

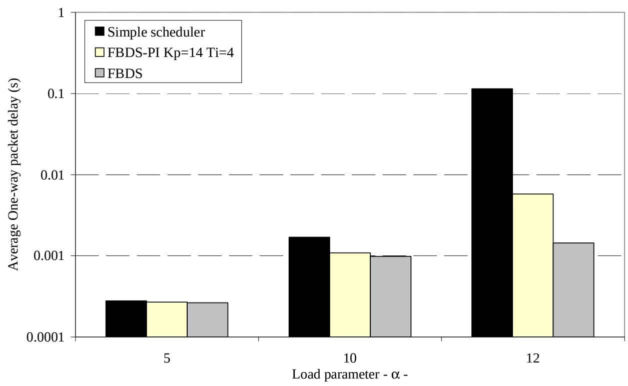

Fig. 5. Average one-way packet delay of the G.729 flows.

resources. In fact, it does not adapt the quota of allocated resources to the actual load because it provides a CBR service. For the same reason, the peak superframe utilizations achieved by the Simple scheduler for $\alpha = 10$ and $\alpha = 12$, i.e., at high traffic load, are smaller than those provided by FBDS and PI-FBDS. This result clearly highlights that the proposed control schemes enable a more proper usage of the bandwidth and allows the bandwidth requirements of the real-time flows to be tracked.

Table 3. Average (Peak) Superframe usage in HCCA mode.

α	PI-FBDS ($k_{p_i} = 14s^{-1}$, $T_{I_i} = 4$)	FBDS	Simple Scheduler
5	6.82(26)%	6.95(31)%	25.62(32.42)%
10	15.15(84)%	15.22(82)%	52.4(62.58)%
12	29.85(81)%	17.66(80)%	63.75(74.98)%

Table 4 reports the overall goodput achieved by the FTP flows, measured as the total amount of data received by FTP receivers over the simulation duration. It shows that FTP flows get the bandwidth left unused by the real-time flows. In fact, when FBDS is used, FTP flows get the highest goodput because, with FBDS, the smallest quota of real-time flows is admitted (see Tabs. 2 and 3).

Table 4. Overall goodput of the FTP flows (Mbps).

α	PI-FBDS ($k_{p_i} = 14s^{-1}$, $T_{I_i} = 4$)	FBDS	Simple Scheduler
5	4.57	4.58	4.57
10	4.32	4.38	4.27
12	2.47	4.09	2.32

In order to investigate the influence of the parameters k_{p_i} and T_{I_i} on system performance with PI-FBDS, we have considered the following parameter sets: ($k_{p_i} = 14s^{-1}, T_{I_i} = 4$), ($k_{p_i} = 18s^{-1}, T_{I_i} = 7$) and ($k_{p_i} = 22s^{-1}, T_{I_i} = 10$). Tab. 5 shows the ratios of admitted flows obtained for various values of the load parameter α. The first observation that turns out by looking at these results

Table 5. Ratios of admitted flows using PI-FBDS with various parameter sets.

α	$k_{p_i} = 14s^{-1}, T_{I_i} = 4$	$k_{p_i} = 18s^{-1}, T_{I_i} = 7$	$k_{p_i} = 22s^{-1}, T_{I_i} = 10$
5	100%	100%	100%
10	100%	100%	99%
12	99%	98%	84%

is that, for increasing values of the two parameters, the ratio of admitted flows diminishes. This is due to the fact that as K_{pi} and T_{Ii} increase, the PI-FBDS becomes similar to FBDS. In fact, for increasing values of T_{Ii}, the integral action becomes less important, whereas, for increasing values of K_{pi}, the control-loop becomes faster.

Obviously, the one-way packet delays experienced by the flows diminish when the number of admitted streams decreases; this result is shown in Figs. 6 and 7, where the one way packet delays experienced by, respectively, the MPEG and G.729 flows are reported. Similar results, not shown here due to lack of space, have been obtained fot the H.263 flows.

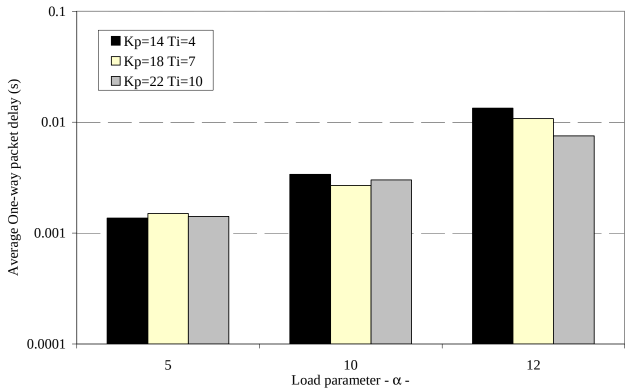

Fig. 6. Average one-way packet delay of the MPEG4 flows when using PI-FBDS with various parameter sets.

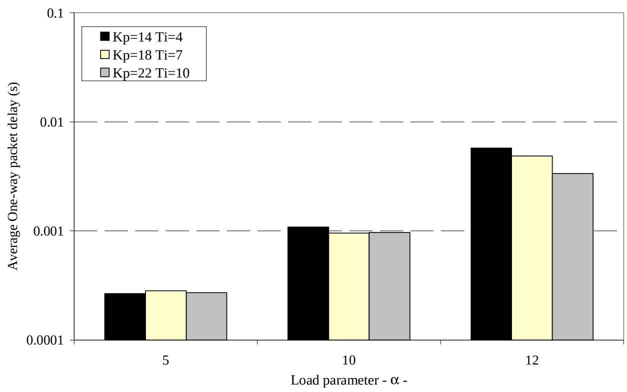

Fig. 7. Average one-way packet delay of the G.729 flows when using PI-FBDS with various parameter sets.

Tab. 6 reports the average and peak superframe utilization in HCCA mode using PI-FBDS with various parameter sets. Obviously, average and peak superframe usages follow the behaviour of the ratio of admitted flows (see Tab. 5). Finally, Table 7 reports the overall goodput achieved by the FTP flows. Also in this case, FTP flows get the highest goodput when the quota of real-time flows admitted is the smallest, i. e. for $k_{p_i} = 22s^{-1}$ and $T_{I_i} = 4$ (see Tab. 5).

Table 6. Average (Peak) Superframe using PI-FBDS with various parameter sets.

α	$k_{p_i} = 14s^{-1}, T_{I_i} = 4$	$k_{p_i} = 18s^{-1}, T_{I_i} = 7$	$k_{p_i} = 22s^{-1}, T_{I_i} = 10$
5	6.82(26)%	6.91(30)%	6.95(29)%
10	15.15(84)%	15.03(82)%	15.34(73)%
12	29.85(81)%	27.69(81)%	24.24(81)%

Table 7. Overall goodput of FTP flows (Mbps) using PI-FBDS with various parameter sets.

α	$k_{p_i} = 14s^{-1}, T_{I_i} = 4$	$k_{p_i} = 18s^{-1}, T_{I_i} = 7$	$k_{p_i} = 22s^{-1}, T_{I_i} = 10$
5	4.57	4.57	4.58
10	4.32	4.32	4.38
12	2.47	2.47	4.09

5 Conclusion

In this paper a HCCA-based control theoretic framework for addressing the first hop bandwidth allocation issue using the 802.11e MAC has been proposed. This framework has been used to design two dynamic bandwidth allocation algorithms, which have been referred to as FBDS and PI-FBDS. Simulation results obtained using the *ns*-2 simulator have shown that, unlike the simple scheduler proposed by the 802.11e WG, both FBDS and PI-FBDS ensure bounded delays also at high network loads. Moreover, when the PI-FBDS is used, the best trade-off between one-way packet delays and network utilization is achieved.

References

1. 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. ANSI/IEEE Std. 802.11, ISO/IEC 8802-11. First edn. (1999)

2. Ni, Q., Romdhani, L., Turletti, T.: A survey of QoS enhancements for IEEE 802.11 wireless LAN. Jour. of Wirel. Commun. Mob. Comput. **4** (2004) 1–20
3. IEEE 802.11 WG: Draft Amendment to Standard for Information Technology - Telecommunications and Information Exchange between Systems - LAN/MAN Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Quality of Service (QoS) Enhancements. IEEE 802.11e/D10.0. (2004)
4. Kong, Z., Tsang, D.H.K., Bensaou, B., Gao, D.: Performance Analysis of IEEE 802.11e Contention-Based Channel Access. IEEE Journal of Selected Areas in Communications **22** (2004) 2095–2106
5. Boggia, G., Camarda, P., Grieco, L.A., Mascolo, S.: Feedback based bandwidth allocation with call admission control for providing delay guarantees in IEEE 802.11e networks. Computer Communications, Elsevier **28** (2005) 325–337
6. Xiao, Y., Li, H., Choi, S.: Protection and guarantee for voice and video traffic in IEEE 802.11e Wireless LANs. In: IEEE Infocom, Hong Kong (2004)
7. Grilo, A., Macedo, M., Nunes, M.: A scheduling algorithm for QoS support in IEEE 802.11e networks. IEEE Wireless Communications (2003) 36–43
8. Ansel, P., Ni, Q., Turletti, T.: Fhcf: A fair scheduling scheme for 802.11e wlan. Technical Report 4883, Institut National de Recherche en Informatique et en Automatique (INRIA) (2003)
9. Ns-2: Network simulator. available at http://www.isi.edu/nsnam/ns (2004)
10. Astrom, K.J., Wittenmark, B.: Computer controlled systems: theory and design. 3 edn. Prentice Hall, Englewood Cliffs (1995)
11. Mascolo, S.: Congestion control in high-speed communication networks using the smith principle. Automatica, Special Issue on Control methods for communication networks **35** (1999) 1921–1935
12. International Telecommunication Union (ITU): Coding of Speech at 8 kbit/s using Conjugate-Structure Algebraic-Code-Excited Linear Prediction (CS-ACELP). ITU-T Recommendation G.729. (1996)
13. MPEG-4 Video Group: Mpeg-4 overview. available at http://mpeg.telecomitalialab.com/ (2002)
14. International Telecommunication Union (ITU): Video coding for low bit rate communication. ITU-T Recommendation H.263. (1998)
15. Video Trace Library. (available at http://trace.eas.asu.edu/)
16. Chuah, C., Katz, R.H.: Characterizing Packet Audio Streams from Internet Multimedia Applications. In: Proc. of International Communications Conference (ICC 2002), New York, NY (2002)
17. Xu, Y., Guerin, R.: On evaluating loss performance deviation: a simple tool and its pratical implications. In: Quality of Service in multiservice IP networks (QoSIP2003), Milan, Italy (2003) 1–18
18. Wang, W., Liew, S.C., li, V.O.K.: Solutions to performance problems in VoIP over a 802.11 wireless LAN. IEEE Trans. Veh. Technol. **54** (2005) 366–384