Providing Quality of Service Guarantees in Wireless LANs compliant to 802.11e

Thanasis Korakis and Leandros Tassiulas

Computer Engineering and Telecommunications Department
University of Thessaly
Volos, Greece
korakis, leandros@inf.uth.gr

Abstract - Quality Quality of Service (QoS) support is a key attribute of a broadband communication system, missing in 802.11 based wireless LAN’s. Recognizing the need, Task Group e (TGe) is working towards 802.11e, an evolution of the standard that provides the means for defining desirable QoS requirements for the traffic streams as well as a framework for scheduling schemes to achieve them. In this paper we propose a scheduling scheme for QoS guarantees provisioning that exploits and is fully compliant with the specifications of 802.11e. The scheduler operates at the access point and relies on the measurement and control parameters available in the fields of the 802.11e header to determine access during the evolution of PCF called HCF. Our scheme takes channel conditions under consideration in making these decisions, since information about the channel state can be conveyed now with the recent modifications adopted by TGe. Evaluation of our algorithm through simulation study shows significant performance improvement compared to earlier schemes that couldn’t take channel conditions under consideration in making scheduling decisions.

INTRODUCTION

As Internet becomes more and more popular, there is a clear tendency for the replacement of the final wired link before the user, with its wireless equivalent. This tendency becomes a global demand, as the wireless access aggregates many advantages, with the most significant of them the mobility of the user. Until now, this replacement was not easy, as the wireless line was too slow compared with its wired equivalent and the wireless equipments was expensive enough. As technology evolves, wireless LANs upgrade their speeds while speed values decay, constituting in this way a very attractive solution as the final link component of the Internet.

This rapid upgrade of the wireless link rates, suggestively we mention that 802.11a provide physical rates up to 54 Mbps [1], as well as the wide spread of wireless LANs, create a growing need for supporting Quality of Service (QoS) from such systems. The traditional best effort services of the wireless LANs must be replaced with more sophisticated services that provide guarantees in a wide array of Quality of Service attributes such as bandwidth, delay, jitter and packet delivery. Different kinds of information as voice, data, image and video have different requirements in the just mentioned attributes. Wireless LANs must support services that handle these kinds of information, with different requirements and different priorities, giving guarantees for the Quality of Service, while resolving the fairness issues that arise between wireless users. The arrangement of these problems is not trivial and constitutes a serious research challenge.

802.11 [2] is the most popular wireless LAN standard that defines the functionality of the PHY and the MAC layers. The MAC protocol of the 802.11 provides two access modes: Distributed Coordination Function (DCF) and Point Coordination Function (PCF). DCF is the fundamental method that supports access in ad-hoc networks as well as access during the contention period (CP) of infrastructure networks (where the Access Point - AP does not participate). PCF is the access method used during the Contention Free Period (CFP) of an infrastructure network called Basic Service Set (BSS) and provides a notion of Quality of Service (QoS) at the wireless stations, giving the opportunity to AP to poll them cyclically.

It is known that PCF lacks on the provision of QoS, as the scheduling model is quite static and there is no mechanism that can provide predefined QoS [3]. For that reason 802.11’s Task Group e (TGe) has been defined. The TGe is working to enhance the current 802.11 MAC to expand support for applications with QoS requirements [4,5]. TGe has added several features in the MAC of 802.11 in order to enable the support of different requirements from different classes of traffic.

The functionality of the QoS provisioning is passed to the scheduler entity [11,14,15] that is collocated with the Access Point of an infrastructure network. The responsibility of the scheduler is the accommodation of resource access and resource sharing among several users. Another goal of scheduling is to provide fair resource allocation to users. In this paper we are going to investigate the individual functional characteristics of a scheduler which operates in a 802.11e wireless environment and its target is the provision of QoS in the mobile STAs. Afterwards we are going to propose a scheduling algorithm that provides delay guarantees and complies with the previous functional characteristics of 802.11e. The scheduler takes under consideration the channel state at the wireless stations. In this way it utilizes the channel very efficiently. Our algorithm is a deadline oriented algorithm and its target is to retain the
scheduling benefits that are provided from the already proposed wireless schedulers, while it is compliant with 802.11e. Although there has been a lot of work recently on wireless scheduling and how to incorporate channel condition information in scheduling decisions, those schemes rely on assumptions that make them non-implementable in the context of 802.11e. Our algorithm is the first one that does so relying exclusively on fields available in the header of 802.11e and 802.11h.

The rest of this paper is organized as follows: in the next section we discuss the functionalities of 802.11e. In section 3 we discuss the issue of the provision of channel condition information in 802.11. In section 4 we give the functional characteristics of a scheduler that is compromised with the 802.11e framework. We propose our scheduler algorithm in subsection 4.A. Finally, in section 5 we evaluate the performance of our algorithm through simulation studies.

I. 802.11e FRAMEWORK

In this section we are going to give a brief description of the new features that have been added to the MAC protocol of 802.11 from the TGe, in order to support QoS provision. Due to the lack of space we assume that the reader is familiar with the 802.11 standard and we use some of the functionalities of 802.11 to describe the new concepts.

There are some changes in the abbreviations that are used for some concepts in 802.11e to demonstrate the added feature of QoS support. So now, a STA that operates under 802.11e is called QSTA (which means a STA that supports QoS). A BSS (Base Service Set) that operates under 802.11e is called QBSS (QoS support BSS). The data frame that supports 802.11e is called QoS data frame.

A. Provision of QoS

802.11e defines two kinds of QoS support for the traffic of a QBSS:

Prioritized QoS: In prioritized QoS the main issue in the definition of the QoS is the priority of a data frame in relation to another. Eight Traffic Categories (TC) are defined. A Traffic Category is a set of otherwise unrelated data frames to be handled by the MAC with a prioritized QoS. This prioritized QoS is indicated by the Traffic Category Identifier (TCID) for this traffic category through predefined priority mapping. The TCID value is in the range of 0-7, inclusive. The bigger the TCID is, the bigger the relative priority.

Parameterized QoS: In parameterized QoS, data frames with common traffic characteristics are grouped in classes of traffic called Traffic Streams (TS). The provision of the QoS is done by the definition of traffic characteristics (nominal MSDU size, mean data rate, delay bound etc.) of the traffic streams that are used in a QBSS. A traffic stream is a uni-directional stream of data frames to be delivered with a parameterized QoS that is indicated by a Traffic Stream Identifier (TSID). The traffic stream identifier (TSID) is an identifier used by higher-layer entities to indicate to the MAC, the traffic stream the frames belong to and the QoS requirements in the transmission of them. The TSID value is in the range of 8-15, inclusive.

B. Traffic Streams

Generation of a Traffic Stream: The generation of a traffic stream is determined by the higher layers, upon a request of an application. This request is given to the Station Management Entity (SME) of the QSTA. Upon this request, the SME enable the MAC to generate a new Traffic Stream with specific characteristics. These characteristics are given to MAC by the SME and their definition is called Traffic Specification.

Traffic Specification (TSPEC) of a TS: Is the definition of the characteristics of this TS, i.e. the parameters that define the kind of traffic that will be generated and exchanged via this TS. These parameters can be classified into two classes. The first class is a class of parameters that describes general characteristics for the TS. It includes the source and destination address, the direction of the stream, the kind of acknowledgments that are needed by the MSDUs of this stream (a new feature that supports the alternatives: no ack, normal ack, delay ack), and the kind of the TS that it describes its traffic pattern (periodic or not).

The second class of parameters is concerned with the frame characteristics for this TS and the requirements for their transmission. It includes the interarrival interval, the nominal MSDU size, the minimum data rate, the mean data rate, the maximum burst size, the delay bound, etc. The last, is a parameter that specifies the maximum amount of time allowed to wait an MPDU into the queues of the source QSTA, before its transmission to the destination and will play a crucial role in scheduler’s behavior.

C. Modification in the access functions

802.11 proposes the use of a new access function called Hybrid Coordination Function (HCF) [16] as well as the modification of the existing DCF to an enhanced DCF (EDCF), in order to provide QoS support. It defines as well, a new concept called Transmit Opportunity (TXOP).

1) Transmit Opportunity (TXOP)

A new functionality in 802.11e is the Transmit Opportunity (TXOP). The TXOP is an interval of time in Contention Period (CP) or in Contention Free Period (CFP), defined by a starting time and a maximum duration. During this interval a particular QSTA has the right to initiate transmissions onto the wireless medium. Within the limits of each TXOP, decisions regarding what to transmit are made locally by the particular QSTA. So when a QSTA obtains the right to transmit, instead of transmitting a single frame acknowledged by an Ack, it can transmit more frames, one after the other,
to the same or to different receivers, until the end of TXOP. Transmissions into a TXOP are separated by SIFS.

2) Hybrid Coordination Function-HCF

As we have mentioned above, 802.11e proposes a new function called Hybrid Coordination Function (HCF). HCF is an additional access method that is used to provide QoS to the QSTAs of an infrastructure QoS BSS (QBSS). HCF can coexist with PCF and DCF and its main function is to provide Transmission Opportunities (TXOPs) to QSTAs, according to their traffic needs. Although HCF can coexist with PCF the use of both is not efficient, as HCF provides more features that PCF and so the first can replace the second. The name of HCF reflects its ability to function in both Contention Free Period (CFP) and Contention Period (CP).

The HCF uses a Point Coordinator (PC), called Hybrid Coordinator (HC). The HC, which by default is collocated with the Access Point of the QBSS, uses the Point Coordinator’s (PC) higher priority of access to the wireless medium (hear the medium for PIFS instead of DIFS) to allocate TXOPs to QSTAs by an enhanced polling mechanism. TXOPs may be allocated at appropriate times to meet predefined service rate and delay requirements of particular traffic flows.

The HC’s QBSS-wide knowledge of the amounts of queued traffic belonging to different traffic categories with different priorities or to different traffic streams, is necessary for the right allocation of the TXOPs. This knowledge is passed through the QSTAs to the HC by the following mechanism: The HC is informed for the amount of queued traffic belonging to a specific category by piggybacking this information in the QoS data frames transmitted by QSTAs to HC. Every QoS data frame has a new field named **QoS Control field** as it is depicted in Fig. 1. In this field the QoS frame indicates its Traffic Category (0-7) or Traffic Stream (8-15) and the queue size (in terms of number of bytes) in the QSTA for this category or stream. A QSTA who needs to request a new TXOP can transmit a Null QoS data frame, just to indicate to HC this need.

The HCF protects each TXOP using the Virtual Carrier Sense mechanism, by setting the NAV in other QSTAs equal to the TXOP duration, rather than silencing all QSTAs in the QBSS for the whole CFP. In this way HCF generates small Contention Free Bursts (CFBs) that take place in TXOPs into CP or CFP.

HC uses a new QoS-poll frame where it indicates except the address of the QSTA that will have the right to use the intending TXOP, the duration of the TXOP. As we have said, the QSTA has the authority to decide for the frames that it will transmit during this time period (Fig. 2). Other QSTAs use this duration information to set their NAV counter.

II. 802.11 MAC AND CHANNEL CONDITION

As many studies describe, the error characteristics of the wireless medium differ significantly from that of the wired medium [12,13]. Packet losses in the wire-line medium are very rare and random. On the other hand, the errors on the wireless medium are bursty and the wireless channel is distinct and time-varying for each wireless user. That means that every user experiences different interference depending on its location, its distance from the transmitter (in our situation the AP), its environment and its mobility.

It is fundamental for an AP to know the channel quality that is experienced by every one of the WSTAs (Wireless Stations) which participate in its BSS (Base Service Set), as this will influence significantly the decisions the AP must make to achieve efficient and reliable communication with the WSTAs.

Among the decisions mentioned above, the most important ones are namely, the decision of the fragmentation threshold and the decision of the transmission rate of every frame. Both of these procedures are dynamic and are related to the channel quality between transmitter and receiver. Additionally such kind of information can have a fundamental role to the decisions of a wireless scheduler that is collocated with the AP. More specifically, if the scheduler has information about the channel state in every wireless STA it communicates, it will serve only the STAs that have a “good” channel, even though some of the STAs that experience a “bad” channel might be

<table>
<thead>
<tr>
<th>Octets: 2</th>
<th>6</th>
<th>6</th>
<th>6</th>
<th>2</th>
<th>6 or 0</th>
<th>2</th>
<th>0-2312</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame Ctrl</td>
<td>Dur / ID</td>
<td>Address 1</td>
<td>Address 2</td>
<td>Address 3</td>
<td>Seq Ctrl</td>
<td>Address 4</td>
<td>QoS Control</td>
<td>Frame Body</td>
</tr>
</tbody>
</table>

Figure 1: The QoS MAC header of a data frame

Figure 2: Example of a typical polling scenario and a TXOP scenario.
scheduled due to their QoS requirements. By “good” and “bad” channel we mean the channel that provides reliable and unreliable communication respectively. In this way the channel is efficiently utilized and the overall throughput is increased.

As it is known, 802.11 provides no information to the MAC layer about the channel quality. There is no mechanism that exchanges information between two WSTA (or an AP and a WSTA) concerning the channel conditions in the area close to each one. So AP’s MAC is unaware of the channel condition that every WSTA experiences. A wireless STA can gather only indirectly this kind of information, by the packet loss rate that it experiences.

802.11h is an ongoing standard of the 802.11 comity and its target is the provision of a framework for the development of Dynamic Frequency Selection (DFS) and Transmission Power Control (TPC). One of the main functionalities of 802.11h is the enhancement of the MAC protocol in order to exchange information about channel’s quality between two STAs. Following this framework, a STA, sending a TPC Request Frame can request another STA to report current transmit power and Link Margin information using a TPC Report Frame. The Link Margin field contains the estimated link margin at the time and for the rate at which the TPC Request frame was received. The link margin is defined as the ratio of the received signal power to the minimum desired by the STA receiving the TPC request. The STA that sends the report must incorporate the rate information and channel conditions, including interference, into its computation of link margin. This margin information can be used by the first STA to realize the channel’s quality in the second STA. So in the framework of 802.11h, a STA can request and receive information about the channel’s condition in the area of a neighboring STA. This mechanism can be used by the AP in order to periodically seize information about the channel condition in the STAs that is going to communicate.

The frequency in which the AP will request this information from every QSTA depends on the overhead that is added by the exchange of the two control packets and on the frequency in which the wireless channel changes state. As the errors in the wireless channel are bursty and, the channel states are time correlated, the changes in the state are slow. On the other hand we use channel information for scheduling decision in the beginning of every scheduling interval. Hence assume that the AP requests this information once every scheduling interval. Then it can use this information in the scheduling decision as we describe later. It is worth mentioning that the AP by this mechanism and due to its limited knowledge of the high dynamic and complex wireless channel characteristics, makes only channel state estimation.

III. PROVIDING QoS IN 802.11e

Many scheduling policies for QoS provision in a wireline network have appeared in literature [6,7,8,10]. More recently, several methods that addressed the scheduling in a wireless environment have been proposed. Reference [12] proposes a channel state dependent scheduling where link quality for different users is tracked to avoid transmission during fades. Fair scheduling has been studied for delay and rate sensitive flows in [13]. They consider bursty channel errors and location dependent channel capacity and propose a modified fair fluid queuing model. Reference [14] presents an architecture for fair queuing based scheduling where the medium access layer, link layer and the network layer are optimized to achieve the desirable QoS using a multirate multipath transparent CDMA based physical layer. Finally, [15] studies the problem of scheduling of QoS requests when the capacity of the channel from the base station to the mobiles is varying with time due to fading. Among their main results has been that a simple greedy algorithm can do at least as well as the offline optimal algorithm that needs to know all future channel conditions and requests. They also show that no online algorithm can do better.

The scheduling problem in 802.11 Point Coordination Function (PCF) is naive, as the scheduler has the responsibility to poll the STAs of the BSS one after another. On the other hand, the scheduling problem in 802.11e is fairly sophisticated. The issue of providing QoS to the wireless QSTAs under this framework is not easy. There are many problems that have to be resolve as well as many decisions that have to be made. These issues are related to maintaining the traffic characteristics that have been agreed in the predefined contract (Traffic Specification) for every Traffic Stream.

An 802.11e Traffic Scheduler is located in the HC of a QBSS and its target is to schedule channel access for Traffic streams. This procedure must take place in a way that will provide the QoS specifications that have been predefined by the definition of traffic streams. Another main issue of the Scheduler is to schedule the appropriate time for the transmission of the Beacons as well as to provide time intervals for EDCF access of the medium. In other words, the Traffic determines how the medium will be shared between HCF and DCF as well as how the HCF period will be shared between the Wireless STAs for their uplink and downlink transmissions. An efficient traffic scheduler mechanism should allocate time in such a way that the agreed QoS to the established Traffic Streams is maintained and at the same time high bandwidth utilization is attained.

An admission control mechanism is an indispensable component of the scheduler functionality. This mechanism will have the responsibility to permit the establishment of a certain number of Traffic Streams which the scheduler can handle as regard their needs in terms of bandwidth. This is not itself a trivial problem
and needs a lot of discussion because of the specific nature of the wireless medium. We will assume that there is an efficient mechanism that provides connection admission control, so the traffic of the QBSS can be handled by the scheduler.

The scheduling of uplink traffic is more difficult in the wireless environment compared with this of downlink traffic [13]. This is because the scheduler that is located into the AP has only limited knowledge about the arrival of packets in uplink streams. The problem is more complicated in 802.11e as the information about the pending traffic in the QSTAs is measured in number of bytes rather than in number of packets. This fact in correlation with the variable length of the transmitting packets that is a feature of 802.11 suggests the problem of transforming of this amount of data expressed in bytes to amount of data expressed in packets. One other problem that has to be addressed by the scheduler for the uplink streams is the time of the arrival of every packet into the MAC of the QSTAs. That time is unknown, but has to be estimated in order to give to the scheduler the starting time of the time period in which the packet can remain in the QSTA’s queue before its expiration and its discard. Finally, for the right allocation of time the scheduler must estimate for the uplink streams their fragmentation threshold, the transmission rate of every intending packet and other related characteristics. For that reason in this study we are going to focus in the scheduling of the uplink.

Based on the previous facts, we are going to study the main issues that the scheduler has to take under consideration for its efficient functionality: First of all the scheduler must define the scheduling interval. That means that the designers must decide about the time instants when the scheduler is activated and calculates the medium sharing between the QSTAs for the next scheduling interval. The interval can be constant as well as variable.

One other important issue for the traffic scheduler is the reservation of a time period for the transmission of every data frame of every TS. More specifically, the scheduling must define the duration of the TXOPs that it will allocate to QSTAs. This is a complicated procedure and the scheduler must act on the following points:

- Decide the time instant in which it will generate the TXOP for a specific packet. This can be calculated as the interarrival time of the specific Traffic Stream or as a specific time instant that will be crucial for the right transmission of the frame. The idea is that the transmission of every frame must take place before the expiration of the time that it can wait into the QSTA’s queues.
- Decide the length of this TXOP. This is not easy as it has to take into account the data frame length, the transmission rates, overheads, necessary time between frames (SIFS), fragmentation and other necessary functions (i.e. the necessary Ack in a data frame). The decision becomes more difficult as some of the previous characteristics (frame length, transmission rates, fragmentation threshold etc.) are not known to the HC, so some estimations must be made.

- Provide a mechanism that will maintain history of the consumed bandwidth for every TS and decides which of them are “excessive”. If the whole requested bandwidth exceeds the available bandwidth, the scheduler must discards requests of “excessive” QSTAs, as they have consumed more bandwidth than they rate. In this way the scheduler will provide fairness among QSTAs.
- Eliminate the overhead from QoS-poll frames, making efficient polls to the QSTAs.
- Take under consideration the retransmissions and the MAC and PHY overheads. For that reason it must allocate an excess amount of allocation time for every TS, proportionally to these needs.
- Take under consideration the channel state in every wireless STA.

The previous issues are crucial for the appropriate functionality of the scheduler. A design that will not take under consideration one of these points may lead to inefficient allocation of the wireless resources to the QSTAs resulting in inability to provide the predefined QoS.

A. The Scheduling algorithm

In the previous section we summarized the functional rules that must be followed in the design of a scheduler that provides QoS in mobile STAs under a 802.11e wireless environment. As 802.11e is a new framework, there is no proposal in the literature for a scheduler that complies to it. In this section we are going to describe a scheduling algorithm that is compliant with the 802.11e framework. Our algorithm has been designed with the guidance of the previous rules and resolves the arising issues in an efficient way.

For every Traffic Stream the Scheduler should maintain two queues:

Request Queues: The Scheduler tries to schedule the transmission of the uplink and downlink streams, based on the QoS characteristics and requirements, as well as

\[\text{Delay bound of the frame (i.e. the time the frame can wait in the queue before transmission)}\]

\[\text{Generation time of the frame (i.e. the time the frame arrives to MAC)}\]

\[\text{Deadline} = \text{Generation time} + \text{Delay bound}\]

Figure 3: Definition of deadline for a frame

\[\text{Figure 3: Definition of deadline for a frame}\]
at the current bandwidth needs of streams. The traffic characteristics for each stream are made available to the HC in the definition phase of the stream (Traffic Specification). The current needs of every TS are expressed by the amount of traffic waiting in the queues of each TS.

For the downlink streams this amount of traffic is updated in real time, as this traffic is queued in the HC and then is forwarded to the QSTAs. On the other hand, the amount of traffic queued in the QSTAs is not updated in real time in HC, but is getting indirectly known to the HC by the piggybacking of this information in the QoS data frames belonging to the same stream and are sent by the QSTA to the HC. This amount of traffic updates the upstream request queues in the HC.

**Virtual Queues:** The Scheduler must have a mechanism that examines the behavior of traffic streams and controls the requests from streams that exceed their predefined characteristics. That is the role of the virtual queue. A virtual traffic generator that is collocated with the Scheduler will generate traffic with the predefined characteristics for every traffic stream. That means that the generator will generate a packet of length equal to the nominal packet size, every inter-arrival interval, following the mean rate of the specific traffic stream. These packets are queued in the virtual queue and constitute the predefined amount of traffic that must be scheduled for this traffic stream. Every time an amount of traffic from the request queue is scheduled by the scheduler, an equal amount of traffic is removed from the virtual queue. A virtual traffic generator that is collocated with the Scheduler will generate traffic with the predefined characteristics for every traffic stream. That means that the generator will generate a packet of length equal to the nominal packet size, every inter-arrival interval, following the mean rate of the specific traffic stream. These packets are queued in the virtual queue and constitute the predefined amount of traffic that must be scheduled for this traffic stream. Every time an amount of traffic from the request queue is scheduled by the scheduler, an equal amount of traffic is removed from the corresponding virtual queue. In this way, at an instance of time, the state of the virtual queue gives an indication of the predefined bandwidth that the corresponding connection has consumed.

1) **Packet oriented Scheduler**

Although the information about the amount of traffic that is located in the queues of QSTAs is in terms of bytes, the Scheduler that we have designed is a packet oriented Scheduler. That means that the queues and the reservations handle the traffic amount of every stream in terms of packets.

There is an issue of transformation of the amount of data that is handled by 802.11e into packets. As we know the amount of traffic waiting for allocation in the QSTAs is represented in terms of bytes, the Scheduler that we have designed is a packet oriented Scheduler. That means that the queues and the reservations handle the traffic amount of every stream in terms of packets.

**Packet deadlines:** The scheduler we propose is delay oriented. The main factor that determines the time instant for the allocation is the waiting time of packets in queues. As we have said above the main targets of the Scheduler is the provision of the agreed QoS requirements and the high bandwidth utilization. For that reason the scheduler tries to schedule every packet near to its deadline. As deadline we define the time the packet arrives to QSTA’s MAC plus the delay bound that has been defined in the traffic specification of the corresponding stream. As we have mentioned, the value of the delay attribute is the time that a MPDU can wait in the queue of the STA before the QSTA discards it as expired. So the time we define as deadline is the latest time in which a packet can be transmitted before it is discarded (Fig. 3).

The delay bound for the specific stream is available to the HC through the traffic specification procedure at the phase of the initialization of the traffic stream. So, for the computation of the deadline of each packet, the scheduler needs to know the time of the arrival of each packet to the MAC. Due to the location of the scheduler (in the HC), the arrival time of the packets of downlink streams are known as they arrive in the local MAC. So the scheduler assigns deadlines to these packets their real generation time plus their delay bound.

On the other hand, the scheduler must provide a mechanism to estimate the arrival time of the uplink packets. For this estimation we use a worst-case mechanism. As we have said, HC is informed for the amount of data stored in the queues of each STA through piggybacking in the data QoS. We are going to use the following notations:

\[ f^M_i \] and \[ f^M_{i+1} \] two sequential frames of a specific stream \( M \) that inform the HC for the condition of the corresponding queue in a QSTA.

\[ q^M_i \] the amount of traffic in the queue of the stream \( M \) that is described by the frame \( f^M_i \).

We assign the amount of data \( q^M_{i+1} - q^M_i \) a worst-case generation time equal to the generation time of the first frame \( f^M_i \). This is demonstrated in Fig. 4. The HC knowing the arrival time of frame \( f^M_i \) and the delay bound for the corresponding stream can estimate a maximum bound of its generation time:

\[
\text{Generation time} + \text{delay bound} + \text{transmission time} = \text{time of arrival of this frame in the HC}
\]

Where the transmission time can be considered as negligible.

After estimation of the generation time of the data as described before we transform the corresponding data to packets by the mechanism we described in the previous paragraph and we assign to these packets their estimated

![Figure 4: Worst-case estimation of the generation time of data](image)

6
deadline as their estimated generation time plus the delay bound of the corresponding stream.

2) Scheduling Priorities
One main issue that must be addressed is the priority by which the Scheduler will serve the traffic streams. There are two classes of traffic streams: Periodic and non-periodic. Periodic streams have higher priority than non-periodic streams because they have strict restrictions in their traffic characteristics and delivery of their traffic after the predefined time is useless. So we design the Scheduler in a way that it first allocates channel access time for periodic traffic streams and then for non-periodic traffic streams.

It is obvious that we can have more than one traffic stream of the same class (periodic or not). In this case the scheduler will serve the traffic streams. There are two classes of traffic streams: Periodic and non-periodic. Periodic streams have higher priority than non-periodic streams because they have strict restrictions in their traffic characteristics and delivery of their traffic after the predefined time is useless. So we design the Scheduler in a way that it first allocates channel access time for periodic traffic streams and then for non-periodic traffic streams.

3) Scheduling Procedure
In the next paragraphs we are going to describe the algorithm that is used by the scheduler we propose, to allocate the appropriate channel access time to the QSTAs of its QBSS. Our scheduler allocates time per stream but as we know 802.11e works in a way that the HC allocates TXOPs per QSTA. For that reason the scheduler makes the schedule per stream and finally transforms this schedule in a per QSTA schedule.

For the implementation of the scheduler we can use any value for the scheduling interval. In the next paragraphs we adopt a constant scheduling interval, equal to a specific number of beacon intervals (as the 802.11 proposes). It is obvious that for this scheduling interval the scheduler must allocate time for the transmissions of the QSTAs in the HCF period and must live an amount of time for the EDCF to take place.

4) The decision for the time needed for the transmission of a frame
As we have mentioned, the scheduler is packet oriented. That means that it tries to allocate the appropriate time for the transmission of every packet waiting for transmission. What we mean by the phrase “appropriate time” is the time for the transmission of the frame under the rules specified by the 802.11e. If we do not take under consideration some specific characteristics of the 802.11 we may be lead to allocation of time periods that are not adequate for the successful transmission of the specific frame.

Lets denote with \( T(f_i^M) \) the time needed for the transmission of the \( i \)th frame of stream \( M \) in the wireless medium. Then:

\[
T(f_i^M) = \text{frame length} / \text{transmission rate}
\]

The frame length for the downlink frames is exactly known. For the uplink frames is estimated as the nominal frame length for that stream (as we mentioned above).

Obviously the transmission rate for the downlink frames is known as the decision for these rates is taken locally in the HC’s MAC. For the uplink frames HC can use the transmission rate that a QSTA has used in its previous transmission. Another alternative is to estimate the intending rate for every QSTA running the same algorithm [17,18] the QSTA runs for the calculation of this value taking under consideration the channel condition in the QSTA (information that is given by the 802.11h framework).

On a way or another, the HC knows or can estimate the transmission time \( T(f_i^M) \) for every frame in the wireless medium. Unfortunately, this time is not enough if the scheduler allocates an equal period for the transmission of a specific frame. There are some special functions of 802.11e that affect the allocation time:

a) Polling
Every downlink frame must be transmitted as the HC decides. So if the scheduler wants to allocate a downlink frame it must allocate and the appropriate time that is needed depending on the transmission case (Fig. 5).
b) Ack procedure

A second functionality is the use of the Ack frame as an acknowledgment of a successful transmission. If this functionality is used by a specific stream (this is declared in the traffic specification) then the scheduler must take under consideration the time for the transmission of the Ack plus a SIFS.

c) Fragmentation

As we know 802.11 and thus 802.11e has the functionality of fragmentation. The scheduler must take under consideration this feature and allocate time for the frame examining if it is fragmented (Fig. 6) or not (Fig. 5). That depends on the fragmentation threshold of every QSTA, a value that is not known to the HC. The HC can use the fragmentation threshold that a QSTA has used in its previous transmission. Another alternative is to estimate the fragmentation threshold of every QSTA running the same algorithm the QSTA runs for the calculation of this value taking under consideration the channel condition in the QSTA (information that is given by the 802.11h framework).

5) Allocation algorithm

Now that the scheduler has made the appropriate calculations and estimations it must start allocating time for every traffic stream. The scheduling algorithm works as follows:

Step 1: Take under consideration the channel state in every QSTA.

The scheduler excludes all the Traffic Streams that belong to QSTAs that experience “Bad” channel state. The scheduler does not affect the Virtual and Request Queues of these Traffic Streams, so they retain their priority once the channel condition will change and they will experience “Good” channel state.

Step 2: Time allocation to the traffic streams.

Among the streams that experience “Good” channel state, the scheduler selects a stream to serve according to the rules we describe above. The periodic streams are served first followed by the non periodic streams. Between streams of the same class it selects the stream with the higher amount of traffic in its virtual queue normalized by the corresponding predefined mean rate.

When a stream is selected, the scheduler estimates the Appropriate Transmission Time (ATT) of the first packet in the request queue of this stream, as this is defined in the previous paragraph. Then, it tries to allocate this ATT as close to the deadline of this packet as it can. We consider two cases:

Case 1: There is sufficient space in the interval between the start of the schedule and the deadline of the specific packet for the allocation of the ATT of this packet. Then the allocations of that interval are shifted to the left starting from the right to the left, until there is approximately ATT period exactly before the deadline of the packet. This procedure is demonstrated in Fig. 7.

Case 2: There is not sufficient space in the interval between the start of the schedule and the deadline of the specific packet for the allocation of the ATT of this packet. Then the allocation can be done on the right side of the deadline, while ensuring that the introduced delay for the current frame and for the other frames that are being affected is not more than their deadline plus the jitter of the corresponding stream.

In this case the algorithm is the following: Shift to the left all the allocations before (and the allocation that is located exactly on) the deadline of the packet under consideration.

Examine if the allocation of ATT of the pending packet in the space starting exactly after the last shifted allocation violates the deadline plus jitter time for that packet.

If no, shift to the right, starting from left to right, all the allocations that are located in the right of the deadline if they do not exceed their deadline plus their jitter.

If yes, abort the procedure.

The whole mechanism is demonstrated in Fig. 8.

Step 3: Concatenating allocations:

Starting from left to right the scheduler scans the
scheduling interval, examining the stream allocations. If it finds two allocations for streams belonging to the same STA (maybe and in the same stream) and at the right side of the one in the left is sufficient space where the second allocation can “fit”, concatenates the two allocations moving the second one after the first one, producing a bigger TXOP for that STA. This will be done only if the length of the new TXOP is not bigger than the maximum TXOP length that is defined by the 802.11e. When the previous concatenation takes place, the second allocation is reduced for \(T(poll)+SIFS\) as there is no need for polling of the second packet because the TXOP has been allocated to the corresponding STA by the first polling. This procedure is depicted in Fig. 9.

Step 4: Shift all the TXOPs to the left, leaving the end of the scheduling interval for the EDCF period.

The scheduling procedure will stop if the packets from the request queues finish or if the remaining idle period in the scheduling interval reaches the minimum EDCF period.

IV. SIMULATION RESULTS

A. Simulation Model

To evaluate the performance of our scheduling algorithm, we developed an event driven simulator and performed simulations for various scenarios. We measure the efficiency of our scheduler in terms of the QoS that it provides when various wireless users use various types of traffic. We measure the packet loss, the packet delay and the bandwidth allocation. The packet delay is defined as the mean time every packet remains at the MAC layer from its generation time until its transmission. Finally, the bandwidth allocation is the bandwidth that is provided by the scheduler in every Traffic Stream.

As there is no other proposed algorithm for scheduling in 802.11e, we compare our algorithm with a simple scheduler that makes static scheduling. For the rest of the paper we will refer to this scheduler as Static Scheduler. The Static Scheduler gives, in every scheduling interval, static TXOPs to every QSTA, proportional to its mean rate needs. The super frame generated by these TXOPs remains constant for the whole simulation time. The packets are transmitted into these allocation intervals in a FIFO way.

As we are going to see, our algorithm handles more efficiently the traffic streams when compared with the static scheduler, due to its dynamic feature to adapt the TXOPs to the dynamic needs of the QSTAs. At the same time, it provides in an efficient way the required QoS while serving as many traffic streams as possible.

We have used periodic as well as non-periodic traffic streams. The arrivals of packets at the periodic streams have a constant rate. In non-periodic streams the packet arrival is a Poisson process. We have used two classes of streams. The traffic characteristics of every class are depicted in Table 1. The channel rate that is dedicated to the allocations is 2Mbps. Each simulation is run for 200 seconds with a warm up period of 50 seconds.

B. Channel model

Many studies [11,12,13] have proposed the use of finite state Markovian models to characterize the error behavior of wireless links. In [12] the authors propose the use of a two-state Markov model (Good - Bad) to model the wireless channel. We have adopted a similar model in our simulations. At any point of time, we model the channel as being in one of the possible states, “Good” or “Bad”. We assume that the packet loss probability when the channel is in good state is much smaller compared with this of the channel in bad state. Generally speaking, a packet transmission is usually successful provided the channel stays in the “good” state during the packet transmission duration. Using the information that is provided to the AP by the wireless STAs about the link margin for every STA, the AP decides if the state of the channel for every one of the STAs is “Good” or “Bad”. As the AP updates the channel information for every STA at every Scheduling interval, we assume that the channel state remains the same for a period equal to the Scheduling interval.

We denote by \(p_e\) the probability that the channel state in the next Scheduling interval will be “Bad” given that it is “Good” in the current interval, and \(p_g\) the probability that the channel state in the next Scheduling Interval will be “Good” given that it is “Bad” in the current interval. We measure the efficiency of our protocol by varying the \(p_e\) from 0 (channel without bad states) to 0.3.

As we have mentioned, the packet losses in the wireless channel are correlated, i.e., a single packet loss
would be followed by many back to back packet losses. So there is a correlation in channel error during the time.

Given that the Markov Chain described above, starts in “Bad” state, it will spend there on the average, time \( T \), given by the following formula:

\[
E[T] = \frac{1}{p_g}
\]

To demonstrate this channel error correlation in time, we keep constant the \( p_g \) and equal to 0.33. By this value, when a STA experiences bad channel, it will remain in this condition for a mean time period \( T \) equal to \( 3 \times \text{Scheduling interval} \).

C. Simulation Results

In order to compare the efficiency of our algorithm with that of the Static Scheduler, defined in the previous section, we run one experiment for every scheduler. In every experiment we vary the number of QSTA from 1 to 30. We use one non-periodic stream for every QSTA. In the first experiment the time allocation is made by the Static Scheduler, while in the second experiment by our Scheduler (we call it Dynamic Scheduler). We measure the packet loss as well as the delay the packets experience.

As we can see in Fig. 10 the packet loss in both experiments increases proportionally to the LAN’s traffic load. However, for higher traffic, our algorithm attains lower packet loss and the gain increases with the number of connections (about 60% in high load). The reason is that our scheduler taking advantage of its dynamic nature and by predicting the deadlines of the packets, makes more efficient allocations, meeting the exact needs of every QSTA in every scheduling interval. On the other hand the Static Scheduler can not handle in an efficient way the packets of the non-periodic streams as the traffic increases, because it can not predict the non-periodic generation times of the packets, acting as it has to handle periodic streams.

What is worth to mention is that in conditions with relatively low load (12 QSTAs) there is a packet loss that is not negligible in both experiments (Our scheduler has lower value compared with the static). We are going to explain the behavior of our algorithm in this case: Our scheduler acts like this due to the estimation of generation time for the packets. As we have said, in non-periodic streams there is an exponential generation of packets. For this kind of traffic in the uplink the scheduler (that is located in the AP) has no direct information about the exact generation time of every packet, therefore it makes a worst case estimation as it is described earlier. As a result, when there is enough traffic, some packets are dropped as the previous estimation forces the simulator to allocate time for each one of these packets before their deadline, allocating periods of time where packets with tighter deadlines could be allocated. The latter packets are dropped from the QSTAs queues, as they can not be allocated before their worst case estimated deadline.

In Fig. 11 we can see the delay that the packets experienced in the previous scenario. Note that the delay increases proportional to the traffic load in both cases and there is a slight increase in the delay of our algorithm compared to that of the static scheduler. This is something normal due to the fact that our algorithm allocates time for every packet close to its deadline. That means that the intention of our algorithm is the transmission of every packet just before its expiration, focusing in this way to the efficient use of the medium. So as every packet is allocated close to its deadline it experiences more delay compared to the case where every packet is allocated in a FIFO way if there is enough space in the scheduling interval. However our scheduler, following this philosophy, is able to schedule more packets before its deadline as compared with the static, as we noted in Fig. 10.
One reasonable question that is generated looking at the Fig. 11 is the following: Why there is an increase of the packet delay in the case of our scheduler, as the traffic increases, while the scheduler schedules every packet near its deadline? (and so someone would wait to see a constant delay, quite lower than the MAC delay).

The key in the answer of this question is Step 4 of our algorithm. During this step the scheduler “packs” the allocations for every QSTA near the beginning of the scheduling interval if there were free space, forcing the packets to be transmitted before its deadline. As the traffic increases, more and more allocations will take place in every scheduling interval resulting in less and less packing in the left, leaving every packet to be transmitted near its initial allocation i.e. near its deadline.

In order to see the efficiency of our scheduler with regard to the way that it handles the quality of service requirements of different traffic streams, we made some simulations keeping the channel perfect i.e. channel in “Good” state. In this way the scheduler does not take into account the channel condition in every QSTA (as it is always “Good”) and therefore, it does not exclude any traffic stream from the scheduling procedure during Step 1. In these simulations we used traffic streams of class 1.

The next scenario measures the effect of the estimation of the generation time in the performance of the algorithm. We run two experiments. In every experiment we use one non-periodic stream for every QSTA and we vary the number of QSTAs from 1 to 30. In the first experiment we use the estimation of the generation time of the packets for the allocation of the scheduling time while in the second experiment we use the real generation times. We have to mention that this information is not available to the scheduler in a real environment. We give this information to the scheduler in our simulations only for the study of the effect of the estimation of the generation times in the scheduling procedure. The results of the two experiments in terms of packet loss are shown in Fig. 12.

As we can see from Fig. 12, the packet loss is higher in the case of the estimated generation times. This is something expected as the scheduler knowing the real generation times of packets it does exactly the right allocation for every packet. On the other hand, using the worst case estimation, the allocations happen earlier resulting in higher packet loss as we have mentioned in the previous scenario. What is worth mentioning is that the increase of packet loss is not very high (about 20% in high load). This means that the estimation procedure has good performance and works well as a part of the whole mechanism, leading to efficient allocations.

In our third scenario we study the behavior of the scheduler when every QSTA disposes two streams, one periodical and one random. We vary the number of QSTAs from 1 to 14. The results, in terms of packet loss, are shown in Fig. 13.

As we can see from Fig. 13 the periodical streams do not loose packets. On the other hand, the packet loss of the random streams has been increased in relation with the previous experiment. This happens due to our scheduler’s property to give priority to the service of periodical streams in association with the service of non-periodical streams. As a result, periodical streams “steal” allocation time from non-periodical streams when they have packets for transmission.

In our next experiment, we study the effect of the deadline of non-periodical streams in the behavior of the scheduler. We use two non-periodical streams for every QSTA, one with strict deadline (300 ms) and one without deadline. We vary the number of QSTAs from 1 to 15. In Fig. 14 we can see the proportion of served packets for the two classes of traffic streams.

As we have realized from the simulation, the packet loss is 0 in both classes of streams. This can be explained by the results of Fig. 14. In this figure we can see that all the packets of the class with the strict deadline are served, even in the case of heavy load. On the other hand the proportion of the served packets for...
the class with no deadline is decreased as the traffic load increases. This happens because the class with the strict deadline uses allocation time of the class with no deadline every time a packet with deadline is ready to expire. As the packets of the class with no deadline never expire, the scheduler gives priority to the class with the strict deadline, leaving in the queues of QSTAs the packets of the class with no deadline. This is the explanation of the packet loss equal to 0 for the class with no deadline, as some of its packets although they are never served, they are not dropped from the queues as their deadline never expires.

In the next class of our simulations we evaluate the efficiency of our protocol when the channel is not perfect i.e. QSTAs experience also periods with “Bad” channel condition. In these conditions we vary $p_e$ and we compare the behavior of our scheduler in two situations:

**Dummy Scheduler:** When the scheduler functions without any information about the channel condition in every QSTA. In this situation the scheduler takes no action in Step 1 of the scheduling procedure.

**Clever Scheduler:** When the scheduler functions taking under consideration the channel condition in every QSTA.

In the first experiment under these conditions, we study the behavior of the scheduler in the following scenario: We use 2 streams of class 1 in every QSTA, one periodic and one non-periodic. We use 10 QSTAs and we increase the $p_e$. The results of this scenario are shown in the Fig. 15.

Fig. 15 depicts the effect of the channel consideration in the scheduling decision. The Clever Scheduler takes advantage of the knowledge of the channel condition in every QSTA and excludes from the polling procedure the QSTAs that experience “Bad” channel, serving the QSTAs with “Good” channel instead. On the other hand the Dummy Scheduler which does not know the channel condition of the QSTAs gives TXOPs in QSTAs that experience “Bad” channel condition which results in the loss of these time periods as the QSTA can not send packets due to their “Bad” channel condition. This procedure makes the Dummy Scheduler to utilize the channel very inefficiently. For that reason, as we observe in the figure, the Clever Scheduler by making a more efficient scheduling achieves to lower the packet losses as compared to the Dummy Scheduler.

In our next experiment we explore the behavior of the scheduler by measuring the bandwidth that it provides in Traffic Streams with different QoS requirements. We have used 2 non-periodic traffic streams in every QSTA, one of class 1 and one of class 2. We run the simulation for 10 QSTA with $p_e = 0.1$. The Scheduler has the ability to take under consideration the channel state in every QSTA. The packet loss that results from this simulation is 7.4% for the streams of class 1 and 6.4% for the streams of class 2. This packet loss appears due to the fact that not only the overall throughput of the network is near the channel capacity (10 STA * 64+128 kbps = 1.92 Mbps), but the channel is not perfect ($p_e=0.1$) as well. The results of the simulation in terms of bandwidth are shown in the Fig. 16.

As we can see in Fig. 16, the Scheduler gives bandwidth proportional to the needs of every stream. So
the Scheduler accomplishes an efficient allocation of packets, guaranteeing the bandwidth requirements of streams. The overall bandwidth of the streams is slightly lower than the expected because of the packet loss they experienced.

V. CONCLUSIONS

In this work we explore the new features that have been added by the task group e in the MAC of 802.11 standard for the support of QoS in wireless LANs. We study the functional characteristics of a scheduler that runs in the AP and provides predefined QoS in wireless users with different requirements into the framework of 802.11e. Based on these guidelines, we propose a scheduling algorithm that is compliant with the 802.11e.

The proposed algorithm works in an efficient manner giving to the Traffic Streams of every QSTA allocation time proportional to their needs. As it is a delay oriented algorithm, it tries to allocate time for every packet before the expiration of its deadline, as it is estimated in the AP, giving to every QSTA allocation time that is proportional to their short-term needs. It keeps the history of the consumed bandwidth for every Traffic Stream, and does not admit them if their inclusion will violate the predefined traffic characteristics. In this way it guarantees the bandwidth requirements of every stream. The scheduler takes under consideration the channel condition in the wireless STA serving each time the STAs that experience good channel and so it utilizes efficiently the wireless channel. The scheduling algorithm we propose is simple in the design and can be implemented easily.

REFERENCES


[16] Michael Fischer, “A Hybrid Coordination Function for QoS”, IEEE 802.11-00/452r2
