Providing Delay Guarantees in IEEE 802.11e Networks

A. Annese, G. Boggia, P. Camarda, L. A. Grieco and S. Mascolo
DEE - Politecnico di Bari, Via E. Orabona, 4 - 70125 Bari (Italy)
Email: {a.annese, g.boggia, camarda, a.grieco, mascolo}@poliba.it

Abstract—This paper proposes a Feedback Based Dynamic Scheduler (FBDS) to allocate the first-hop bandwidth in the IEEE 802.11e using the HCF controlled channel access. Properties of the proposed control algorithm are investigated both theoretically and using ns-2 simulations. The obtained results show that the proposed FBDS algorithm succeeds in guaranteeing delay bounds required by audio/video applications in the presence of a very broad set of traffic conditions and network loads.

I. INTRODUCTION

The interest for ubiquitous network services is leading to a wide development of Wireless Local Area Networks (WLANs) as access systems. Moreover, the need for wireless multimedia transmission services is continuously growing [1]. For these reasons the IEEE Working Group 802.11e has proposed the new Hybrid Coordination Function (HCF) medium access control mechanism as an innovative framework to address the first hop bandwidth allocation issue in WLAN [2]. This paper proposes a novel control theoretic approach for designing a feedback based channel bandwidth allocation algorithm with the HCF controlled channel access, which guarantees delay constraints of audio/video applications. Properties of the proposed Feedback Based Dynamic Scheduler (FBDS) are theoretically investigated; moreover, ns-2 simulations are developed to highlight the effectiveness of the proposed control algorithm in the presence of realistic traffic conditions. The FBDS algorithm is compared with the static one proposed by 802.11e and with the DCF and EDCA access methods, described in the next section. Simulation results show that FBDS is able to guarantee delay bounds required by audio/video applications in presence of a large set of traffic conditions and network loads, whereas other schemes fail.

The rest of the paper is organized as follows: Section II gives an overview of the 802.11 MAC protocol and of QoS enhancements; Section III describes the FBDS allocation algorithm; Section IV reports ns-2 simulation results. Finally, the last section draws the conclusions.

II. THE IEEE 802.11 MAC

The basic 802.11 MAC protocol is the Distributed Coordination Function (DCF), which is based on the Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) mechanism: for each frame a station listens the channel before transmitting; if a station detects an idle channel for a minimum interval time called DCF Interframe Space (DIFS), then it transmits immediately a MAC Protocol Data Unit (MPDU). Otherwise, if the medium is sensed busy, transmission is deferred until the channel is sensed idle for a DIFS period plus an additional random backoff time [3]. In order to try to support time-sensitive services, the 802.11 standard defines the Point Coordination Function (PCF) as an optional access method which provides a contention-free medium access. The Point Coordinator (PC) polls the stations asking for time-sensitive service and allows them to transmit a data frame without channel contention. With PCF, the time is divided into repeated periods, called SuperFrames (SFs), which consist of a Contention Period (CP) and a Contention Free Period (CFP). During the CP, the channel is accessed using DCF whereas, during the CFP, is accessed using the the PCF. It is worth to note that the PC cannot adaptively allocate the wireless channel capacity by taking into account the status of mobile stations, because it does not know the starting time and the transmission duration of the polled stations under its coordination [4].

A. IEEE 802.11e QoS enhancements

The IEEE 802.11e Working Group has proposed a new MAC mechanism for supporting QoS in WLAN called the Hybrid Coordination Function (HCF) [2]. The HCF has a contention-based channel access, known as the Enhanced Distributed Coordination Access (EDCA), and a HCF Controlled Channel Access (HCCA). Stations operating under 802.11e specifications are usually known as enhanced stations or QoS Stations (QSTAs). The use of the HCF requires a centralized controller, which is called the Hybrid Coordinator (HC) and is generally located at the access point, which connects stations with a wired backbone. Similarly to the PCF, the time is partitioned in a infinite series of superframes. In order to provide service differentiation, the 802.11e defines 8 Traffic Categories (TCs) with the priority values of the IEEE 802.1Q standard [5]. To satisfy QoS requirements to each TC, the concept of TXOP (Transmission Opportunity) is introduced, defined as the time interval during which a station has the right to transmit [2], [4].

1) The EDCA method: defines four Access Categories (ACs) to support the mentioned eight TCs (there is a map between ACs and TCs; for details see [2]) It operates as the basic DCF access method with the difference of using different contention parameters per access category. In this way, a service differentiation among ACs is statistically pursued [6].

2) The HCCA method: combines some of the EDCA characteristics with some of the PCF basic features as it is shown in
remains idle for at least a PIFS interval during the CP. Thus, the HC
starts a Contention Access Phase (CAP). During the CAP, only polled
(special QoS CF-Poll frame) and granted QSTAs are allowed to transmit. As a consequence, 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. CAP length cannot
exceed the value of the system variable dot11CAPLimit [2].
Each CAP ends after the medium remains idle for a DIFS
interval. During each CAP, the HC assigns TXOPs to QSTAs
in order to meet predefined service rate, delay and/or jitter
requirements. In particular, the contiguous time during which
TXOPs are granted to the same STA is called Service Period.
The interval between two successive SPs is called Service Interval. The IEEE 802.11e specifications allow QSTAs to feed
back queue lengths to guarantee the typical QoS requirements of audio/video applications.

Each QSTA has N queues, with N ≤ 4, one for any AC in
the 802.11e proposal. Every interval time T_{CA}, the AP must allocate the bandwidth that will drain each queue during the
next CAP. We assume that at the beginning of each CAP, the
AP is aware of all the queue levels q_i. i = 1, ..., M, where
M is the total number of traffic queues in the WLAN system.

The dynamics of the i-th queue can be described by the
following discrete time linear model:

\[ q_i(k+1) = q_i(k) + [d_i^s(k) - d_i^{CP}(k) + u_i(k)] \cdot T_{CA} \] (1)

where \( q_i(k) \geq 0 \) is the i-th queue level at the beginning of the
k-th CAP, \( d_i^s(k) \geq 0 \) is the average input rate at the i-th
queue during the k-th interval between CAPs, \( d_i^{CP}(k) \geq 0 \)
the amount of data transmitted by the i-th queue during the
k-th CP divided by \( T_{CA} \), and \( u_i(k) \leq 0 \) is the average depletion
rate of the i-th queue (i.e., the bandwidth assigned to drain the
i-th queue). Eq. (1) may be rewritten as follows:

\[ q_i(k+1) = q_i(k) + d_i(k) \cdot T_{CA} + u_i(k) \cdot T_{CA} \] (2)

where \( d_i(k) = d_i^s(k) - d_i^{CP}(k) \).

The bandwidth \( u_i(k) \) is dynamically assigned by the HC
for allowing data transmission during the CAPs. On the other hand, \( d_i(k) \) is unpredictable since it depends on the
behavior of the traffic source that feeds the i-th queue and on
the number of packets transmitted during the CPs. From a
control theoretical perspective this function can be modelled
as a disturbance. Without loss of generality, the following
model for the disturbance \( d_i(k) \) can be assumed [7]:
\( d_i(k) = \sum_{j=0}^{\infty} d_{ij} \cdot 1(k - t_j) \), where \( 1(k) \) is the unitary step function,
\( d_{ij} \in \mathbb{R} \), and \( t_j \) is a time lag. Due to this assumption and
the superposition principle that holds for linear systems,
we will design the feedback control law by considering a step
disturbance: \( d_i(k) = d_{0i} \cdot 1(k) \). The goal of the control law we
want to design is to drive the queueing delay \( \tau_i \) experienced by each frame going through the i-th queue to a desired target value \( \tau_i^T \). The target queueing delay \( \tau_i^T \) represents the QoS
requirement of the AC associated to the i-th queue. For that
purpose we consider the control scheme in Fig. 2, where
both a proportional controller and feedback disturbance
compensation have been exploited. For the time being, the
transfer function of the feedback branch \( H(z) \), which models feedback delays, will be assumed equal to one, then the impact
of a delayed feedback will be analyzed. By considering Fig.
2, the Z-transforms of \( u_i(k) \) and \( q_i(k) \) are:

\[
U_i(z) = -\frac{k_i \cdot TXOP}{z \cdot (z - 1 + k_i \cdot TCA)} \cdot D_i(z) \quad (3)
\]

\[
Q_i(z) = \frac{z + k_i \cdot TCA}{z \cdot (z - 1 + k_i \cdot TCA)} \cdot D_i(z). \quad (4)
\]

From Eq. (4), the system has a pole at \( z_p = 1 - k_i \cdot TCA \) and it is asymptotically stable if and only if \( |z_p| < 1 \). The latter condition turns out the following upper bound on the admissible proportional gain \( k_i \):

\[
0 < k_i < 2/TCA. \quad (5)
\]

In the sequel, we will always assume that \( k_i \) satisfies the asymptotic stability condition (5).

By applying the final value theorem to Eqs. (3) and (4), where again \( D_i(z) = d_0 \cdot \frac{1}{z-1} \), the following result turns out:

\[
u_i(\infty) = -d_0; \quad q_i(\infty) = d_0 \cdot \tau_i^T,
\]

which implies that the the steady state queueing delay is:

\[
\tau_i(\infty) = |q_i(\infty)/u_i(\infty)| = \tau_i^T. \quad (6)
\]

The analysis proposed so far is based on the assumption that QSTAs feed back queue lengths \( q_i(k+1) \) to the HC at the beginning of each CAP. Actually, this feedback is asynchronous since it is carried by frame headers, as described in Sec. II-A. Thus, 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 \( TCA \) seconds. For this reason, now we investigate the stability of the proposed control system when in the presence of a delay of one step affecting the feedback branch. In this case \( H(z) = z^{-1} \) and from Fig. 2 it is easy to derive the system poles, which are \( z_p = \frac{1 \pm \sqrt{1 - 4 \cdot TCA \cdot k_i}}{2} \), which give an asymptotically stable system if and only if \( |z_p| < 1 \), that is:

\[
0 < k_i < 1/TCA. \quad (7)
\]

By considering the system reported in Fig. 2, where \( d_i(k) = d_0 \cdot 1(k) \), \( 0 < k_i < 1/TCA \), and \( H(z) = 1 \), a necessary and sufficient condition to avoid overshoot (undershoot) of the queue length (assigned bandwidth) \( u_i \) is \( \tau_i^T \geq TCA \). This result can be easily derived by transforming back to time domain Eqs. (3) and (4), where \( D_i(z) = d_0 \cdot \frac{1}{z-1} \).

A. TXOP assignment and channel saturation

We have seen in Sec. II-A that every CAP HC allocates TXOPs to mobile stations in order to meet the QoS constraints. This sub-section shows how to transform bandwidth assignments \( u_i \) into \( TXOP_i \). In particular, if the \( i^{th} \) queue is drained at data rate \( C_i \), the following relation holds:

\[
TXOP_i(k) = \frac{u_i(k) \cdot TCA}{C_i} + O \quad (8)
\]

where \( TXOP_i(k) \) is the TXOP assigned to the \( i^{th} \) queue during the \( k^{th} \) CAP and \( O \) is the overhead due to ACK packets, SIFS and PIFS time intervals (see Fig. 1). \( O \) is estimated by the bandwidth allocation algorithm by assuming that all MSDUs have the same nominal size, which is specified in the TSPEC of each traffic stream.

1) Channel saturation: The proposed FBSD algorithm is based on the implicit assumption that the sum of the TXOPs assigned to each queue is smaller than the maximum CAP duration, which is the \( dot11CAPLimit \). When the latter condition is violated, it is necessary to reallocate the TXOPs to avoid exceeding the CAP limit. This task is performed by decreasing each computed \( TXOP_i(k) \) by an amount \( \Delta TXOP_i(k) \). In particular, the generic amount \( \Delta TXOP_i(k) \) is computed as a function of the total amount \( \Delta = \sum_{i=1}^{M} [TXOP_i(k) - TXOP_i(k_0)] \) as follows:

\[
\Delta TXOP_i(k) = \frac{TXOP_i(k_0)C_i}{\sum_{i=1}^{M} [TXOP_i(k_0)C_i]} \Delta. \quad (9)
\]

Notice that Eq. (9) provides a \( \Delta TXOP_i(k) \) that is proportional to \( TXOP_i(k_0)C_i \), thus avoiding to penalize too much multimedia sources transmitting at low rates.

IV. PERFORMANCE EVALUATION

To test the effectiveness of the proposed control scheme, we have implemented the FBDS algorithm using the ns-2 simulator [9] and we have run computer simulations involving audio, video and FTP data transfers. We have considered a WLAN network shared by a mix of audio flows encoded with the G.729 standard [10], video flows encoded with the MPEG-4 [11] or the H.263 standards [12], and FTP best effort flows. Main characteristics of the considered multimedia flows are summarized in Table I. In the ns-2 implementation the

<table>
<thead>
<tr>
<th>Type of flow</th>
<th>Nominal (Maximum) MSDU Size</th>
<th>Mean Data Rate</th>
<th>Target Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPEG-4 HQ</td>
<td>1536 (2304) byte</td>
<td>770 kbps</td>
<td>40 ms</td>
</tr>
<tr>
<td>H.263 VBR</td>
<td>1536 (2304) byte</td>
<td>450 kbps</td>
<td>40 ms</td>
</tr>
<tr>
<td>H.263 256kbps</td>
<td>1536 (2304) byte</td>
<td>256 kbps</td>
<td>40 ms</td>
</tr>
<tr>
<td>G.729 VAD</td>
<td>60 (60) byte</td>
<td>8.4 kbps</td>
<td>30 ms</td>
</tr>
</tbody>
</table>

TABLE I

MAIN FEATURES OF THE CONSIDERED MULTIMEDIA FLOWS.

\( TCA \) is expressed in Time Units (TU), which in the 802.11 standard [3] are equal to 1024µs. We assume a \( TCA \) of 29 TU when G.729 flows are present otherwise the \( TCA \) is 39 TU. The proportional gain \( k_i \) is set equal to \( 1/\tau_i^T \).

In order to investigate the behavior of considered algorithm on each type of multimedia flow, we first consider a simple scenario where only MPEG and best effort FTP flows access the WLAN. Then, we consider more complex scenarios with heterogeneous coexisting flows.

A. A Simple Scenario

In this simple scenario the FBDS is compared with respect to the simple scheduler and the basic DCF access method. We have considered seven MPEG-4 video flows (i.e., the “Jurassic Park” trace taken from [13]). The 7 MPEG-4 flows share the radio medium of a 802.11b WLAN with seven best-effort FTP flows, the data rate has been assumed equal to 11Mbps for all the mobile stations. During Contention Periods the DCF is used as MAC protocol. Fig. 3 shows
the CDFs of the 7 video flows. The FBDS guarantees a delay less than $\tau_i^T + T_{CA} = 80\text{ms}$ to all the packets. On the other hand, when the simple scheduler is used, only the 87% of the packets experience a delay less than 80 ms. This result can be explained by noticing that while the FBDS dynamically allocates the WLAN medium using a closed loop control algorithm, the standard algorithm just provides a Constant Bit Rate service, which cannot match the bandwidth requirements of bursty media flows. In order to provide a further insight into the behavior of the considered bandwidth allocation algorithms, we have measured the average fraction of $T_{CA}$ allocated by the HC, which is 55.8% in the case of the proposed algorithm and 74.6% for the simple scheduler. This highlights that the simple scheduler is not able to provide an efficient use of the WLAN medium because it always allocates a constant TXOP to the flows also when they do not require it. Moreover, even if the simple scheduler allocates on average a fraction of $T_{CA}$ larger than that allocated by FBDS, it fails to respect the delay bound of 80 ms. Finally, it is worth noticing that the basic DCF access method cannot support multimedia delivering since 80% of the packets experience a delay larger than 140 ms.

![CDFs of the one-way packet delay of seven MPEG multimedia flows sharing the WLAN medium with seven best effort FTP flows.](image)

#### B. A complex scenario with many heterogeneous flows

This subsection compares the FBDS algorithm, the simple scheduler, the EDCA, and the DCF access schemes in a broad set of network scenarios. In particular, we have compared the considered access schemes and scheduling algorithms with different network loads. We consider an 802.11a WLAN shared by a traffic mix whith the number of flows expressed as a function of the load parameter $\alpha$; in particular, we consider $\alpha$ MPEG-4 flows, $\alpha$ H.263 VBR flows, $3\alpha$ G.729 flows, and $\alpha$ FTP flows. The data rate has been assumed equal to 54Mbps for all the mobile stations. The load parameter has been varied in order to investigate the effect of different network loads on the performance of the considered allocation algorithms. In this case, we have plotted the average delays experienced by the flows of each traffic class and the respective standard deviation. Before we illustrate simulation results, it is worth pointing out that in accordance with [14], the EDCA has been configured to give G.729 voice flows a higher priority than to video MPEG-4 and H.263 flows.

Figs. 4 and 5 show average delays and standard deviations obtained for the G.729 audio flows. The EDCA provides the best performance due to the higher priority given to G.729 voice flows. The DCF produces highest average delays and standard deviations. The FBDS algorithm provides a better performance than the simple scheduler. An important point is that the FBDS algorithm is not sensitive to the MAC algorithm used during the CFs, whereas the simple scheduler provides better performance when used with the EDCA method. The reason is that the FBDS is closed loop, so that if during the CP more data are transmitted with the EDCA or DCF access method, then during the CAP a smaller amount of data will be transmitted. On the opposite, the simple scheduler is a static algorithm that transmits always the same amount of data during the CAP. Figs. 6 and 7 show average delays and standard deviations obtained for the H.263 flows. In this case the FBDS algorithm provides the best performance whereas the EDCA algorithm behaves worst. This is due to the fact that video flows are treated with a lower priority than audio flows. As a consequence, even the basic DCF access method provides results better than the EDCA. This clearly shows that tuning the EDCA parameters is a key issue to be addressed in order to provide statistical service differentiation and that EDCA can starve flows with a lower priority (see also [15]). Analogous considerations can be derived by looking at Figs. 8 and 9 where average values and standard deviations of the one-way delays.
packet delay of the MPEG-4 flows are reported. It is worth noting that for all video/voice flows, FBDS guarantees the required delay bounds for all flow types and for a broader set of network load with respect to other schemes. As discussed above, only EDCA provides better results for voice flows but at the price of penalizing video flows.

V. CONCLUSION

In this paper, a novel Feedback Based Dynamic Scheduler has been designed using feedback control theory. The proposed FBDS algorithm has been implemented in the ns-2 simulator and a comparison with respect to the simple scheduler proposed by the IEEE 802.11e working group, the EDCA and the DCF access methods has been developed, in a wide range of traffic and network load conditions. Results have shown that the FBDS allows a more cautious usage of the WLAN bandwidth with respect to the others schemes, giving better results also in the presence of high network load.

REFERENCES