Priority-Oriented Adaptive Control With QoS Guarantee for Wireless LANs

Thomas D. Lagkas, Georgios I. Papadimitriou, *Senior Member, IEEE*, Petros Nicopolitidis, and Andreas S. Pomportsis

***Abstract*—In today's wireless networks, there is a great need for quality of service (QoS), because of the time-bounded voice, audio, and video traffic. A new QoS enhanced standard is being standardized by the IEEE 802.11e workgroup. It uses a contention free access mechanism called hybrid control channel access (HCCA) to guarantee QoS. However, HCCA is not efficient for all types of time-bounded traffic. This paper proposes an alternative protocol which could be adapted in hybrid coordination function (HCF). The priority-oriented adaptive control with QoS guarantee (POAC-QG) is a complete centralized channel access mechanism, it is able to guarantee QoS for all types of multimedia network applications, it enhances the parameterized traffic with priorities, and it supports time division access using slots. Furthermore, it instantly negotiates the quality levels of the traffic streams (TSs) according to their priorities, supporting multiple streams to the best quality it can achieve. POAC-QG, compared to HCCA, provides higher channel utilization, adapts better to the characteristics of the different traffic types, differentiates the TSs more efficiently using priorities, and, generally, exhibits superior performance.**

***Index Terms*—Hybrid control channel access (HCCA), IEEE 802.11e, priority-oriented adaptive control with quality of service (QoS) guarantee (POAC-QG), wireless local area network (WLAN) medium access control (MAC) protocol.**

I. Introduction

IN MODERN networks, the need to integrate data with multimedia traffic is obvious. Voice, audio, and video have to be efficiently transmitted along with the traditional data traffic. The multimedia network applications that concern real-time traffic (RT) have some special transmission demands regarding the quality of the communication. Real-time applications require quality of service (QoS) guarantee, because they are time-bounded, while slightly unreliable connections are allowed. On the other hand, data traffic does not demand low delay or jitter, but reliability is essential. Thus, today's networks should be able to meet all types of traffic requirements.

In wired networks, the available resources seem sufficient to provide QoS, but in wireless local area networks (WLANs) supporting QoS is challenging. Wireless networks are characterized of unreliable links and limited bandwidth, that is why guaranteeing QoS is a difficult issue. Medium access control (MAC) protocols play a crucial role in QoS support. The IEEE 802.11e [1] workgroup proposes the hybrid coordination function (HCF) channel access mechanism, which considers a contention based [enhanced distributed channel access (EDCA)] and a contention free protocol [hybrid control channel access (HCCA)]. HCCA requires central control and can guarantee QoS in many cases. However, it does not efficiently support variable bit rate (VBR) traffic, while the bandwidth utilization is not high. Considering that a lot of real-time applications (such as live video) produce VBR traffic, and the fact that the WLAN bandwidth is scarce, a more efficient protocol could be used.

This paper proposes the priority-oriented adaptive control with QoS guarantee (POAC-QG) protocol which is able to cooperate with EDCA under the HCF model. It belongs to the centralized reserved access protocols. It supports real-time applications, by providing delay and jitter guarantees for both constant bit rate (CBR) and VBR traffic. Priorities are used in order to differentiate the traffic streams (TSs). POAC-QG instantly negotiates the quality levels of the TSs, supporting as many TSs as possible to the best quality it can achieve. Infrastructure network topology with central control is required. It is considered that the POAC-QG scheme is implemented in the access point (AP), which is responsible for guaranteeing QoS for the real-time TSs. Data communications that are not time-bounded use EDCA. This paper assumes that stations are able to communicate directly when they are in range, however, the model where the AP acts as a packet forwarder could be also used. HCF also provides a direct link protocol (DLP) as an extra feature.

This paper is organized as follows. Section II discusses traffic categorization and examines WLAN MAC protocols presented in literature highlighting the QoS capabilities. Section III presents the IEEE 802.11e HCF MAC, focusing on the HCCA mechanism. In Section IV, the proposed POAC-QG protocol is analyzed, the respective algorithm is examined in depth, and the considered frame structure is presented. Section V presents the simulation environment and the simulation results, which prove the efficiency of POAC-QG by comparing it with HCCA. Section VI concludes this paper.

II. Access Control and QoS for Wireless Networks

MAC protocols are responsible for ensuring efficient and fair sharing of the available bandwidth. There are various

Manuscript received February 17, 2006; revised July 12, 2006 and August 4, 2006. The review of this paper was coordinated by Prof. M. Guizani.

The authors are with the Department of Informatics, Aristotle University, 54124 Thessaloniki, Greece (e-mail: tlagkas@csd.auth.gr; g.i.papadimitriou@ieee.org; petros@csd.auth.gr; apompo@csd.auth.gr).

Digital Object Identifier 10.1109/TVT.2007.897223

relative proposals in the literature for different kinds of network conditions [2]–[24]. In wireless networks, the role of the MAC protocol is crucial. The available resources are limited, so there is a great need for efficient control of the transmissions. QoS support is also strongly related with the access control mechanism. A QoS supportive MAC protocol is able to distinguish different types of traffic and treat them accordingly. Usually, traffic is prioritized and high-priority data is favored by the access control mechanism.

A. *Traffic Categorization*

Traffic can be categorized according to the transmission requirements [8]. Various ways to classify traffic have been proposed. First of all, we can distinguish between nonreal-time (such as background data) and RT (such as voice and video). Background data traffic is not time-critical. It does not require low delay or jitter, but it demands reliable packet delivery. Usually, it is considered as low-priority traffic and an acknowledgment mechanism is used to ensure reliability. On the other hand, RT mainly concerns digital voice and video transmission and is time-bounded. Low packet delay is required in order to have qualitative audio and video reproduction. Jitter must be also kept at low values, because the packet buffer size is limited and the lifetime of the packets is small. For these reasons, high jitter increases the packet drop ratio. Live voice and video transmissions are even more demanding, because they involve extra delay caused by the real-time digital encoding at the source. However, some packet losses or bit errors can be allowed, because high reliability is not essential.

Traffic is also classified according to the way packets are generated [9]. When packets of the same size are generated at constant time intervals, then traffic is characterized as CBR. Numerous real-time voice and video digital encoders, such as G.711 and MPEG-4, respectively, produce CBR traffic. The advantage of this kind of traffic generation is that transmission time intervals can be reserved at the beginning of the communication and remain unchangeable and sufficient for its whole duration. The disadvantage is that usually this type of encoders are not bandwidth optimized, although they are rather fast. VBR traffic is produced when the generated packets are not of the same size or the generation time interval is not constant. VBR traffic is common in both background data and real-time transmissions. Background data VBR traffic is usually called nrt-VBR (nonreal-time VBR), while rt-VBR (realtime VBR) traffic mainly concerns compressed voice and video transmission. VBR voice-audio and video encoders, such as MPEG Audio Layer 3 and H.261, respectively, are not particularly bandwidth demanding, but the encoding time is rather long. Furthermore, the initially reserved average bandwidth for a rt-VBR communication is usually not capable to provide sufficient QoS, because the transmission requirements change dynamically. Since bandwidth is limited, particularly in wireless networks, the use of efficient VBR digital coding techniques is necessary. For this reason, adaptive control mechanisms that can efficiently support both CBR and VBR traffic seem nowadays quite useful. A summary of this traffic classification is given in Table I.

TABLE I
CLASSIFICATION OF TRAFFIC

Traffic Type	Examples	Characteristics
CBR (Constant Bit Rate)	real-time voice-video	efficient bandwidth reservation fast digital encoding increased produced data
nrt-VBR (non-real-time Variable Bit Rate)	background data transmission	high reliability required delay-jitter tolerant
rt-VBR (real-time Variable Bit Rate)	real-time audio-video	changeable bandwidth requirements increased encoding delay compressed produced data

B. *WLAN MAC Protocols*

Providing QoS in *ad hoc* wireless networks is a difficult task. The absence of central control is the reason why QoS cannot be guaranteed. However, the use of packet priorities can partially provide QoS, thus, there are some distributed MAC protocols that favor high-priority packets. In decentralized WLANs, the level of QoS support depends on the network characteristics, such as load and number of stations. Specifically, distributed access mechanisms are contention based, thus, high load and increased number of stations cause high collision rate and low channel utilization. Under these conditions, packet delay and jitter are increased. Thus, QoS cannot be really guaranteed in *ad hoc* WLANs. The elimination yield–nonpreemptive multiple access protocol [10], [11] used in high-performance local area network, which is standardized by European Telecommunications Standards Institute, and the EDCA protocol used by IEEE 802.11e provide partial QoS for *ad hoc* WLANs.

Infrastructure wireless networks, where central control is employed, are more suitable for supporting QoS. The access control and the schedule mechanism are implemented in the AP, which is responsible for giving transmission permissions to the mobile stations. One of the centralized access methods that provide QoS involves station polling according to the previous or following packet priorities. This method does not include bandwidth reservation. The AP analyzes the feedback and decides which station should be allowed to transmit, taking into account packet priorities. These polling schemes usually ensure low collision rate and high channel utilization, and they can provide QoS but with no guarantees. The QoS supportive adaptive polling [12], [13] and the group randomly addressed polling protocols [14] belong to this class of access mechanisms.

The MAC protocols that can actually guarantee QoS in a WLAN are the reservation centralized protocols. The access mechanisms of this class give the ability to the different TSs to reserve bandwidth. According to this model, the stations send transmission requests to the AP asking for transmission intervals, usually using a contention-based scheme. The scheduling algorithm implemented in the AP decides the bandwidth distribution in the contention free period according to the stations' requests, the priorities, the available resources etc. This type of channel access method guarantees QoS by ensuring that the packet delay of a TS will not exceed an agreed maximum limit, however the values of the actual packet delay and jitter vary and depend on the specific MAC protocol. The usual drawbacks of

TABLE II
CLASSIFICATION OF QoS SUPPORTIVE MAC PROTOCOLS

Protocol Type		Examples	Characteristics
Distributed		EY-NPMA EDCA	no infrastructure required low performance poor QoS support
Centralized	Random Access	QAP GRAP	high performance not guaranteed QoS support low feedback requirements
Centralized	Reserved Access	DQRUMA MASCARA DSA++ DTDMA PRMA HCF	increased QoS guarantee not optimal channel utilization high feedback requirements

this model include the waste of bandwidth at the contention-based period, because of the high collision probability, and the inability to efficiently support all types of RT. Specifically, if the assigned transmission periods remain constant for the whole duration of the communication, then VBR traffic cannot be efficiently supported. Representative reservation centralized WLAN MAC protocols are the following: distributed-queuing request update multiple access [15], mobile access scheme based on contention and reservation for ATM [16], dynamic slot assignment [17], dynamic time division multiple access [18], and packet reservation multiple access [19]–[21]. Variants of these protocols have also been proposed in literature. The general concept of the previously mentioned protocols is focusing on the RT and the use of a simple contention-based scheme, like slotted ALOHA, for the transmission of the requests and the nonreal-time data. The hybrid solution proposed by the IEEE 802.11e workgroup is examined in the next section. This classification of the QoS supportive MAC protocols is presented in Table II.

III. IEEE 802.11e HCF MAC

IEEE 802.11 is a standard concerning the WLANs [25]. It has dominated the market providing data rates up to 54 Mb/s (802.11a/g). The employed MAC protocol does not support QoS. However, some modifications that enhance partial QoS support have been proposed [26].

A. HCF Operation

The need for QoS in the modern WLANs has led IEEE to form the 802.11e workgroup [1]. The 802.11e channel access mechanism is called HCF and it comprises a contention-based scheme (EDCA) and a contention free scheme (HCCA). HCCA is able to guarantee QoS to some degree. It operates in infrastructure mode and its role is to efficiently support real-time voice and video communications. EDCA is designed to support prioritized traffic similar to DiffServ, whereas HCCA supports parameterized traffic similar to IntServ.

The basic concept of HCF is the transmission opportunity (TXOP), that is the time interval in which a station (also called quality enhanced station in 802.11e) is allowed to transmit. In HCCA, the TXOP is decided by the AP according to the QoS request. Specifically, the hybrid coordinator (HC) is responsible for the central control and it is colocated with the AP. However, here, we never refer particularly to the HC, but generally to the AP.

The superframe of HCF is defined as the beacon interval. It is composed of alternated modes of contention period (CP) and optional contention-free period (CFP), as it can be seen in Fig. 1. EDCA operates only in CP while HCCA can operate both during CP and CFP. HCCA mode can be started by the AP several times during a CP and these periods are called controlled access periods (CAPs). The beacon transmitted by the AP at the start of every superframe contains control information, such as the maximum duration of CFP, the maximum duration of TXOP *et al.* The end of CFP is signaled by the AP using a CFP-End message. When the AP wants to initiate a CAP, it occupies the channel and uses a CF-Poll message to Grant an HCCA-TXOP to a station.

In HCCA, every TS has its own packet buffer. The traffic specification (TSPEC) is responsible for the TS management. It provides the management link between higher layer QoS protocols such as IntServ or DiffServ with the 802.11e channel access functions (HCCA or EDCA, respectively). TSPEC describes characteristics of TSs, such as the mean data rate, the MAC service data unit (MSDU) size and the maximum required service interval (RSI). Each TS first sends a QoS request to the AP containing these characteristics. The scheduling algorithm calculates first the minimum value of all the RSIs, and then chooses the highest submultiple value of the beacon interval duration as the selected service interval (SI), which is less than the minimum of all the maximum RSIs. SI is the time interval between any two successive TXOPs allocated to a station.

B. HCCA Scheduling

The simple scheduling algorithm used in HCCA calculates the TXOPs allocated to the different TSs as follows. The TXOP corresponds to the duration required to transmit all packets generated during an SI in a TS buffer. The mean number of packets (N_{ij}) generated in the TS buffer (j) for a station (i) during an SI is

$$N_{ij} = \left\lceil \frac{\overline{r}_{ij}\mathrm{SI}}{M_{ij}} \right\rceil \tag{1}$$

where $\overline{r}_{ij}$ is the application mean data rate and M_{ij} is the nominal MSDU size. The TXOP (T_{ij}) is, finally, as follows:

$$T_{ij} = \max\left(\frac{N_{ij}M_{ij}}{R} + 2\mathrm{SIFS} + T_{\mathrm{ACK}},\right.$$
$$\left.\frac{M_{\max}}{R} + 2\mathrm{SIFS} + T_{\mathrm{ACK}}\right) \tag{2}$$

where R is the transmission rate supported by the physical layer and $M_{\max}$ is the maximum MSDU size. The time interval corresponds to the overhead during a TXOP. Equation (2) guarantees that the TXOP will be long enough for the transmission of at least one packet with maximum size. The total TXOP

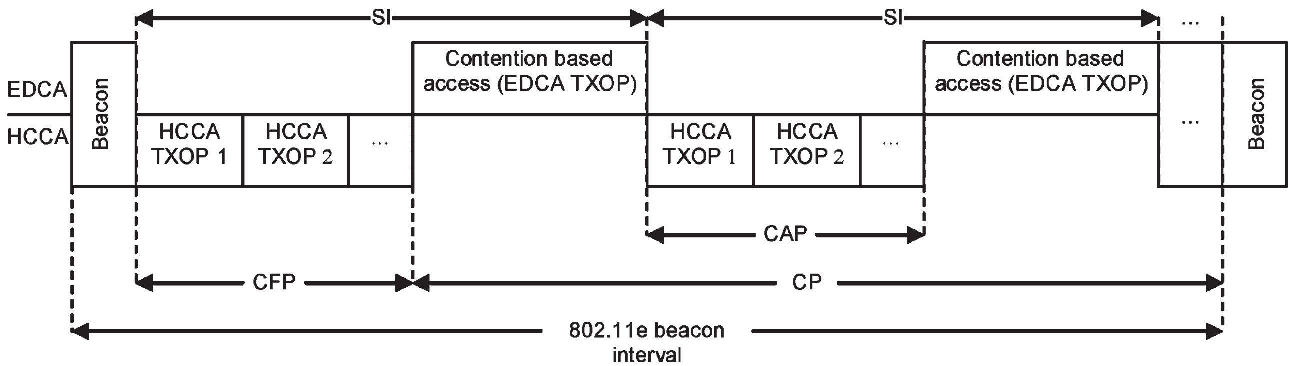

Fig. 1. 802.11e superframe.

assigned to a station is the sum of the TXOPs assigned to the different TSs of this station, that is

$$\mathrm{TXOP}_i = \sum_{j=1}^{Fi} T_{ij} \tag{3}$$

where F_i is the number of TSs in station i. The admission control algorithm checks for available bandwidth before assigning TXOP to a new TS. The fraction of total time assigned to a station i is as follows: $\mathrm{TXOP}_i/\mathrm{SI}$. If the total number of QoS stations that are assigned TXOPs is K, then the scheduler needs to check if the new request of TXOP_{K+1} will keep the fraction of time allocated for TXOPs lower than the maximum fraction of time that can be used by HCCA

$$\frac{\mathrm{TXOP}_{K+1}}{\mathrm{SI}} + \sum_{i=1}^{K} \frac{\mathrm{TXOP}i}{\mathrm{SI}} \leq \frac{T_{\mathrm{CAPLimit}}}{T_{\mathrm{Beacon}}} \tag{4}$$

where T_{CAPLimit} is the maximum duration of HCCA in a beacon interval (T_{Beacon}).

There are some drawbacks concerning the operation of HCCA. Regarding the polling mechanism, some valuable bandwidth is spent because of the polling packets sent to the stations. The use of acknowledgments is bandwidth costly, too. Since the target is to attain high throughput rather than reliability, acknowledging the RT packets seems useless. Also, all the stations have to stay constantly fully awake waiting for data packets or polls, so there is increased power consumption. Concerning the scheduling algorithm, a major drawback is the fact that the allocated TXOPs are fixed. Thus, VBR traffic cannot be supported efficiently, because possible sudden increases in the bit generation rates would cause increased delays and packet drops. Furthermore, the scheduling algorithm does not take into account prioritized TSs. It just uses the quality requirements in order to assign TXOPs. This means that the traffic is not efficiently differentiated according to the demands for QoS support. These issues and the solutions given by the proposed POAC-QG protocol are detailed in the next sections.

IV. POAC-QG Protocol

This paper proposes the POAC-QG protocol. This access mechanism operates in infrastructure WLANs and can be used in a 802.11e network in place of HCCA. The need that has led to the development of this protocol is the necessity for bandwidth saving, strict QoS with efficient VBR traffic support, and traffic-type distinction. POAC-QG is presented analytically in this section.

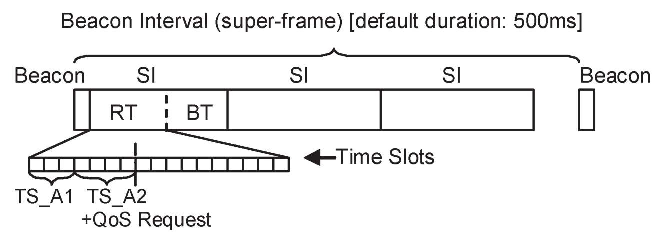

Fig. 2. POAC-QG superframe.

A. Overview of the Protocol

The superframe used in POAC-QG is separated into RT periods and background traffic (BT) periods. POAC-QG operates during the RT periods, which are contention free. During the BT periods a contention-based access mechanism can be used. The 802.11e superframe is suitable for adapting POAC-QG into it. The CFPs and CAPs correspond to the RT periods, and the CPs during which EDCA takes place correspond to the BT periods.

The POAC-QG access mechanism is not based on polling, but on a TDMA scheme. The concept is to reduce the bandwidth waste due to the polling model, keep the stations synchronized by dividing the RT period into time slots, and keep them informed of the time interval, source, and destination of the coming transmissions. Thus, a potential power saving model could be used, since stations can stay in "sleep" mode during the RT period and "wake" only to transmit or receive data. The AP uses the beacon signal to inform the stations of the assigned slots for RT transmissions and the SI duration for the current superframe. In the beginning of every SI, except from the first one in the superframe, the AP broadcasts an SI_Start message which carries the same information with the initial beacon signal. If a station fails to receive the beacon signal, it defers, until it successfully receives an SI_Start (or a new beacon signal).

When a station becomes aware of the beacon information, it ignores all subsequent SI_Start messages in the current superframe. We assume that the stations send their QoS requests for every TS during the BT periods or the last RT slots assigned to them. An overview of the superframe is shown in Fig. 2.

It is known that a multimedia application can be carried out with different quality levels (depending on the codec, the audio-video quality, etc.). The admission control negotiates instantly multiple quality levels that can be supported by the requesting TS. The corresponding algorithm tries to serve the higher priority TSs with maximum quality level, but it can lower the provided quality levels in order to allocate slots for lower priority TSs, as well. It is of course assumed that the higher the quality level is, the higher are the resource requirements (bandwidth, delay). The main purpose of the protocol is to serve as many TSs as possible, favor the higher priority TSs, and provide the higher possible quality levels. When a station sends a QoS request to ask for slots for its TSs, it includes the TSPECs of the different quality levels (traffic rate, maximum intertransmission interval, maximum and nominal packet size).

Every running TS can ask for a different number of RT slots, according to its current traffic rate and the total size of its buffered packets. Therefore, the QoS request frame that can be sent at the end of the assigned slots or during the BT periods, includes TSPECs for both running and new TSs. This way VBR traffic can be efficiently supported. The algorithm calculates first the minimum value of all the maximum intertransmission intervals required by the running and the new TSs, and then chooses the highest submultiple value of the beacon interval duration as the selected SI, which is less than the minimum of all the maximum intertransmission intervals. Then, the AP allocates slots for the running TSs according to their latest requests. The reason why the running TSs are examined first is the effort of the protocol to keep the quality of the existing communications steady. After all, a new requested voice call can wait for admission, but it is unacceptable for a running call to be suddenly terminated or experience increased delays. The rest of the bandwidth is then assigned to the new TSs, according to the admission control mechanism. The new SI duration is calculated, based on the requests of the accepted TSs and finally the time slots are assigned to the running and the new accepted TSs. In Fig. 3, an overview of the processes that take place according to POAC-QG is presented.

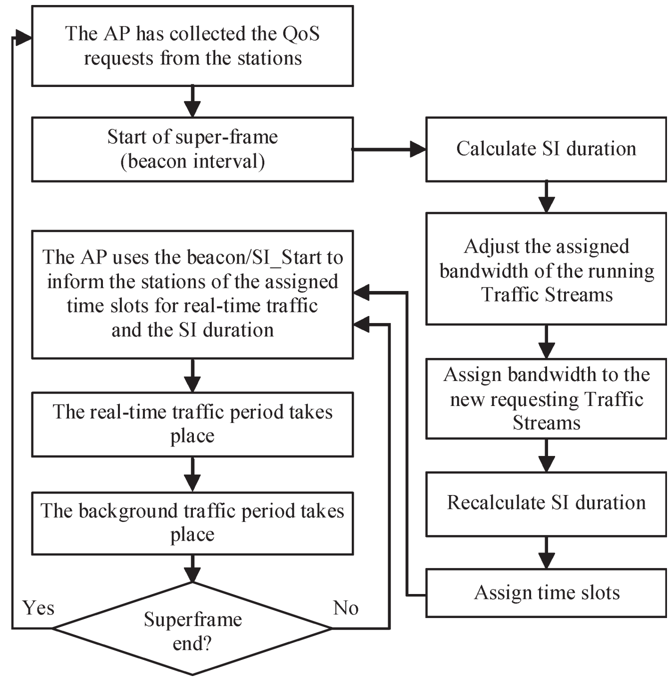

Fig. 3. POAC-QG operation overview.

TABLE III
EXAMPLE OF THE QUALITY LEVELS NEGOTIATION IN THE TSs ADMISSION PROCEDURE

	Priority_A	Priority_B	Priority_C	Priority_D
Case 1	High QL	Out	Low QL	High QL
If more than the available bandwidth is required for case 1, then go to case 2				
Case 2	Retain High QL	Out	Retain Low QL	Drop to Low QL
If more than the available bandwidth is required for case 2, then go to case 3				
Case 3	Drop to Low QL	Out	Raise to High QL	Raise to High QL
If more than the available bandwidth is required for case 3, then go to case 4				
Case 4	Retain Low QL	Out	Retain High QL	Drop to Low QL
If more than the available bandwidth is required for case 4, then go to case 5				
Case 5	Retain Low QL	Out	Drop to Low QL	Raise to High QL
If more than the available bandwidth is required for case 5, then go to case 6				
Case 6	Retain Low QL	Out	Retain Low QL	Drop to Low QL
If more than the available bandwidth is required for case 6, then go to case 7				
Case 7	Raise to High QL	Out	Retain Low QL	Out

B. Admission Control

Before assigning bandwidth to the new requesting TSs, these are sorted according to their priorities (highest priority first). The corresponding algorithm starts with the highest priority TS and checks if there is enough available bandwidth in order to serve the specific TS with maximum quality level. Otherwise, the QoS requirements of the lower quality level are checked. If neither the minimum quality level can be supported, then the TS is rejected and the next priority TS is examined. When there is no bandwidth left to serve a TS with minimum quality, then the quality levels of the previously examined higher priority TSs are lowered in order to save some bandwidth for the new TS. When the quality levels of the high-priority TSs are lowered, then we also check if it becomes possible to increase the quality of the low-priority TSs. This way, the best combination of supported quality levels is provided. An example of this process is described in Table III, where we assume two available quality levels (high QL, low QL) and four new TSs with different priorities (Priority_A is the highest, while Priority_D is the lowest). The first three TSs are already examined. Let us assume that, so far, Priority_A TS has been accepted with high QL, Priority_B TS has been rejected, Priority_C TS has been accepted with low QL, and Priority_D TS is now examined for admission. This means that we are looking for the best quality levels combination of these four TSs, which can be served using the current available bandwidth. In this example, there are seven possible cases. Each time, the algorithm checks if there is enough available bandwidth in order to serve the TSs providing the corresponding quality levels combination. If there is not, then we proceed to the next best quality levels combination (case). The final case is the rejection of the examined TS (quality level: OUT). A simple form of the code is presented in Fig. 4.

```
for i=0 To NumberOfNewTSs-1 //Each new TS is examined
 {
  GetNextTS=false //A flag to proceed to the next TS
  do //Searching for the best combination of quality levels of the TSs examined so
   { //far that requires bandwidth not more than the available bandwidth
    Calculate NewBandwidthAssigned //Calculate bandwidth assigned so far
    if NewBandwidthAssigned+BandwidthTS[i]>AvailableBandwidth
     { //If the current combination of quality levels requires more bandwidth than
       //the available, then we check the next "best" combination.
      j=i//Starting from the last examined TS,we search back for the first TS that
      do // has higher quality level than MIN, so we can lower it.
       {
        if QualityLevelTS[j]!=OUT //If a TS has been rejected, then it is not
         {                        //further considered.
          if QualityLevelTS[j]==MIN //When the examined TS is assigned the min
           QualityLevelTS[j] = MAX  //quality level, then it gets the max level to
                                    //ensure best combination and we proceed.
          Else
           QualityLevelTS[j]-- //The quality level of the specific TS is lowered
          j-- //Proceed to the next TS, that is the previously examined while it
         }   //also carries a higher priority (lower TS index -> higher TS priority)
       }while (QualityLevelTS[j+1]==MAX OR QualityLevelTS[j+1]==OUT) AND j>-1
      if j==-1 AND (QualityLevelTS[0]==MAX OR QualityLevelTS[0]==OUT)
       QualityLevelTS[i]=OUT // All combinations have been examined. The current TS
     }                      // i is rejected, because there is no way to get the
                            // bandwidth requested by any of the quality levels
    else // The quality levels of the TSs 0 to i have been decided
     {
      Assign the decided quality levels for the TSs 0 to i
      GetNextTS=true // Proceed to the next (lower priority) TS (i+1)
     }
    if QualityLevelTS[i]==OUT //If the examined TS is rejected, then the quality
     GetNextTS=true           //levels of all the previous TSs do not change and we
                              //proceed to the next (lower priority) TS (i+1)
   }while GetNextTS==false //Examine the next (lower priority) TS (i+1)
 }
```

Fig. 4. Code of the admission control algorithm.

C. Control Adapted to Requirements

The POAC-QG protocol efficiently supports VBR RT by adapting to the changing requirements of the running TSs. When the AP assigns RT slots to a station, it provides some extra slots allocated for its QoS request frame transmission. The station uses this frame to send the TSPECs both of the new requesting and the running TSs. Before sending a QoS request, the station calculates the current traffic rate of all the running TSs by counting the generated bits for a short time interval (default value is 2 s). It also includes in the QoS request the size of the corresponding packet buffer. At the start of every superframe, the AP assigns slots to the running TSs according to their new QoS requests. The rest of the RT bandwidth is then assigned to the new TSs as we have already discussed. The quality level initially provided to a TS remains static, because our aim is to have steady and reliable transmissions.

The algorithm that assigns time slots to the running TSs tries to adapt to the variable traffic rate without sudden alterations of the allocated bandwidth. When there is not enough RT bandwidth, it assigns a proportion of the requested bandwidth to each TS according to its priority. It is considered that all the generated and buffered packets of a TS can be transmitted during an SI, if the allocated bandwidth corresponds to the theoretical traffic rate

$$\text{TheoreticalTR} = \text{CurrentTR} + \text{BufferedBits}/\text{SI} \qquad (5)$$

where CurrentTR is the current traffic rate defined in the QoS request. Since we try to avert sudden and continuous alterations of the allocated bandwidth, a proportion of the requested bandwidth accession or reduction is considered to be the target. Specifically, the considered target traffic rate is

$$\text{TargetTR} = \text{PreviousTR} + \text{BW\_DifPercent} \times (\text{TheoreticalTR} - \text{PreviousTR}) \qquad (6)$$

where PreviousTR is the traffic rate corresponding to the bandwidth assigned during the previous superframe, and BW_DifPercent (default value is 0.8) is the percentage of the requested bandwidth accession or reduction which is considered to be the target. We also use a down limit for the target traffic rate related to the initial traffic rate requested, in order to avoid packet drops in cases of sharp increase of the generated packets after a long silent interval.

Obviously, when a TS requests to give back some of its assigned bandwidth because it does not need it anymore, this is done with no further consideration. An issue arises when there is not enough bandwidth to cover all the extra requests of the running TSs. For this reason, an algorithm that distributes the available bandwidth taking into account the traffic priorities has been developed. It initially calculates the percentage of the available bandwidth that each requesting TS deserves (eligible bandwidth). The available bandwidth corresponds to the slots left in the maximum RT period, after assigning to all the running TSs the slots that already occupied in the previous beacon interval and freeing the returned slots. The eligible bandwidth percentage depends on the traffic priority and the amount of extra bandwidth requested by the TS. Specifically, we use the weights W_PR (default value is 5) and W_BW (default value is 1) to control the contribution of the traffic priority and the extra bandwidth requested, respectively, to the eligible extra bandwidth. It is obviously assumed that the traffic

```
//Examine each running TS requesting extra bandwidth
for i=0 to NumberOfRequestingTSs-1
 {
  SumPR+=PriorityWeight[i]
  SumBW+=ExtraBW_Requested[i]
 }
for i=0 to NumberOfRequestingTSs-1
 {
  PerPR[i]=PriorityWeight[i]/SumPR //Normalize priority
  PerBW[i]= ExtraBW_Requested[i]/SumBW //Normalize extra bandwidth requested
  Per[i]=W_PR*PerPR[i]+W_BW*PerBW[i] //Non-normalized eligible bandwidth percentage
  if i belongs to the AP
   Per[i]=W_AP*Per[i]
  IsExtraBW_Decided[i]=false //Initialization of the flag
 }
IsAnyExtraBW_Decided=true //Initialize the loop termination flag to enter the loop
While IsAnyExtraBW_Decided==true //The loop terminates at that step that no new
 {                                //extra bandwidth is decided. This means that
                                  //all requests have been examined.
  IsAnyExtraBW_Decided=false //Initialize the flag in the loop
  SumPer=0;
  for i=0 to NumberOfRequestingTSs-1
   if IsExtraBW_Decided[i]==false
    SumPer+=Per[i]
  for i=0 to NumberOfRequestingTSs-1
   if IsExtraBW_Decided[i]==false //Normalize the eligible bandwidth percentage for
    {                              //the TSs that are not examined yet and calculate
     nPer[i]=Per[i]/SumPer         //the eligible bandwidth
     ExtraBW_Eligible[i]=AvailableBandwidth*nPer[i]
    }
  for i=0 to NumberOfRequestingTSs-1
   if IsExtraBW_Decided[i]==false               //Check all the unexamined TSs and
    if ExtraBW_Requested[i]<=ExtraBW_Eligible[i]//if the bandwidth requested is not
     {                                          //higher than the eligible bandwidth
      ExtraBW_Assigned[i]=ExtraBW_Requested[i]  //then assign the requested
      IsExtraBW_Decided[i]=true                 //bandwidth to the specific TS,
      AvailableBandwidth-=ExtraBW_Assigned[i]   //update the flag which shows that
      IsAnyExtraBW_Decided=true                 //the TS is examined and lower the
     }                                          //available bandwidth.
 }
//The TSs that are not assigned extra bandwidth while being in the loop are those
//that cannot get the whole extra bandwidth requested. So, finally, we assign
//these TSs the eligible extra bandwidth.
for i=0 to NumberOfRequestingTSs-1
 if IsExtraBW_Decided[i]==false
  ExtraBW_Assigned[i]=ExtraBW_Eligible[i]
```

Fig. 5. Code form of the dynamic bandwidth assignment mechanism.

priority is clearly the most significant factor. The equation that gives the nonnormalized eligible bandwidth percentage for the TS i is

$$\text{Per}[i] = W\text{\_PR} \times \text{PerPR}[i] + W\text{\_BW} \times \text{PerBW}[i] \quad (7)$$

where PerPR is the normalized traffic priority

$$\text{PerPR}[i] = \frac{\text{PriorityWeight}[i]}{\sum_{j=0}^{\text{NumberOfRequestingTSs}-1} \text{PriorityWeight}[j]} \quad (8)$$

and PerBW is the normalized extra bandwidth requested

$$\text{PerBW}[i] = \frac{\text{ExtraBW\_Requested}[i]}{\sum_{j=0}^{\text{NumberOfRequestingTSs}-1} \text{ExtraBW\_Requested}[j]}. \quad (9)$$

We use the term "priority weight" instead of just "priority," because the weight of a traffic priority might be considered to be different than the index of the specific priority. We assume

$$\text{PriorityWeight} = \text{Priority} + 1 \quad (10)$$

(e.g., priority: 0 → weight: 1). Since the AP is the "heart" of the WLAN and it often interconnects the WLAN with the backbone wired network, any traffic coming from the AP should be served with definitely higher priority. In order to favor the AP TSs, we use the W_AP (default value is 5) factor to calculate the nonnormalized eligible bandwidth percentage. Therefore, for every TS i transmitted by the AP it stands

$$\text{Per}[i] = W\text{\_AP} \times (W\text{\_PR} \times \text{PerPR}[i] + W\text{\_BW} \times \text{PerBW}[i]). \quad (11)$$

We finally normalize

$$\text{nPer}[i] = \frac{\text{Per}[i]}{\sum_{j=0}^{\text{NumberOfRequestingTS}-1} \text{Per}[j]}. \quad (12)$$

At each step, if the eligible bandwidth of a TS is higher than its requested bandwidth, then the latter is immediately granted to this TS. Finally, a proportion of the requested bandwidth is assigned to the TSs that cannot be fully served. The algorithm that calculates the extra bandwidth that would be assigned to every requesting TS is presented in pseudocode form in Fig. 5. An example is given in Table IV. This method of continuous and dynamic bandwidth assignment completes the support provided by POAC-QG to VBR traffic.

V. Performance Evaluation

At this point, it should be mentioned that it is not feasible to model channel access in POAC-QG based on the concept

TABLE IV
EXAMPLE OF ASSIGNING EXTRA REQUESTED BANDWIDTH TO THREE RUNNING TSs (A, B, C) ACCORDING TO THE CORRESPONDING POAC-QG ALGORITHM

Step	TS	Priority	Requested Bandwidth	Available Bandwidth	Eligible Bandwidth	Assigned Bandwidth
1	A	6	5 Mbps	10 Mbps	5.6 Mbps	5 Mbps
	B	3	3 Mbps		2.9 Mbps	-
	C	1	4 Mbps		1.5 Mbps	-
2	B	3	3 Mbps	5 Mbps	3.3 Mbps	3 Mbps
	C	1	4 Mbps		1.7 Mbps	-
3	C	1	4 Mbps	2 Mbps	2 Mbps	2 Mbps

of the classical Bianchi two-state Markov chain [27]. The access scheme of POAC-QG is deterministic, since the AP is informed of the stations' transmission needs by the QoS request frames. POAC-QG, does not actually involve any idle time during the RT slots. However, we do use a three-state Markov process to simulate the link status between each pair of stations, as it is explained later. Regarding the proposed mechanism for TS admission control and dynamic adjustment of the allocated resources, the algorithmic complexity and heuristic nature makes further theoretical analysis impossible and actually unnecessary. In related work, it can be seen that this is a common concept. In [28], the proposed SETT-EDD scheduling algorithm for HCCA is evaluated via simulation and no theoretical analysis is performed. The authors state that the typical two-state Markov chain used to model the channel does not accurately represent a WLAN with link adaptation. In [29], there is no theoretical analysis of the open-loop and closed-loop scheduling proposals for HCCA. It is stated that the authors are not particularly concerned in finding a theoretical optimal scheduler, since it could turn out to be computationally complex or lose its optimality properties due to implementation impairments. In [30], no Markov modeling is used for the analysis of the proposed FHCF scheduling scheme for HCCA. Similarly to our approach, the authors propose a formula for resource allocation based on the queue length. Lastly, in [31], a TS admission control is proposed for HCCA, employing a sequence of computations and checks which involve the TSPECs and the available resources. However, no asymptotic analysis is performed to validate the efficiency of the mechanism, instead, simulation comparison is used.

A. Simulation Environment

In the developed simulation environment, the condition of any wireless link was modeled using a finite-state machine with three states. These are the following [32], [33].

1) State G denotes that the wireless link is in a relatively "clean" condition and is characterized by a small BER, which is given by the parameter G_BER.
2) State B denotes that the wireless link is in a condition characterized by increased BER, which is given by the parameter B_BER.
3) State H denotes that the pair of communication stations is out of range (hidden stations).

We assume that the background noise is the same for all stations, and thus, the principle of reciprocity stands for the condition of any wireless link. Therefore, for any two stations A and B, the BER of the link from A to B and the BER of the link from B to A are the same. The time spent by a link in states G, B, and H is exponentially distributed, but with different average values, given by the parameters TG, TB, TH, respectively. The status of a link probabilistically changes between the three states. When a link is in state G and its status is about to change, the link transits either to state H, with probability given by the parameter $P_{\rm h}$, or to state B, with transition probability $1 - P_{\rm h}$. When a link is in state B and its status is about to change, the link transits either to state H, with probability given by the parameter $P_{\rm h}$, or to state G, with transition probability $1 - P_{\rm h}$. Finally, when a link spent its time in state H, it transits either to state G or B, with the same probability (0.5). It can be easily seen that by setting the parameter $P_{\rm h}$ to zero, a fully connected network topology can be assumed, whereas for values of $P_{\rm h}$ greater than zero, the effect of the well-known "hidden station" problem on protocol performance can be studied.

In a "clean" network, it stands for the interstation links: TG $= 3$ s, TB $= 1$ s, TH $= 0.5$ s, G_BER $= 0$, B_BER $= 0$, $P_{\rm h} = 0$. Similarly, for the AP-station links it stands: TG_AP $=$ 6 s, TB_AP $= 0.5$ s, TH_AP $= 0.25$ s, G_BER_AP $= 0$, B_BER_AP $= 0$, $P_{\rm h}$_AP $= 0$. The links among the AP and the stations are considered to be more reliable than the interstation links, because the range of the AP is usually greater than the stations' range, its emitted signal is usually stronger, and its default position is the center of the cell. In a rather not "clean" wireless environment, it stands for the intersta-tion links: TG $= 3$ s, TB $= 1$ s, TH $= 0.5$ s, G_BER $= 0$, B_BER $= 0.00001$, $P_{\rm h} = 0.05$. For the AP-station links it stands: TG_AP $= 6$ s, TB_AP $= 0.5$ s, TH_AP $= 0.25$ s, G_BER_AP $= 0$, B_BER_AP $= 0.000001$, $P_{\rm h}$_AP $= 0.01$. The BERs are assumed to be resulted after the application of the standard's predefined coding techniques.

The default values of the network parameters used in our simulation scenarios are presented here. The medium bit rate is 36 Mb/s, the signal propagation delay is 0.0005 ms corresponding to distances among the stations of 150 m, the maximum percentage of the superframe reserved for RT transmissions is 0.95, and the maximum allowed packet size is 10 KB. According to the specifications of 802.11e, we consider the following total packet sizes: "POLL" $= 34$ B, "BEACON" $= 124$ B, and "QoS_Request" $= 44$ B. The total overhead of every traffic packet is 106 B, including physical, MAC, RTP, UDP, IP, and SNAP headers.

Regarding the simulation engine, the random number generator used by our simulator is a classic multiplicative congruential random number generator with period 2^{32} provided by ANSI C. The simulation results presented in this section are produced by a statistical analysis based on the "sequential simulation" method [34]. We perform simulations in a sequential way, until the relative statistical error of the estimated mean value falls below an acceptable threshold. When the relative statistical error is low, the confidence interval is narrow, since the relative statistical error is defined as the ratio of the half-width of the given confidence interval at the point estimate. For this statistical analysis we used 95% confidence intervals. The relative statistical error threshold varies depending on the meaning

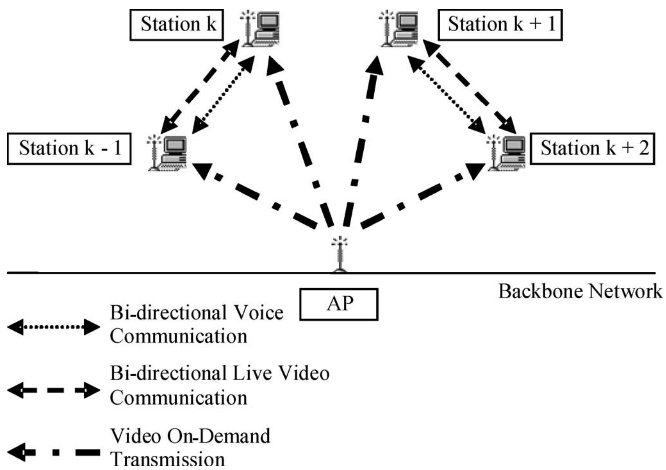

Fig. 6. Transmissions in the first simulation scenario.

TABLE V
CHARACTERISTICS OF THE TRAFFIC TYPES USED IN SIMULATION SCENARIO 1

Application	Coding	Packet Data Size (bytes)	Packet Interarrival Time (ms)	Data Bit Rate	Packet Delay Bound (ms)
Voice (Priority: 6)	G.711 (PCM)	160	20	64 Kbps (CBR)	50
Live Video (Priority: 5)	H.261 [QCIF]	Exponential [20-1024] Mean: 660	Exponential Mean: 26	~200 Kbps (VBR)	100
Video on Demand (Priority: 4)	MPEG-4 [4CIF]	800	2	3.2 Mbps (CBR)	200

of the metric and the magnitude of its value. However, this threshold was usually assumed to be lower than 2% and never exceeded 5%.

B. First Simulation Scenario

We have used two simulation scenarios to compare the performance and the general behavior of the HCCA and the POAC-QG protocols. We consider only real-time TSs, because the BT access mechanism (EDCA) is the same for the two cases. In the first scenario, we have live voice and video communications (bidirectional transmissions) between the adjacent mobile stations (station 1 communicates with station 2, station 3 communicates with station 4, and so on), and a video on demand TS transmitted by the AP to each station. In Fig. 6, we have a representation of the transmissions taking place in the first simulation scenario. Our aim is to compare the QoS provided by the two protocols, when there is just one quality level, that is there is no QoS negotiation. Therefore, in this case, the proposed QoS negotiation mechanism of POAC-QG does not affect the simulation results. The characteristics of the network traffic can be found in Table V. The simulation duration is 30 s, every communication lasts for 20 s, a new set of transmissions (voice, live video, video on demand) are generated every second, and the simulated WLAN consists of 10 mobile stations (that is 30 TSs). Also, we consider "clean" links, so we used the respective network parameters' values mentioned earlier. It should be noticed that in both scenarios, we do not drop the packets that exceed their delay bound, so as to get results from all transmissions.

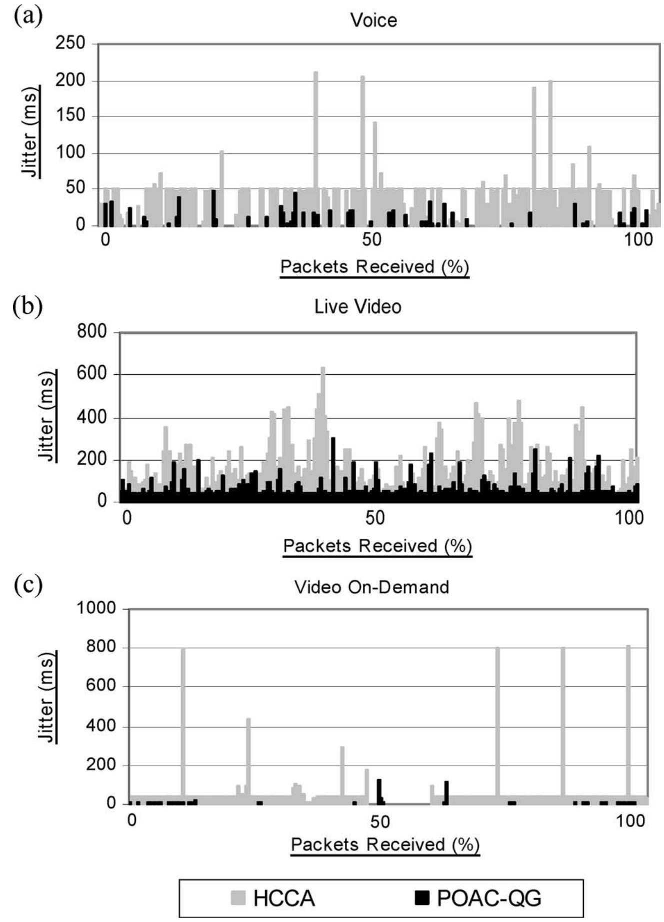

Fig. 7. Packet jitter measurements concerning (a) voice, (b) live video, and (c) video on demand traffic.

In the first simulation scenario, we get measurements of the packet jitter and the TS buffer size. These two metrics are representative of the capability of the MAC protocol to efficiently provide QoS. In Fig. 7, we have plotted the results regarding packet jitter. It is obvious that in all cases POAC-QG exhibits much lower jitter than HCCA. The jitter of the voice packets is always kept below 50 ms. The graph that concerns live video, shows that POAC-QG can efficiently support VBR traffic by providing significantly low jitter values. Furthermore, it is capable of successfully serving high bit-rate CBR TSs, like video on demand. This superior performance of POAC-QG is partially owed in its ability to adapt to the special requirements of every TS and continuously provide the bandwidth actually needed.

The buffer size of the source station was also measured when a packet was transmitted. It is important for the source to be able to transmit on time the RT packets that arrive at the TS buffer. The ideal case would be the constant counterbalance of the transmission rate and the packet generation rate. However, in a real situation, it is quite difficult to adapt the transmission rate to the packet generation rate. This is particularly true when dealing with VBR traffic, where the packet generation

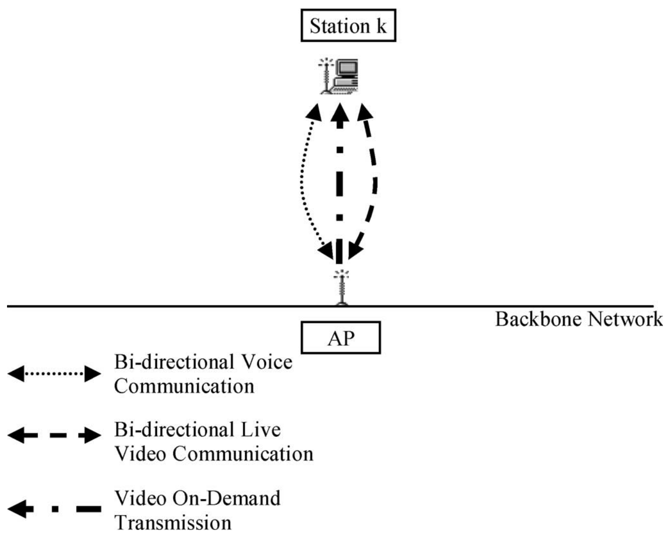

Fig. 8. Transmissions in the second simulation scenario.

rate changes continuously. In such cases, when a large number of packets suddenly arrive at the buffer, the station might be unable to transmit all packets on time, so there could be packet drops due to lack of buffer space or excess of packet lifetime. The results (which are relative to the jitter results) showed that in all cases POAC-QG manages to "unload" the buffers more efficiently than HCCA. This happens because of the proposed adaptive bandwidth assignment mechanism which continuously provides transmission rates according to the current packet generation rates. Also, the optimized access mechanism, which provides resources saving, significantly contributes to the superior performance of POAC-QG.

C. Second Simulation Scenario

In the second simulation scenario, our aim is to examine the packet delays and the QoS negotiation efficiency of POAC-QG in a rather not "clean" environment. For these reasons, we used two quality levels (MIN, MAX) and we set the network parameters to the earlier mentioned values that correspond to links of decreased reliability. The employed traffic model involves only AP-station communications. We have live voice and video communications (bidirectional transmissions) between the AP and each station, while the AP transmits a video on demand TS to each station. In Fig. 8, we have a representation of the transmissions taking place in the second simulation scenario. The traffic characteristics can be found in Table VI. We notice that voice and live video traffic support two quality levels, while video on demand traffic actually supports a single quality level. This is not a problem for the operation of POAC-QG, since it does not require that all TSs support all the provided quality levels. The simulation duration is 60 s, every communication lasts for 30 s, half of the transmission sets (voice, live video, video on demand) start at the beginning of the simulation and the other half start 30 s later, and we simulated 15 WLAN topologies consisting of 2 to 30 stations (that is 10 to 150 TSs).

In Fig. 9, we have plotted the average packet delay versus the number of the total offered TSs. In all cases (voice, live video, video on demand traffic), POAC-QG provides lower packet delays than HCCA, while the latter some times fails to provide delays lower than the maximum tolerable value. It should be noticed, that the total number of offered streams corresponds to the streams scheduled to take place during each simulation. However, some of them may not get permission to start at all due to limited available bandwidth. Also, the served TSs are assigned different quality levels with different bandwidth requirements, and not all of the accepted streams are served for the same time. For these reasons, we need a new metric in order to get a clear and fair view of the comparison of POAC-QG and HCCA.

We call this new metric "Q_Score." It depends on the priority of each served TS, its quality level, the number of served TSs, and the network's throughput. First of all, we define the factor "Q_Factor" which concerns the assigned quality level.

TABLE VI
CHARACTERISTICS OF THE TRAFFIC TYPES USED IN SIMULATION SCENARIO 2

Application	QL	Coding	Packet Data Size (bytes)	Packet Inter-arrival Time (ms)	On/Off Periods (sec)	Data Bit Rate	Packet Delay Bound (ms)
Voice (Priority: 6)	MAX	G.711 (PCM)	160	20	Expo. (mean) On: 1.5 Off: 1.8	64 Kbps (CBR)	50
	MIN	G.729_A (CS-ACELP)	20			8 Kbps (CBR)	
Live Video (Priority: 5)	MAX	H. 261 [CIF]	Expo. [40-2048] Mean: 1320	Expo. Mean: 13	Always On	~800 Kbps (VBR)	100
	MIN	H.261 [QCIF]	Expo. [20-1024] Mean: 660	Expo. Mean: 26		~200 Kbps (VBR)	
Video On-Demand (Priority: 4)	MAX	MPEG-4 [4CIF]	800	2	Always On	3.2 Mbps (CBR)	200
	MIN						

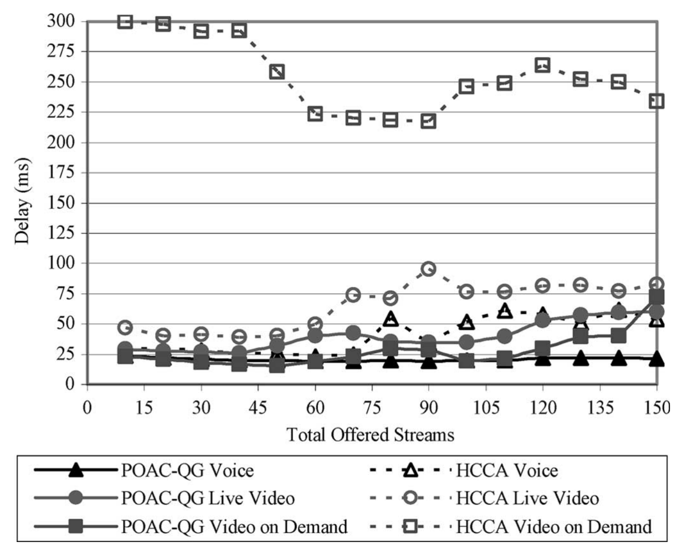

Fig. 9. Average packet delay versus the number of total offered TSs.

Q_Factor is higher when a TS is assigned the MAX quality level. However, we want to get a clearly higher score when serving two MIN TSs than one MAX TS. Since it is more important to serve multiple low-quality TSs than one with high quality, we decided to set $Q\text{\_Factor} = 1$ when the TS is assigned the MIN quality level, and $Q\text{\_Factor} = 1.1$ when it is assigned the MAX quality level. First, we calculate the score for each TS

$$\text{Stream}Q\text{\_Score} = Q\text{\_Factor} \times \text{PriorityWeight} \times \text{TimeServedRatio} \quad (13)$$

where the PriorityWeight depends on the stream's traffic priority and the TimeServedRatio is the ratio of the time interval the TS was served to the total time it was scheduled to last. At this point, it should be reminded that according to our simulation settings all TSs are scheduled to last no more than the simulation duration. Therefore, in an ideal situation, all the TSs would be completed before the simulation termination. The IdealStream$Q$_Score is the score of a MAX quality TS that is completed before the simulation termination $(\text{TimeServedRatio} = 1)$. It stands as

$$\text{IdealStream}Q\text{\_Score} = \text{Max}Q\text{\_Factor} \times \text{PriorityWeight}. \quad (14)$$

The RatioNet$Q$_Score, which concerns the total offered streams, is defined as

$$\text{RatioNet}Q\text{\_Score} = \frac{\sum_{i=1}^{\text{OfferedStreams}} \text{Stream}Q\text{\_Score}}{\sum_{i=1}^{\text{OfferedStreams}} \text{IdealStream}Q\text{\_Score}}. \quad (15)$$

Finally, we calculate each simulated network's Q_Score in relation to the score of the same network when using a different protocol. It stands as

$$Q\text{\_Score} = \text{RatioNet}Q\text{\_Score} \times \frac{\text{Throughput}}{\text{HigherThroughput}} \quad (16)$$

for the network with the lower throughput and $Q\text{\_Score} = \text{RatioNet}Q\text{\_Score}$ for the network with the higher throughput. For example, if an HCCA network has $\text{RatioNet}Q\text{\_Score} = 1$ and $\text{Throughput} = 0.6$, and the same network using POAC-QG has $\text{RatioNet}Q\text{\_Score} = 1$ and $\text{Throughput} = 0.8$, then the Q_Score for the HCCA network is 0.75 while for the POAC-QG network is 1. Thus, Q_Score as it is formed in (16), can only be used to compare the performance of two networks and not as an individual metric.

The statistical results concerning the Q_Score of 15 network topologies (two to 30 mobile stations) are depicted in Fig. 10. Obviously, POAC-QG always exhibits higher Q_Score than HCCA. This is a definite indication of the efficiency of the QoS negotiation mechanism employed by POAC-QG. In all cases, the proposed protocol ensures a better combination of MAX and MIN quality level TSs, as shown in Fig. 11. It appears that POAC-QG always serves as many TSs as possible to the best quality it can achieve.

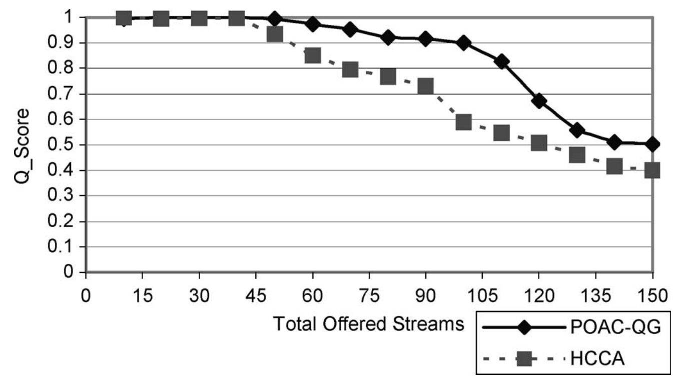

Fig. 10. Q_Score versus the number of the total offered TSs.

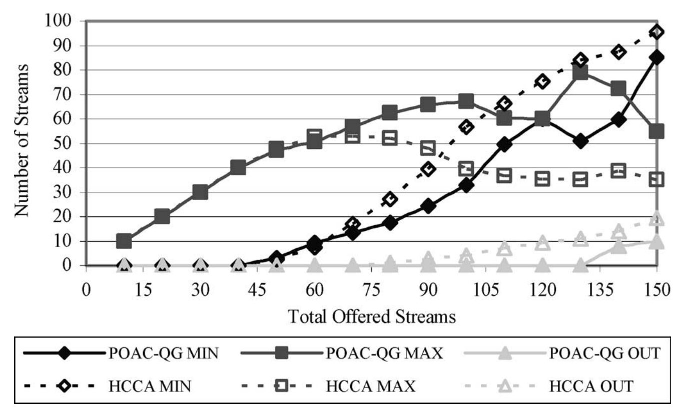

Fig. 11. Number of the TSs assigned the minimum quality level or the maximum quality level or they were rejected versus the number of the total offered TSs.

VI. Conclusion

This paper proposed the POAC-QG protocol for WLANs. It can be adapted into the HCF protocol of the IEEE 802.11e standard in place of HCCA. A TDMA scheme is adopted for the access mechanism. POAC-QG is designed to efficiently support all types of RT. It guarantees QoS both for CBR and VBR traffic, by continuously adapting to their special requirements. Since numerous network multimedia applications produce VBR traffic, it is essential to support it with high quality. HCCA, on the other hand, appears to be unable to efficiently support VBR traffic. POAC-QG makes extended use of traffic priorities in order to differentiate the TSs according to their application. The proposed superframe using slots decreases the total overhead, provides better synchronization, since every station is informed by the beacon of the exact time slots assigned to each station, and thus it potentially allows the use of an efficient power saving mechanism. POAC-QG employs a direct QoS negotiation mechanism that supports multiple quality levels for the TSs. This mechanism and the dynamic bandwidth allocation provide support to multiple TSs to the best quality the protocol can achieve. The simulation results reveal this behavior and show that POAC-QG always performs superiorly than HCCA when comparing the packet jitter, TS buffer size and packet delay. As future work, POAC-QG can be enhanced with a power saving mechanism and it can be combined with an efficient BT protocol in place of EDCA in order to form a complete high-performance protocol for infrastructure WLANs.

REFERENCES

[1] IEEE 802.11e WG, *P802.11e: Draft Amendment to Standard [for] 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: Amendment 7: Medium Access Control (MAC) Quality of Service (QoS) Enhancements*, 2005. Draft Version: D13.

[2] P. Nicopolitidis, M. S. Obaidat, G. I. Papadimitriou, and A. S. Pomportsis, *Wireless Networks*. Hoboken, NJ: Wiley, Jan. 2003.

[3] P. Nicopolitidis, G. I. Papadimitriou, and A. S. Pomportsis, "Distributed protocols for Ad-Hoc wireless LANs: A learning-automata-based approach," *Ad Hoc Netw.*, vol. 2, no. 4, pp. 419–431, Oct. 2004.

[4] P. Nicopolitidis, G. I. Papadimitriou, and A. S. Pomportsis, "Learning-automata-based polling protocols for wireless LANs," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 453–463, Mar. 2003.

[5] P. Nicopolitidis, G. I. Papadimitriou, and A. S. Pomportsis, "Using learning automata for adaptive push-based data broadcasting in asymmetric wireless environments," *IEEE Trans. Veh. Technol.*, vol. 51, no. 6, pp. 1652–1660, Nov. 2002.

[6] G. I. Papadimitriou and A. S. Pomportsis, "Adaptive MAC protocols for broadcast networks with bursty traffic," *IEEE Trans. Commun.*, vol. 51, no. 4, pp. 553–557, Apr. 2003.

[7] G. I. Papadimitriou and A. S. Pomportsis, "Learning automata-based TDMA protocols for broadcast communication systems with bursty traffic," *IEEE Commun. Lett.*, vol. 4, no. 3, pp. 107–109, Mar. 2000.

[8] A. Chandra, V. Gumalla, and J. O. Limb, "Wireless medium access control protocols," *Commun. Surveys Tuts.*, vol. 3, no. 2, pp. 2–15, Apr. 2000. Second Quarter.

[9] I. F. Akyildiz, J. McNair, L. Carrasco, and R. Puigjaner, "Medium access control protocols for multimedia traffic in wireless networks," *IEEE Netw.*, vol. 13, no. 4, pp. 39–47, Jul./Aug. 1999.

[10] ETSI Broadband Radio Access Network (BRAN), *HIgh PErformance Radio Local Area Network (HIPERLAN) Type 1; Functional Specification*. EN 300 652 V1.2.1 (1998-07).

[11] G. I. Papadimitriou, T. D. Lagkas, and A. S. Pomportsis, "HIPERSIM: A sense range distinctive simulation environment for HiperLAN systems," *Simul., Trans. Soc. Model. Simul. Int.*, vol. 79, no. 8, pp. 462–481, Aug. 2003.

[12] T. D. Lagkas, G. I. Papadimitriou, and A. S. Pomportsis, "QAP: A QoS supportive adaptive polling protocol for wireless LANs," *Comput. Commun.*, vol. 29, no. 5, pp. 618–633, Mar. 2006.

[13] T. D. Lagkas, G. I. Papadimitriou, and A. S. Pomportsis, "A high performance QoS supportive protocol for wireless LANs," in *Proc. IEEE MELECON*, Dubrovnik, Croatia, May 2004, pp. 571–574.

[14] K.-C. Chen and C.-H. Lee, "Group randomly addressed polling for wireless data networks," in *Proc. IEEE ICC*, 1994, pp. 1713–1717.

[15] M. J. Karol, Z. Liu, and K. Y. Eng, "An efficient demand-assignment multiple access protocol for wireless packet (ATM) networks," in *Wireless Networks*, vol. 1.3. New York: ACM Press, 1995, pp. 267–279.

[16] F. Bauchot, S. Decrauzat, G. Marmigere, L. Merakos, and N. Passas, "MASCARA, a MAC protocol for wireless ATM," in *Proc. ACTS Mobile Summit*, Granada, Spain, Nov. 1996, pp. 647–651.

[17] D. Petras and A. Kramling, "MAC protocol with polling and fast collision resolution for an ATM air interface," in *Proc. IEEE ATM Workshop*, San Francisco, CA, Aug. 1996.

[18] D. Raychaudhuri, L. J. French, J. Siracusa, S. K. Biswas, R. Yuan, P. Narasimhan, and C. A. Johnston, "WATMnet: A prototype wireless ATM system for multimedia personal communication," *IEEE J. Sel. Areas Commun.*, vol. 15, no. 1, pp. 83–95, Jan. 1997.

[19] J. G. Kim and I. Widjaja, "PRMA/DA: A new media access control protocol for wireless ATM," in *Proc. ICC*, Dallas, TX, Jun. 1996, pp. 1–19.

[20] D. A. Dyson and Z. J. Haas, "The dynamic packet reservation multiple access scheme for multimedia traffic," *ACM/Baltzer J. Mob. Netw. Appl.*, vol. 4, no. 2, pp. 87–99, 1999.

[21] G. Bianchi, F. Borgonovo, L. Fratta, L. Musumeci, and M. Zorzi, "C-PRMA: A centralized packet multiple access for local wireless communications," *IEEE Trans. Veh. Technol.*, vol. 46, no. 2, pp. 422–436, May 1997.

[22] I. Chlamtac, M. Conti, and J. J.-N. Liu, "Mobile ad hoc networking: Imperatives and challenges," *Elsevier Ad Hoc Netw.*, vol. 1, no. 1, pp. 13–64, 2003.

[23] J. Sanchez, R. Martinez, and M. W. Marcellin, "A survey of MAC protocols proposed for wireless ATM," *IEEE Netw. Mag.*, vol. 11, no. 6, pp. 52–62, Nov./Dec. 1997.

[24] T. Issariyakul, E. Hossain, and D. I. Kim, "Medium access control protocols for wireless mobile ad hoc networks: Issues and approaches," *Wirel. Commun. Mob. Comput.*, vol. 3, no. 8, pp. 935–958, Dec. 2003.

[25] IEEE 802.11 WG, *International Standard [for] Information Technology—Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements—Part 11:Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 1999. Reference number ISO/IEC 8802-11:1999(E).

[26] Q. Ni, L. Romdhani, and T. Turletti, "A survey of QoS enhancements for IEEE 802.11 wireless LAN," *Wiley J. Wirel. Commun. Mob. Comput. (JWCMC)*, vol. 4, no. 5, pp. 547–566, 2004.

[27] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[28] A. Grilo, M. Macedo, and M. Nunes, "A scheduling algorithm for QoS support in IEEE 802.11e networks," *IEEE Commun. Mag.*, vol. 10, no. 3, pp. 36–43, Jun. 2003.

[29] P. Larcheri and R. Lo Cigno, "Scheduling in 802.11e: Open-loop or closed-loop?" in *Proc. WONS*, Les Menuires, France, Jan. 18–20, 2006, pp. 114–122.

[30] Q. Ni, P. Ansel, and T. Turletti, "A fair scheduling scheme for HCF," *IEEE 802.11e Working Group Document: IEEE 802.11-03-0577-01-000e*, Jul. 23, 2003.

[31] C.-T. Chou, S. N. Shankar, and K. G. Shin, "Achieving per-stream QoS with distributed airtime allocation and admission control in IEEE 802.11e wireless LANs," in *Proc. IEEE INFOCOM*, 2005, vol. 3, pp. 1584–1595.

[32] E. Gilbert, "Capacity of a burst noise channel," *Bell Syst. Tech. J.*, vol. 39, no. 5, pp. 1253–1265, Sep. 1960.

[33] M. Zorzi, R. R. Rao, and L. B. Milstein, "On the accuracy of a first-order Markov model for data transmission on fading channels," in *Proc. ICUPC*, Tokyo, Japan, Nov. 1995, pp. 211–215.

[34] K. Pawlikowski, H.-D. J. Jeong, and J.-S. R. Lee, "On credibility of simulation studies of telecommunication networks," *IEEE Commun. Mag.*, vol. 40, no. 1, pp. 132–139, Jan. 2002.

Thomas D. Lagkas, photograph and biography not available at the time of publication.

Georgios I. Papadimitriou (M'96–SM'02), photograph and biography not available at the time of publication.

Petros Nicopolitidis, photograph and biography not available at the time of publication.

Andreas S. Pomportsis, photograph and biography not available at the time of publication.