An Adaptive Multimedia QoS Scheduler for 802.11e Wireless LANs

Inanc Inan *Student Member*, Feyza Keceli *Student Member*, and Ender Ayanoglu *Fellow*

Center for Pervasive Communications and Computing
Department of Electrical Engineering and Computer Science
The Henry Samueli School of Engineering
University of California, Irvine
Email: {iinan, fkeceli, ayanoglu}@uci.edu

***Abstract*— The IEEE 802.11e standard defines the Hybrid Coordination Function (HCF) which specifies the QoS algorithms in the MAC layer of the next generation 802.11 wireless LAN (WLAN). The HCF is composed of two access functions: A distributed contention-based channel access function (EDCA) providing prioritized QoS and a centralized polling-based channel access function (HCCA) providing parameterized QoS. In this paper, we address the efficient medium access control (MAC) of the mostly variable bit rate (VBR) and bursty multimedia traffic in HCCA and propose a novel application-aware adaptive 802.11e QoS scheduler. The proposed scheduler coordinates the resource allocation of the associated flows using adaptive service intervals, transmit opportunities, and polling order. The service schedule is based not only on the traffic classification done according to the packet size, interval distribution, and the direction but also on the instantaneous buffered traffic conditions. The proposed scheduler is shown to have significant performance gains in terms of multimedia traffic channel utilization, packet loss ratio, delay, and jitter when compared with the other schemes.**

I. Introduction

IEEE 802.11 Wireless LAN (WLAN) [1] standard provides best-effort service at the medium access control (MAC) layer. Due to its low cost and easy installation, the 802.11 WLAN is enjoying widespread deployment. On the other hand, increasing use of real-time multimedia applications in the 802.11 WLAN require Quality-of-Service (QoS) support.

An IEEE 802.11 task group, TGe, has approved the QoS enhanced protocol, IEEE 802.11e [2]. The IEEE 802.11e standard specifies improved MAC functions which provide prioritized and parameterized QoS services at the MAC layer. The 802.11e MAC model defines the Hybrid Coordination Function (HCF) which combines a distributed contention-based channel access mechanism, referred to as Enhanced Distributed Channel Access (EDCA), and a centralized polling-based channel access mechanism, referred to as HCF Controlled Channel Access (HCCA). In a basic service set that provides the QoS facility (QBSS), the medium access is controlled by the alternating operation of EDCA and HCCA.

Although the EDCA function may provide satisfactory service differentiation in low-load environments, its contention-based nature results in impaired performance, low channel efficiency, and lack of QoS guarantee in the presence of heavy traffic load [3], [4]. On the other hand, the HCCA function is designed to always meet the negotiated QoS requirements of the admitted flows. The HCCA mechanism uses the Hybrid Coordinator (HC) located at the Access Point that supports the QoS facility (QAP) to manage radio resource allocation. The IEEE 802.11e standard proposes a simple reference scheduling algorithm, referred to as TGe scheduler, for controlling the medium access of stations that support QoS (QSTA) [2]. However, the TGe scheduling algorithm has the basic flaw that it always sticks with fixed transmit durations, polling order and service interval which cannot capture the varying network conditions. This algorithm is not efficient for the mostly variable bit rate (VBR) and/or bursty multimedia/internet traffic and the time-varying wireless channel.

There is ongoing research on improving/optimizing VBR traffic scheduling performance of the HCCA function. TGe scheduler's scheme is extended in FHCF [5] and FDBS [6] where real-time queue length estimations are used to tune transmit durations of the stations on the run. TGe scheduler's static approach on polling list and service interval is made adaptive regarding the VBR traffic specifications (TSPEC) in [7] and the HCF packet exchange sequences in [8]. A variant of the Earliest Deadline First (EDF) algorithm [9], SETT-EDD [10], enables using distinct service intervals for each QSTA, but has extensive polling overhead and misses the handling of instantaneous traffic conditions.

Our contribution in this paper is an application-aware adaptive HCCA scheduling method. The novel approach is that the proposed scheduler adapts the service schedule depending on the direction of the traffic flows, the guidelines specified in a multimedia traffic profile classification, and the instantaneous MAC layer buffer conditions monitored by the HC. Via simulations, the approach is shown to be effective in providing parameterized QoS and efficient channel utilization in the case of the scenario of interest, a WLAN consisting of a QAP and several QSTAs.

The rest of the paper is organized as follows. We give a brief overview of the IEEE 802.11e HCCA function in Section II. In Section III, the proposed application-aware adaptive HCCA scheduler is described. The performance evaluation of the proposed scheduler is the topic of Section IV. Finally, we report our concluding remarks in Section V.

1-4244-0355-3/06/$20.00 (c) 2006 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

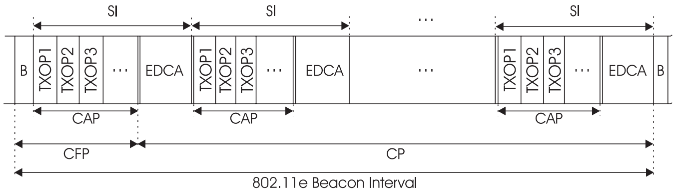


Fig. 1. HCF superframe structure

II. IEEE 802.11e HCCA Function

The HCCA function [2] defines a centrally-controlled polling-based medium access scheme for IEEE 802.11e WLANs. It is designed to provide parameterized QoS support through efficient polling and scheduling.

The HC periodically broadcasts beacon frames to communicate network identification and management parameters specific to the wireless network. The beacon initiates the so-called 802.11e superframe which comprises an optional Contention Free Period (CFP) followed by a Contention Period (CP). The 802.11e superframe corresponds to an IEEE 802.11e beacon interval as shown in Fig. 1. The HC may start Controlled Access Phases (CAP) to allocate Transmit Opportunities (TXOP) to the QSTAs either in CP or CFP. During a CAP, the medium access is controlled by the QAP, and no QSTA can transmit unless it is polled by the QAP. In CP, the access is contention-based and left to EDCA functions of the QSTAs and the QAP [2].

The QAP has the highest priority to access the medium among all QSTAs since it may seize the channel by using a shorter interframe space (PIFS) without waiting any backoff time. The QAP may either send a poll frame, presenting an HCCA TXOP to a QSTA, or just start transmitting download traffic. Once the QAP or the QSTA gains the HCCA TXOP, it may send as many frames as it can fit in its TXOP Duration (TD). If a QSTA with no buffered data receives a poll frame, it may send a null frame, so that the QAP that hears it may change the TXOP owner or end the CAP.

Every QSTA runs a QoS reservation procedure with the HC for all of its traffic streams that need parameterized QoS support. The Station Management Entity (SME) at the HC decides whether the Traffic Stream (TS) is admitted or not regarding the Traffic Specification (TSPEC) provided by the QSTA. The main parameters of the TSPEC for a TS are [2]:

- **Traffic Type :** A bit length parameter denoting whether the TS follows a periodic (e.g., isochronous TS of MAC Service Data Units (MSDU), with constant or variable sizes, that are originated at a fixed rate) or an aperiodic pattern.
- **Traffic Stream Identification** ($TSID$)**:** Identification number assigned by the QSTA for the TS.
- **Direction :** Two-bit length field showing whether the TS is downlink, uplink, or direct link.
- **Mean Data Rate** (ρ)**:** Average bit rate at the MAC Service Access Point (SAP) for the transfer of the frames.
- **Delay Bound** (D)**:** Maximum amount of time allowed to transport the frames across the wireless interface including the queueing delay.
- **Nominal MSDU Size** (L)**:** Average MSDU size of the frames.
- **Maximum MSDU Size** (M)**:** Maximum MSDU size of the frames.
- **Minimum Service Interval** (mSI)**:** Minimum interval between the start of successive service periods.
- **Maximum Service Interval** (MSI)**:** Maximum interval between the start of successive service periods.
- **Burst Size** (BS)**:** Maximum burst size that arrives at the MAC SAP at the peak data rate.
- **Minimum PHY Rate** (R)**:** Desired minimum physical bit rate to be used for the TS.
- **Service Start Time** (SST)**:** The time when the first service period (SP) is expected to start.

The IEEE 802.11e standard document proposes a simple reference scheduler for the HCCA function [2]. This algorithm uses ostensibly mandatory TSPEC parameters, ρ, MSI (or D if specified), and L, for radio resource reservation deciding on an aggregate service schedule. Since the TSID specified within a poll packet is not binding, the polls are issued to the QSTAs and not to individual TSs.

According to the TGe scheduler, the HC first decides on a service interval (SI), the highest submultiple value of the beacon interval which is smaller than the minimum of the MSIs of all admitted flows. According to the SI, aggregate HCCA TXOPs allocated to the QSTAs are calculated. Let TD for a QSTA i with n admitted traffic streams be denoted as TD_i. TD_i is calculated as

$$TD_i = \sum_{j=1}^{n} max\left(\frac{N_j \cdot L_j}{R_j} + O, \frac{M_j}{R_j} + O\right), \quad \text{where } N_j = \left\lceil \frac{SI \cdot \rho_j}{L_j} \right\rceil . \qquad (1)$$

In this formula, O denotes the total MAC/PHY overhead in the frame exchange sequences. The integer N_j shows the ceiling of average number of frames generated by the TS in the SI period. As shown in Fig. 1, the TGe scheduler cuts the beacon interval into fixed SIs. The scheduler issues polls to the QSTAs consecutively in a round robin manner.

For admission control, the HC should check whether with the newly admitted stream the fraction of time that can be used for CAPs go over a limit value or not. The $dot11CAPrate$ specifies the proportion of time per $64\mu s$ that can be spent for CAPs. The SI and TDs are recalculated including the new stream in the calculations. The stream is admitted if the inequality

$$\sum_{i=1}^{k} \frac{TD_i}{SI} \leq \frac{T_b - T_{CP}}{T_b} = \frac{dot11CAPrate}{64\mu s} \qquad (2)$$

holds, assuming there are a total of k stations, T_b denotes the beacon interval, and T_{CP} is the minimum time required for EDCA traffic in each T_b.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

III. THE APPLICATION-AWARE ADAPTIVE HCCA SCHEDULER

Although TGe scheduling method of constantly using the fixed TD calculated approximately from mean TSPEC figures may be sufficient for CBR applications, this cannot absorb the variances in the rate and the packet size of the VBR multimedia traffic that has frequent packet bursts. Having the same SI for all QSTAs that are probably serving various traffic profiles with different average/instantaneous QoS requirements is not efficient (note that QAP is also counted as a QSTA in the description). A good scheme should exploit the traffic characteristics as well as the monitored instantaneous network conditions. The time-varying VBR network load on a time-varying wireless channel can be best controlled and served by an adaptive algorithm.

The proposed application-aware adaptive HCCA scheduler uses the Earliest Deadline First (EDF) scheduling discipline [9], [10]. The EDF approach makes the polling order adaptive according to the calculated deadlines of associated traffic profiles. The multimedia traffic is scheduled in the sense that each QSTA has distinct SI. The novelty is that the SI and the TXOP of the QSTA are adapted according to the traffic characteristics as well as instantaneous buffer occupancy information in real time.

In the core of the scheduling algorithm, each QSTA including QAP has its own dynamically updated minimum QSTA service interval ($mQSI$) and maximum QSTA service interval ($MQSI$) values. The $mQSI$ and $MQSI$ are adaptively tuned for reducing packet delay, jitter, and losses as well as for increasing channel utilization.

If we let t_i' and t_i'' be the time when an HCCA TXOP starts and ends for QSTA$_i$ respectively, at an arbitrary time t, another HCCA TXOP may be issued to a QSTA which satisfies the inequality in (3). Then, we propose that S be defined to be the set of QSTAs that are pollable at t

$$S = \{i : t_i'' + mQSI_i \leq t\}. \tag{3}$$

As EDF implies, the first QSTA to be polled at any time is the one with the smallest deadline among the QSTAs in S:

$$\min_j(t_j'' + MQSI_j) \text{ where } j \in S. \tag{4}$$

The HC always has the exact instantaneous knowledge of downlink traffic buffered at the transmission queues. On the other hand, for uplink traffic, this information is usually limited to the scheduler at the QAP especially when VBR multimedia applications are run at the QSTAs. Making the distinction on the direction of the multimedia traffic, the proposed application-aware adaptive HCCA scheduler effectively processes the information provided by TSPEC parameters and MAC packet headers resulting in efficient scheduling. The originality of the proposed MAC layer scheduling algorithm is that it uses both application layer traffic characteristics and estimated/known link layer buffer size information (monitored conditions) in real-time resource allocation and it differentiates between uplink and downlink scheduling.

A. Uplink Scheduler

The proposed uplink scheduler adapts the service interval parameters, $mQSI$ and $MQSI$, of each QSTA according to the application characteristics, the TSPEC parameters, and the previous TD requests of the associated flows at the end of an HCCA TXOP. Adapting $mQSI$ and $MQSI$ at the end of the TXOP enables the scheduler to exploit instantaneous buffer size information in the scheduling decision. On the other hand, the aggregate service length parameter, TD, is calculated at the start of the HCCA TXOP, including the average estimate on the number of the packets to be buffered since the last service time and the previous TD requests in the decision.

Our proposed uplink scheduler stores the following parameters for each associated flow j of QSTA$_i$ at a table.

- Next service due time (t_{due_j}): The closest time that flow j can be served again.
- Next service deadline time (t_{dead_j}): The deadline for the next service to be issued for flow j.
- Last packet transmission end time (t_{pt_j}): The end of the last recorded packet transmission time at the HC for flow j.
- Last TD request (t_{req_j}): The last recorded TD request for flow j. If no request is made, t_{req_j} is set to 0.

Assume QSTA$_i$ has n associated flows, then our proposed uplink service schedule equations are as follows:

$$mQSI_i = \begin{cases} \min\limits_{1\leq j\leq n}\{t_{due_j} - t_i''\}, & if\ \forall j:\ t_{req_j} = 0 \\ 0, & if\ \exists j:\ t_{req_j} \neq 0 \end{cases} \tag{5}$$

$$MQSI_i = \begin{cases} \min\limits_{1\leq j\leq n}\{t_{dead_j} - t_i''\}, & if\ \forall j:\ t_{req_j} = 0 \\ \min\limits_{j\in\{t_{req_j}\neq 0\}}\{t_i' + MSI_j\}, & if\ \exists j:\ t_{req_j} \neq 0 \end{cases} \tag{6}$$

$$\begin{aligned} & TD_i = \textstyle\sum_{j=1}^{n} min(t_{burst_j}, t_{mean_j} + t_{req_j}) \\ where\quad & t_{burst_j} = \frac{BS_j}{R_j} + O, \\ & t_{mean_j} = max\left(\frac{N_j \cdot L_j}{R_j} + O, \frac{M_j}{R_j} + O\right), \\ & N_j = \left\lceil \frac{(t_i' - t_{pt_j}) \cdot \rho_j}{L_j} \right\rceil. \end{aligned} \tag{7}$$

As (5), (6), and (7) imply, the uplink service schedule is adapted according to the presence of the TD requests from the associated QSTAs. If there is not any TD request from any flow j at the end of an HCCA TXOP issued to QSTA$_i$, the service interval parameters, $mQSI_i$ and $MQSI_i$, are calculated using t_{due} and t_{dead} values of each individual flow respectively. The decision on t_{due} and t_{dead} is made according to the TSPEC parameters and the flow characterization. This approach ensures that the QSTA with no buffered frames is not polled during the time when the associated flows are not expected to generate a packet. On the other hand, if there is at least one TD request, this means that the QSTA$_i$ already has buffered frames to transmit and is labelled as immediately pollable by setting $mQSI_i$ to 0. Therefore, the bursts that cannot be transmitted at the current HCCA TXOP because of time considerations can be served at an upcoming HCCA TXOP within the delay bound by granted TXOP requests.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

On top of varying packet sizes and interarrival times of VBR flows, the QSTAs will also be adapting their PHY rates with respect to time-varying wireless channel which may also highly alter the packet transmission times in practice. Therefore, the proposed uplink scheduler basically uses mean TSPEC figures in TD calculation as other studies of the literature [2], [8], [10]. As in (7), the service length of QSTA$_i$, TD_i, is calculated at the start of an HCCA TXOP including the service time for the average number of packets generated since t_{pt_j} (t_{mean_j}) and the TD requests (t_{req_j}) in the calculation. The assigned TD for any flow j is ensured not to be over the specified burst size (t_{burst_j}). Moreover, practically, a module in the HC should check whether the QSTAs go over their specified medium time or not. If yes, the TXOP requests are not authorized and the TD is assigned according to the remaining medium time.

The application-aware adaptive scheduler handles the associated uplink traffic streams depending on the periodic or aperiodic structure of the traffic pattern (detected by using traffic type field of the TSPEC) and the constant or variable sizing of the frames (detected by comparing nominal and maximum MSDU sizes of the TSPEC). The state-of-the-art multimedia applications used in this study fall into three different categories. The service schedule parameters, t_{due_j} and t_{dead_j}, are calculated depending on this categorization for each flow j.

1) Constant packet size with variable packet interval: Due to the inherent nature of voice data, voice over IP (VoIP) traffic fits into an on/off traffic model [11]. The VoIP codec model used in this study, G.279A, generates frames with constant size at constant rate when it is on (active). However, silent intervals present in voice traffic make the VoIP application have variable packet interval. In [11], the audio model proposes exponentially distributed on and off periods.

The uplink HCCA scheduler calculates t_{due_j} and t_{dead_j} of audio flows thus adapts $mQSI$ and $MQSI$ of the QSTAs running VoIP applications according to the outcome of a Voice Activity Detection (VAD) algorithm in order not to send excessive polls when the audio stream is off (inactive). This approach decreases MAC overhead considerably and enables efficient use of radio resources during off times of audio traffic. The design of an efficient VAD method is out of the scope of this paper. In this work, we employed the Hybrid Activity Detection (HAD) method proposed in [12] in the uplink scheduler.

The scheduler labels an audio flow j as inactive if it does not transmit in any HCCA TXOP for a predetermined time interval, τ_j. Any active flow is scheduled for transmission based on mSI of the flow. On the other hand, the scheduler issues polls (detection point) to inactive audio flows in longer service intervals of varying length, $T_{th}(k)$. The non-increasing values for $T_{th_j}(k)$ are updated recursively at each detection point using (8) until the audio flow j is detected to be active. Let silent periods of the audio flow be exponentially distributed with parameter α, and p be a constant real number, then

$$T_{th_j}(k) = \min\{p \cdot r_k, mSI_j\}$$
$$\text{where } r_k = r_{k-1} - T_{th_j}(k-1),\ r_0 = \frac{1}{\alpha},$$
$$T_{th_j}(0) = 0,\ \text{and } 0 < p < 1. \quad (8)$$

In [12], it is analytically shown that (8) can reduce the polling overhead in terms of the number of the polls before detecting a silence-to-talkspurt change as it is compared with IEEE 802.11e Round-Robin algorithm. An appropriate choice of p ensures an acceptable higher bound to exist on the worst-case average polling delay.

When the HC detects a talkspurt in an HCCA TXOP, it labels the flow again as active, and resets the activity detection parameters. If a talkspurt occurs before the next detection point, the VoIP packet is sent during CP using voice access category (AC_VO) of EDCA function (HAD). For the details of the HAD method, the reader is referred to [12].

Therefore, at the end of an HCCA TXOP, which started at t', t_{due_j} and t_{dead_j} for a VoIP flow j is set as follows:

$$t_{due_j} = \begin{cases} t' + mSI_j, & if\ j\ is\ active \\ t' + T_{th_j}, & if\ j\ is\ inactive \end{cases} \quad (9)$$

$$t_{dead_j} = \begin{cases} t' + MSI_j, & if\ j\ is\ active \\ t_{due_j} + MSI_j, & if\ j\ is\ inactive \end{cases} \quad (10)$$

2) Variable packet size with constant packet interval: MPEG-4 video encoder traces used in this study employ a constant reference frame rate [13] which result in constant packet intervals. On the other hand, each frame, depending on its frame type, quality level, and codec parameters vary in size.

In the case of an uplink isochronous stream with a specified SST within the TSPEC, the HC has the deterministic knowledge of the frame generation times. Especially, the overhead to start a CAP, i.e., waiting for the medium to become idle for PIFS, delays the polling service schedule at every service period which results in excessive packet delay. However, the proposed scheduler also bases the $mQSI$ of the isochronous stream on the deterministic time it calculates from SST instead of using only the last HCCA TXOP end time. This enables serving the isochronous stream frames in shorter delays, and even protects the flow from potential losses.

Therefore, at the end of an HCCA TXOP, which started at t', t_{due_j} and t_{dead_j} for a MPEG-4 flow j is set as follows:

$$t_{due_j} = SST_j + k \cdot mSI_j$$
$$where\ k = \min_l\{t' \leq (SST_j + l \cdot mSI_j)\} \quad (11)$$

$$t_{dead_j} = t_{due_j} + MSI_j. \quad (12)$$

3) Variable packet size with variable packet interval: H.263 video encoder traces used in this study have both variable packet size and interval time [13]. The H.263 encoder groups some frames into one frame in order to improve video quality which makes the interval time variable.

Proposed application-aware uplink scheduler bases $mQSI$

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

on the MSI for aperiodic streams. The H.263 traffic has variable packet intervals and the instantaneous packet interval time may deviate highly from mSI. Therefore, in such a case, using $\gamma \cdot MSI$ in t_{due} calculation is more efficient for increased channel utilization where $0 \leq \gamma \leq 1$. A good γ value can be selected based on the channel load and state taking necessary retransmissions into consideration. Therefore, at the end of an HCCA TXOP, which started at t', t_{due_j} and t_{dead_j} for a MPEG-4 flow j is set as follows:

$$t_{due_j} = t' + \gamma \cdot MSI_j \tag{13}$$

$$t_{dead_j} = t' + MSI_j \tag{14}$$

The uplink scheduler performance highly depends on the decision made for mSI and MSI at a QSTA. The SETT-EDD algorithm [10] suggests the calculation of mSI within the MAC sublayer management entity (MSME) as L/ρ. Such an approach makes mSI vary over a broad range for the same VBR multimedia application, since L depends not only on the source characteristics but also on the fragmentation algorithms used at any layer of the QSTA. While a small mSI can lead to excessive polls, thus low channel utilization, a high mSI can result in fewer polls than necessary, thus high packet loss. Therefore, for optimizing the performance of the uplink HCCA scheduler, we propose that the SME, where the TSPECs are constructed, should base the mSI selection on the application characteristics and requirements as well as the information specific to the MAC layer. The MSI is calculated as $\beta \cdot D$ where $0 \leq \beta \leq 1$. A good β value can be selected based on the limit for the number of retries at the MAC layer and wireless channel state.

B. Downlink Scheduler

The application-aware adaptive HCCA scheduler follows a very simple yet highly efficient method for downlink scheduling. In the most common scenario, the downlink traffic is generated at a distinct point in the Internet. The information to construct downlink TSPECs as accurate as it is for uplink may not always be available. Therefore, basing the downlink service schedule mainly on instantaneous traffic conditions and controlling the access with respect to TSPEC information can be an effective approach.

The downlink scheduler assigns TD required to serve the buffered traffic to the downlink flows as long as the transmission time does not go over the provisioned medium time for the flows. The scheduler can always calculate an exact TD analyzing the buffered packets. If we let B_j be the buffer size for flow j, R_c be the current PHY rate used at the QAP, O be the MAC/PHY overhead and assume QAP has n associated downlink multimedia flows, then TD is calculated as follows:

$$TD = \frac{\sum_{j=1}^{n} B_j}{R_c} + O. \tag{15}$$

Rather than using the "wait for SI and then poll" approach in downlink scheduling employed in [2],[5],[6],[10], the proposed downlink scheduler schedules any downlink frame for transmission as soon as it arrives at the MAC SAP of the QAP. This approach minimizes downlink delay without introducing any extra overall overhead. Therefore, the scheduling parameters for a QAP serving n downlink streams are set as follows where p_{n_j} denotes the number of packets of flow j buffered in its transmission queue.

$$mQSI = \begin{cases} 0, & if\ \exists j:\ p_{n_j} \neq 0 \\ \infty, & if\ \forall j:\ p_{n_j} = 0 \end{cases} \tag{16}$$

$$MQSI = \begin{cases} \min\limits_{j \in \{p_{n_j} \neq 0\}} \{MSI_j\}, & if\ \exists j:\ p_{n_j} \neq 0 \\ \infty, & if\ \forall j:\ p_{n_j} = 0 \end{cases} \tag{17}$$

IV. Simulation Environment and Results

At the time of this writing, the official distribution of the public domain network simulator, ns-2 [14], only supports Distributed Coordination Function (DCF) in the IEEE 802.11 MAC layer. We constructed a functioning IEEE 802.11e HCF MAC simulation model for ns-2.28 [15]. The model is developed on top of the TKN ns-2.26 EDCA patch [16]. The model includes all the EDCA and HCCA functionalities stated in [2]. All HCF QoS data/control frame exchange sequences as well as optional dynamic link setup and immediate block ACK policy implementations are supported. The module implements abstract PHY layer models for IEEE 802.11a/b/g/n. Moreover, AWGN channel support is included according to the framework presented in [17].

The focus of the simulation study is to analyze the performance of the proposed HCCA scheduler in terms of total throughput, channel utilization, packet loss ratio, packet delay, and jitter under simulation scenarios including VBR multimedia traffic models. Any QoS frame that is not served within its delay bound is counted as a packet loss. The mean delay calculation includes the MAC queueing delay and denotes the average delay observed by successfully received packets at all QSTAs. The jitter is calculated as the average delay difference of two consecutively received frames at a QSTA. The performance of the proposed application-aware adaptive scheduler is also compared with the performance of EDCA, TGe scheduler, and SETT-EDD for the same set of scenarios to show proposed scheduler's effectiveness in handling VBR traffic.

The simulations consider three types of traffic sources; audio, video, and data. The voice traffic model implements G.729A VoIP application as an on/off traffic profile where talk-spurt and silent periods are both independent and exponentially distributed (mean on time: 352ms, mean off time: 650ms) [11]. For the video source models, we used traces of real MPEG-4 and H.263 video streams [13]. For data traffic, we used FTP traffic generator of ns-2 which is modelled as bulk data transfer and there is at least one packet to send at the transmission buffers anytime. While the transport protocol for multimedia flows is UDP, FTP traffic uses TCP. The TCP parameters are set to the default values of TCP agent implementation in ns-2. Real-time packets have 40-byte length RTP/UDP/IP header. Best-effort packets are of fixed length, 1540 bytes, including

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

TABLE I

TSPEC PARAMETERS FOR MULTIMEDIA FLOWS

TSPEC	G.729A	MPEG-4	H.263
ρ (Kbps)	24	174	255
D (ms)	60	60	60
L (bytes)	60	821	2419
M (bytes)	60	2088	3112
BS (bytes)	180	7072	10961

TCP/IP header. Table I summarizes some TSPEC parameters for the real-time multimedia applications.

The simulation scenarios implement a QBSS controlled by a QAP. Voice and video streams are scheduled for transmission via the HCCA function if it is used. The data traffic always uses the best-effort access category (AC_BE) of the EDCA function. The EDCA parameters are set to the default values stated in [2]. The beacon interval is 100ms. The simulation time is 100s.

The simulation results are reported for AWGN channels with very high signal-to-noise ratio (SNR), i.e., the wireless channel is assumed to be not prone to any errors during transmission. All the QSTAs have IEEE 802.11g PHY layer [18] without multirate support. Both the data and control rate are selected as the highest 802.11g mandatory rate defined, 24Mbps. Since there are no retransmissions because of channel errors, β is set to 1 for the proposed scheduler. In order to compensate for poll delay and collisions, γ is selected as 0.7. Throughout extensive simulations, these are seen to be appropriate selections for the scenarios. We assign the mSI of MPEG-4 or H.263 flows with the reference frame rate of the video encoder assuming this cross layer information is supplied via the SME. The mSI of G.729A flows is set to audio frame interarrival period, ρ/L. The QSTAs should not consider inactive periods in the ρ decision of VoIP flows for accurate use of mSI in the scheduler. The inactivity detection period of G.729A flows, τ, is set to $2 \cdot mSI$. In (8), p is set to 0.3 as suggested in [12].

Any QoS stream is admitted to the WLAN. We set the $dot11CAPrate$ to $64\mu s$ in order to test the performance limits of the scheduling algorithms at heavy load and observe HCCA scheduling channel utilization efficiency. FTP traffic fills up the bandwidth that HCCA does not use with EDCA traffic. The lower the HCCA overhead for CAP scheduling, the higher the FTP throughput. Therefore, in a sense, the total throughput represents HCCA scheduling performance in terms of channel utilization efficiency.

We are considering a simulation scenario where the associated QSTAs run one of the following sessions: A bidirectional G.729A VoIP session, an uplink video streaming session using an MPEG-4 or an H.263 codec, or an uplink FTP data session. Although a more likely scenario is that a wireless user will be connected to a video server on the wired side and will be using downlink bandwidth for video streaming, we chose a less likely scenario to test the scheduling performance on

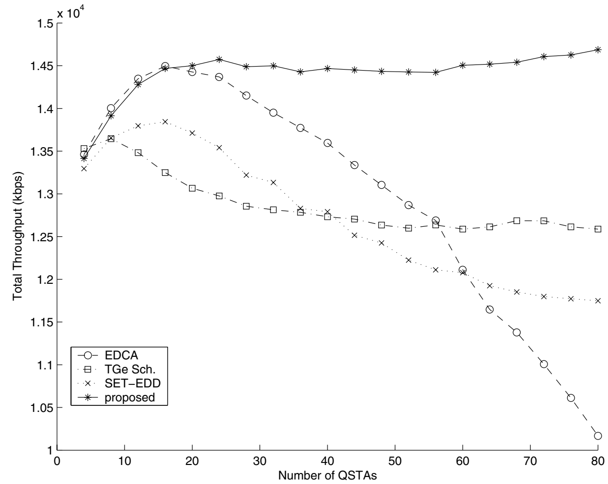


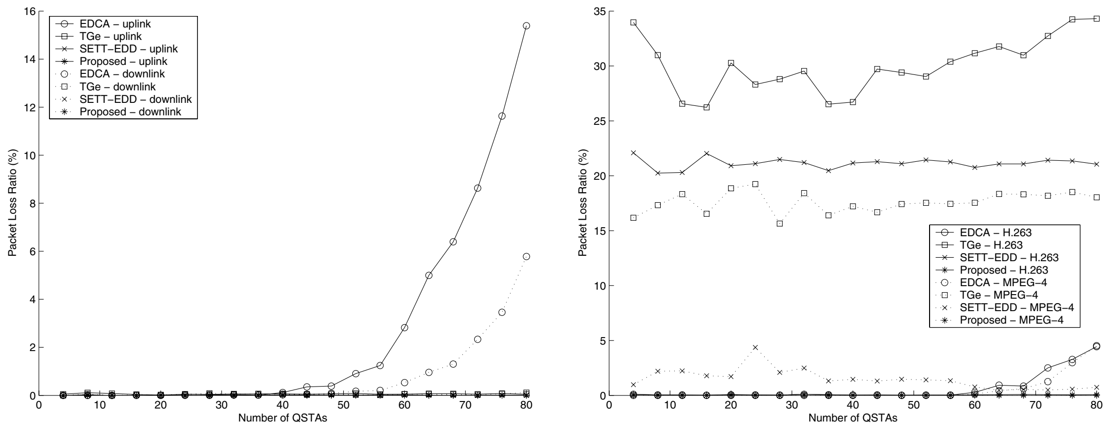
Fig. 2. Total aggregate throughput of audio, video and data streams

the uplink. The downlink performance of the scheduler can be observed to be satisfactory via downlink G.729A results. At a time, there are 4α QSTAs and a QAP in the network with a total of 2α G.729A, α MPEG-4, α H.263 and α FTP flows. The performance is measured against increasing channel load, i.e. increasing α. The control traffic of the applications is neglected since it consumes much less bandwidth when compared with the application.

Fig. 2 depicts the total aggregate throughput of all flows. The results also represent the efficiency of different methods in channel utilization. As the load increases while the adaptive scheduler still provides efficient channel utilization and high throughput, the performance of EDCA and SETT-EDD decrease because of increasing collision and polling overhead respectively. When compared to SETT-EDD, the proposed scheduler provides more than 20% total throughput at moderate loads. Although not simulated here, the improvement is expected to be more pronounced in a mixed environment where some QSTAs could only support lower rate PHY modes making SETT-EDD polling bandwidth overhead even higher. At high loads, TGe scheduler has higher total throughput than SETT-EDD since the polling overhead is less. However, packet loss ratio analysis shows TGe scheduler has degraded QoS performance.

Fig. 3 shows the average packet loss ratio for audio and video flows. As the results imply, EDCA has the collision overhead which makes it inefficient for QoS provision at high loads. Because of sticking with fixed TXOPs, the TGe scheduler has high packet loss ratio for VBR MPEG-4 and H.263 flows which can not be tolerable for video streaming. SETT-EDD has also considerable QoS impairment for H.263 since it bases its service interval on L/ρ. For the SETT-EDD simulation, $mQSI$ for H.263 is calculated to be around $75ms$ although H.263 source may generate some frames in $40ms$ intervals. Such high $mQSI$ selection results in significant QoS performance loss. MPEG-4 flows, for which L/ρ happens to be approximately $35ms$ (which is appropriate), slightly suffer

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.



(a) G.729A

(b) MPEG-4 & H.263

Fig. 3. Packet loss ratio for multimedia flows

from packet loss because the token bucket cannot always handle the bursty traffic. On the other hand, the application-aware adaptive scheduler allocates the parameterized bandwidth using TXOP requests and $mQSI$ adaptation in service scheduling which ends up with negligible packet loss ratio under all loads.

Fig. 4 and Fig. 5 display the average packet delay and average jitter respectively for audio and video flows with respect to increasing number of QSTAs. EDCA and TGe scheduler results are removed from the graphs in order not to complicate the figures. Their QoS performance in terms of packet loss ratio is unsatisfactory.

SETT-EDD has an average delay of approximately $mSI/2$ in both downlink and uplink as a result of high EDCA best effort traffic. The proposed downlink scheduler has significant improvements on average downlink packet delay. Adapting service intervals according to the outcome of the HAD algorithm also results in up to 40% percent gain in uplink voice packet delay when compared with SETT-EDD. Via adapting service intervals with respect to reference video frame rate and service start time, the proposed scheduler serves VBR MPEG-4 flows with considerably lower packet delay and jitter. H.263 flows experience improved delay and jitter with lower scheduling overhead since service intervals are adapted according to the MSIs and reference video frame rate.

V. Conclusions

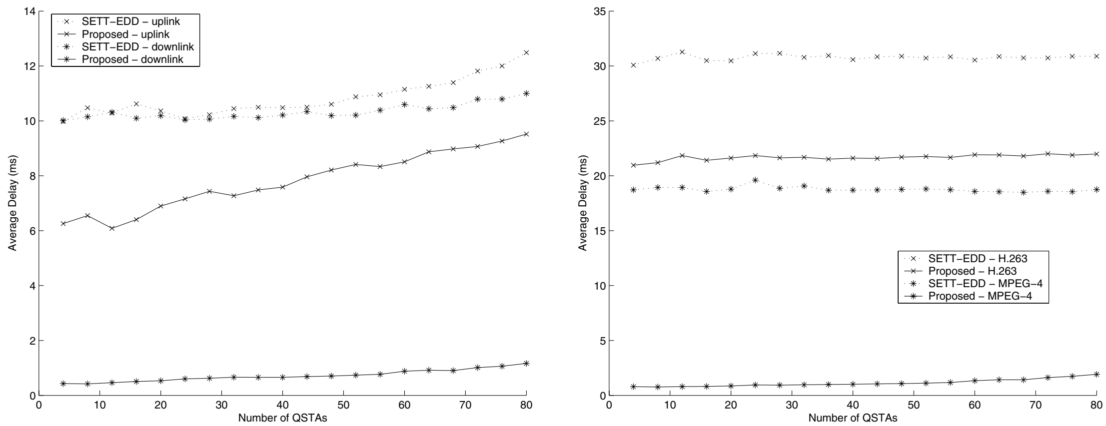
In this paper, we proposed an application-aware adaptive HCCA scheduling algorithm which is fully compliant with the specifications of the IEEE 802.11e standard. The approach used in the service scheduling is novel in the sense that the algorithm adapts service intervals, polling order, and transmission opportunities depending both on the traffic characteristics and instantaneous network conditions. The algorithm introduces additional hardware/firmware complexity in HCCA scheduling when compared with the TGe scheduler. The uplink scheduling performance also requires effective cross-layer coordination for the TSPEC construction. However, these are likely necessary extensions for efficient HCCA scheduling. The performance improvements in terms of efficient channel utilization and timely multimedia traffic delivery are very promising.

The extensive simulations carried out indicate the effectiveness of the algorithm in QoS provisioning for VBR multimedia traffic flows of a WLAN scenario composed of a QBSS. The proposed scheduler provides QoS guarantees with lower packet loss ratio, delay, and jitter. More flows can be admitted to the WLAN as a result of higher channel utilization.

Although the simulation results for the wireless channel with low SNR are not reported in this paper, the proposed scheduler has a number of specifications useful in mitigating channel noise effectively such as adapting the uplink service schedule regarding the TXOP requests from the QSTAs and basing the downlink service schedule on buffer occupancy at the QAP. These extensions make the failed data packets, retransmissions of which cannot fit in the current TXOP, be retransmitted much sooner rather than waiting another fixed service interval as it is in previous proposals, resulting in smaller packet loss ratio, delay, and jitter. We actually have simulation results that indicate the stated extensions improve the scheduling performance for AWGN channel with low SNR. The proposed application-aware adaptive scheduler shows similar performance gains in low SNR case as it is for very high SNR case. Due to space limitations, these results are not included.

Having the applications dynamically tuning the scheduling parameters poses some challenges on admission control. An efficient admission control algorithm design is left as future work. It should be noted that using a fair admission control algorithm and accepting fewer number of flows than the proposed scheduler can actually deal with efficiently will not alter the improved HCCA performance. Effective HCCA scheduling will increase the EDCA bandwidth, thus the best-effort QoS.

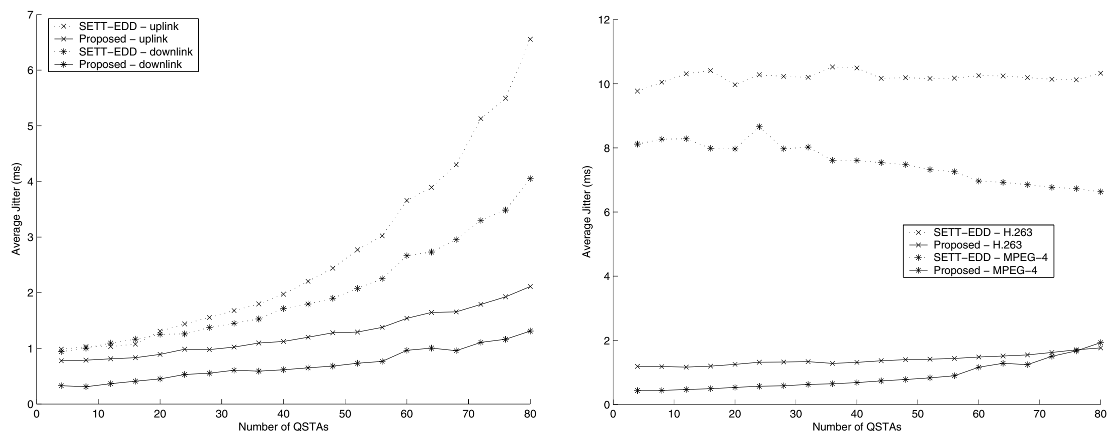
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.



(a) G.729A

(b) MPEG-4 & H.263

Fig. 4. Average delay for multimedia flows



(a) G.729A

(b) MPEG-4 & H.263

Fig. 5. Average jitter for multimedia flows

REFERENCES

[1] *IEEE Standard 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, IEEE 802.11 Std., 1999.

[2] *IEEE Standard 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: Medium access control (MAC) Quality of Service (QoS) Enhancements*, IEEE 802.11e Std., 2005.

[3] D. Chen, D. Gu, and J. Zhang, "Supporting Real-time Traffic with QoS in IEEE 802.11e Based Home Networks," in *Proc. IEEE CCNC '04*, January 2004.

[4] J. del Prado Pavon and S. S. N, "Impact of Frame Size, Number of Stations and Mobility on the Throughput Performance of IEEE 802.11e," in *Proc. IEEE WCNC '04*, March 2004.

[5] P. Ansel, Q. Ni, and T. Turletti, "An Efficient Scheduling Scheme for IEEE 802.11e," in *Proc. Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, WiOpt.*, March 1999.

[6] A. Annese, G. Boggia, P. Camarda, L. A. Grieco, and S. Mascolo, "Providing Delay Guarantees in IEEE 802.11e Networks," in *Proc. IEEE VTC '04 - Spring*, 2004.

[7] W. F. Fan, D. H. K. Tsang, and B. Bensaou, "Admission Control for Variable Bit Rate Traffic Using Variable Service Interval in IEEE 802.11e WLANs," in *Proc. IEEE ICCCN '04*, October 2004, pp. 447–453.

[8] I.-G. Lee, J.-B. Son, S.-R. Yoon, and S.-C. Park, "Efficient Block Size Based Polling Scheme for IEEE 802.11e Wireless LANs," in *Proc. IEEE VTC '05*, May 2005.

[9] J. A. Stankovic, M. Spuri, and K. Ramamritham, "Deadline Scheduling for Real-Time Systems: EDF and Related Algorithms." Kluwer, 1998.

[10] A. Grilo, M. Macedo, and M. Nunes, "A Scheduling Algorithm for QoS Support in IEEE 802.11e Networks," *IEEE Wireless Commun. Mag.*, pp. 36–43, June 2003.

[11] P. T. Brady, "A Model for Generating On-Off Speech Patterns in Two-Way Conversation," *Bell Syst. Tech. J.*, pp. 2445–2472, September 1969.

[12] Y.-J. Kim and Y.-J. Suh, "Adaptive Polling MAC Schemes for IEEE 802.11 Wireless LANs Supporting Voice-over-IP (VoIP) Services," *Wireless Communications and Mobile Computing*, pp. 903–916, December 2004.

[13] P. Seeling, M. Reisslein, and B. Kulapala, "Network Performance Evaluation Using Frame Size and Quality Traces of Single-Layer and Two-Layer Video: A Tutorial," *IEEE Communications Surveys and Tutorials*, vol. 6, no. 2, pp. 58–78, Third Quarter 2004. [Online]. Available: http://www.eas.asu.edu/trace

[14] (2006) The Network Simulator, ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns

[15] IEEE 802.11e HCF MAC model for ns-2.28. [Online]. Available: http://www.ece.uci.edu/∼fkeceli/ns.htm

[16] S. Wietholter and C. Hoene, "Design and Verification of an IEEE 802.11e EDCF Simulation Model in ns-2.26," TKN, Tech. Rep. TKN-03-19, November 2003.

[17] D. Qijao and S. Choi, "Goodput Enhancement of IEEE 802.11a Wireless LAN via Link Adaptation," in *Proc. IEEE ICC '01*, June 2001.

[18] *IEEE Standard 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: Further Higher Data Rate Extension in the 2.4 GHz Band*, IEEE 802.11g Std., 2003.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.