On the Distributed Power Saving Mechanisms of Wireless LANs 802.11e U-APSD vs 802.11 Power Save Mode

Xavier Pérez-Costa *, Daniel Camps-Mur and Albert Vidal

NEC Network Laboratories, Kurfürstenanlage 36, 69115, Heidelberg, Germany

Abstract

The integration of the wireless LAN technology in mobile devices such as cellular phones, PDAs or laptops has become a user need due to its popularity in providing high speed wireless Internet access at a low cost. Such devices though should meet users' expectations with regard to QoS, e.g., guarantee a reasonable voice quality when VoIP is used, and power saving efficiency, e.g., standby and calling times should be similar to the ones of cellular phones. The IEEE 802.11e standard, which extends the 802.11 wireless LAN MAC layer with QoS and power saving enhancements, should be the most appropriate solution to address users' wishes in those devices.

In this paper, we focus on the 802.11e functionality likely to be included in mobile devices in the short-term, EDCA for QoS and U-APSD for power saving, and evaluate the performance improvements and associated costs of two possible configurations of U-APSD as compared to the 802.11 power save mode. In addition, the dependency between the QoS and power saving enhancements obtained with U-APSD and the available channel capacity is analyzed considering three different scenarios: 802.11b, 802.11b+g and 802.11g. The evaluation is based on our proposed implementation of U-APSD: Static U-APSD (SU-APSD).

The main conclusions that can be drawn from our results are that U-APSD significantly outperforms the 802.11 power save mode in all considered performance metrics and that the performance enhancements obtained with U-APSD are independent of the available channel capacity.

Key words: Wireless LAN, 802.11, 802.11e, QoS, Power Save Mode, U-APSD

1. Introduction

The wide adoption of the IEEE 802.11 wireless LAN technology by home and business users due to its capability of providing low cost high speed wireless Internet access is driving a strong trend toward the inclusion of this technology in mobile devices like cellular phones, personal digital assistants (PDAs) or laptops. Several challenges though need to be addressed with respect to QoS and power saving limitations to achieve a seamless integration of Wireless LAN in such devices.

The two different access methods provided by the 802.11 standard, DCF and PCF, are expected to be insufficient to meet the QoS requirements of certain key applications as Voice over IP (VoIP). Therefore, an enhancement of the 802.11 MAC layer has been designed, the IEEE 802.11e standard [1], which defines mechanisms to provide QoS differentiation. The IEEE 802.11e standard defines the Hybrid Coordination Function (HCF) to support QoS. Two channel access methods are defined: a contention-based channel access method called the Enhanced Distributed Channel Access (EDCA) and a contention-free channel access referred to as HCF Controlled Channel Access (HCCA). Within a superframe two phases of operation are possible in 802.11e, contention period (CP) and contention-

* Corresponding author.
Email address: xavier.perez-costa@netlab.nec.de (Xavier Pérez-Costa).

free period (CFP). HCCA can be used in both CP and CFP while EDCA can be used only during CP. A thorough overview of the 802.11e QoS enhancements can be found in [2].

Regarding the battery usage efficiency, the intrinsic nature of 802.11, which is based on a shared channel access (CSMA/CA), forces wireless stations to continuously listen to the channel to determine its current status. As a result, a handheld device connected to a network through an 802.11 wireless LAN, would drain its battery after a few hours as opposed to for instance cellular phones that have a standby battery lifetime up to several days. Ideally, mobile devices including the wireless LAN technology should achieve a battery consumption similar to current handheld devices in order to meet users' expectations.

The IEEE 802.11 standard provides a power save mode that reduces the time required for a station to listen to the channel. Although this mechanism significantly alleviates the power consumption problem, a dependency between the data frames MAC downlink delay (AP $\rightarrow$ station) and the Beacon interval is introduced. Consequently, some Beacon interval values can result in downlink delays unacceptable for certain QoS-sensitive applications, e.g., VoIP. An overview of the 802.11 power save mode will be given in the next section.

In order to address some of the power saving issues that arise when a specific QoS is desired, the 802.11e standard includes an optional extension of the 802.11 power save mode defined as Automatic Power Save Delivery (APSD). If the AP supports APSD, 802.11e-capable stations can select the method for the delivery of the frames buffered at the AP between 802.11 power save mode and APSD. A main difference between the 802.11 power save mode and APSD is the introduction of the Service Period (SP) [1] concept. Two types of SPs are possible under APSD: Unscheduled and Scheduled. *Unscheduled* SPs (U-APSD) are defined only for stations accessing the channel using EDCA while *Scheduled* SPs (S-APSD) are defined for both access mechanisms, EDCA and HCCA.

New mobile devices incorporating 802.11e functionality are more likely to include first the *distributed* mechanisms of 802.11e, i.e., EDCA and U-APSD, than the centralized ones, i.e., HCCA and S-APSD. This can be seen for instance in the fact that the Wi-Fi^{TM} Alliance [3] has started first the certification of the Wi-Fi^{TM} Multimedia (WMM^{TM}) and the WMM Power Save^{TM} extensions, which include EDCA and EDCA plus U-APSD functionalities respectively, while certifications for HCCA and S-APSD are being deferred. Based on that, we focus our study on the analysis of the performance enhancements provided by EDCA in combination with U-APSD since it is expected to be the most common functionality implemented in the short-term in 802.11e-capable mobile devices.

In the area of providing QoS in a wireless LAN a lot of research has been done during the last several years, see for example [2], [4]. With respect to the 802.11 power save mode, the infrastructure mode has been studied for instance in [5] and [6] where the main focus was to improve the performance for web-like traffic. Regarding U-APSD, in [7] a modified version of the U-APSD functionality (UPSD) based on an 802.11e draft (Dec'04) was studied where the performance of three different UPSD modes was analyzed in a single scenario with 12 stations and two types of traffic, voice and FTP.

In our previous work [8], we studied the effect of using the 802.11 power save mode combined with the 802.11e EDCA QoS mechanism. In [9] we proposed a MAC power saving algorithm to adaptively control the delay introduced by the 802.11 power save mode for applications requiring QoS guarantees. The paper at hand extends our previous results by i) defining an implementation of U-APSD (SU-APSD) ii) evaluating their QoS and power saving performance as compared to the 802.11 power save mode and iii) analyzing the dependency between the U-APSD performance improvements and the channel capacity available based on the scenario considered: 802.11b, 802.11b+g and 802.11g. The main conclusions that can be drawn from our results are that U-APSD significantly outperforms the 802.11 power save mode in all considered performance metrics and that the performance enhancements obtained with U-APSD are independent of the available channel capacity. In parallel to this work we designed an adaptive algorithm for U-APSD (AU-APSD) which aims to dynamically configure the SU-APSD algorithm presented in this paper based on the information available at the MAC layer [10].

The rest of the paper is structured as follows. In Section 2 an overview of the 802.11 power save mode and U-APSD functionality is given followed by a description of our proposed implementation of U-APSD in Section 3. An evaluation of the performance enhancements obtained by using U-APSD as compared to the 802.11 power save mode and their dependency with the physical layer capacity is provided in Section 4. Finally, Section 5 summarizes the results and concludes the paper.

[1] The Service Period definition will be provided in the next section.

2. Wireless LAN Distributed Power Saving Mechanisms

2.1. *802.11 Power Save Mode*

The IEEE 802.11 standard defines two independent power management mechanisms, depending on whether the infrastructure or ad hoc mode is used, that allow mobile stations to enter a power saving mode of operation where they turn off their receiver and transmitter to save power. In our study we focus on the infrastructure power management scheme during the contention period as the most relevant case for mobile devices including wireless LAN capabilities. Note that nowadays most of the wireless LAN deployments use the infrastructure mode with the access arbitrated by the distributed coordination function (DCF).

2.1.1. *Functionality Description*

In infrastructure mode, Access Points (APs) maintain a power management status for each currently associated station that indicates in which power management mode the station is currently operating. Stations changing power management mode inform the AP of this fact using the power management bits within the frame control field of transmitted frames.

The AP buffers unicast and multicast data frames destined to any of its associated stations that are in power save mode. If an AP has buffered frames for a station, it will indicate it in the traffic indication map (TIM) which is sent with each Beacon frame. During the association process every station is assigned by the AP an Association ID code (AID). The AID indicates with a single bit in the TIM whether frames are buffered for a specific station. APs can also have an aging function that deletes buffered traffic when it has been buffered for an excessive period of time.

Stations request the delivery of their buffered frames at the AP by sending a Power Save Poll (PS-Poll). A *single* buffered frame for a station in power save mode is sent by the AP after a PS-Poll has been received from a station. Further PS-Poll frames from the same station are acknowledged and *ignored* until the frame is either successfully delivered, or presumed failed due to the maximum number of retries being exceeded. This prevents a retried PS-Poll from being treated as a new request to deliver a buffered frame.

Stations operating in 802.11 power save mode reduce their power consumption by switching to a low power state during the inactivity periods. From the protocol's perspective a WLAN station can be in one of these two different power states:

– *Awake*: station is fully powered.
– *Sleep*: station is not able to transmit or receive and consumes very low power.

While being in power save mode a station awakes for listening a beacon once every n beacons, where n is an integer ≥ 1. The *Listen Interval* value used by a station is communicated to the AP in its association request.

Each station learns through the TIM in the beacon whether the AP buffered any frames destined to them while they were in the sleep state. If a station sends a PS-Poll to retrieve a buffered frame, the AP can respond acknowledging (ACK) it or sending directly the data frame. Current implementations though, consider only the possibility of sending an ACK after the reception of a PS-Poll. The main reason for this is the technical difficulty of having to find in the power save buffer of the AP a frame addressed to a certain station and send it in a SIFS time interval. In the event that neither an ACK nor a data frame is received from the AP in response to a PS-Poll frame, then the station retries the sequence, by transmitting another PS-Poll frame.

In the frame control field of the frame sent in response to a PS-Poll, the AP sets a bit labeled More Data if there are further frames buffered for this station. The station is required to send a PS-Poll to the AP for each data frame it receives with the More Data bit set. This ensures that stations empty the buffer of the frames held for them at the AP.

Mobile stations should also awake at times determined by the AP, when broadcast(BC)/multicast(MC) frames are to be transmitted. This time is indicated in the Beacon frames as the delivery traffic indication map (DTIM) interval. If ReceiveDTIM is true a station must awake at every DTIM.

Finally, note that the usage of the 802.11 power save mode functionality by a station does not imply that frames sent from the station to the AP are delayed until the next beacon is received, i.e., mobile nodes wake up whenever they have data to send and follow the regular 802.11 DCF transmission procedure.

2.1.2. *Example of 802.11 power save mode operation*

Figure 1 illustrates the infrastructure power save mode procedure during the contention period assuming a listen interval for station 1 (STA 1) and station 2 of two beacon intervals, and a DTIM period of 5 beacon intervals. In the figure, the process for delivering broadcast (BC) and multicast (MC) frames is shown first. A beacon is sent by the AP containing the DTIM, station 1 with ReceiveDTIM set to true powers-up its receiver

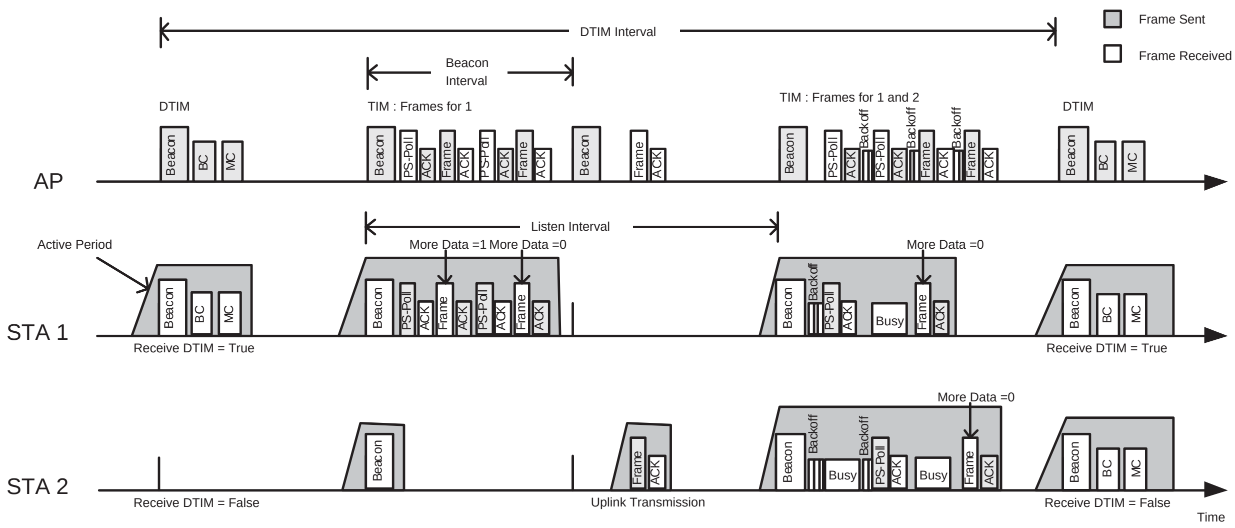

Fig. 1. Example of 802.11 power save mode operation

for the reception of the broadcast and multicast packets while STA 2 does not because ReceiveDTIM is set to false. Next, both stations awake, as required by their listen interval, to receive a beacon. STA 2 switches to the sleep mode since no packets are buffered in the AP for that station as indicated in the TIM for its AID. On the other hand, STA 1 sends a PS-Poll to ask for a frame buffered in the AP because its bit in the TIM is true following the regular DCF access procedure. After receiving the corresponding data frame, since the More Data bit was set to true, a new PS-Poll is sent by the same station.

In the following beacon the stations should not awake due to their listen interval but STA 2 awakes for sending a data frame to the AP. As mentioned before, the power save mode does not imply any restriction for awaking when a packet needs to be sent in the uplink. The next beacon, listened by both stations, results in both having to transmit a PS-Poll. They perform the backoff procedure to try to avoid a collision and STA 1 gets a shorter backoff time and thus, transmits first its PS-Poll and STA 2 defers. After receiving the PS-Poll and sending an ACK, the AP has to defer because of the transmission of the PS-Poll of STA 2. Finally, the AP transmits the frames for both stations in the order they were requested. In the next beacon including DTIM, STA 2 wakes up for that beacon even if ReceiveDTIM is false since it has to awake for some of them.

For further details about the power save mode operation the reader is referred to [11].

2.2. *Unscheduled Automatic Power Save Delivery (U-APSD)*

Unscheduled Automatic Power Save Delivery (U-APSD) is the distributed APSD method defined in 802.11e to improve the QoS provided to stations accessing the channel using the EDCA mechanism. The main idea behind the U-APSD design is the usage of data frames sent in the uplink by stations (STA → AP) as indications (*triggers*) of the instants when the power saving stations are awake. When such an indication is received at the AP from a power saving station, the AP takes advantage of it for delivering any data frames buffered while the station was in sleep mode. Because of its specific functionality, this method is specially suited for bi-directional traffic streams even though it provides alternative methods for its usage in other cases. APSD has been designed such that it provides backwards compatibility to legacy terminals implementing the 802.11 power save mode only. Therefore, all new functionality has been built on top of the already available 802.11 power save mode one re-using as much as possible without altering the original power save mode specification. In the following we describe in detail the U-APSD functionality assuming a basic knowledge of the 802.11 power save mode and of the EDCA mechanism of 802.11e. Please see [2] for a thorough overview of 802.11e.

2.2.1. *Functionality Description*

The main difference between the power saving method defined in the 802.11 standard and APSD is that with APSD a station is awake during a Service Period (SP) instead of being awake from the transition to the awake state for receiving a Beacon until the return to the sleep state after acknowledging receipt of the last frame buffered at the AP through PS-Polls.

An *unscheduled SP* begins when the AP receives a *trigger frame*, QoS data or QoS Null, from a station and ends when the station receives a QoS Data or QoS Null frame indicating the end of the service period (EOSP). QoS Null frames are the substitutes in U-APSD of PS-Polls in 802.11 power save mode to allow a station to request the delivery of the frames buffered at the AP even if a station has no data frame to transmit in the uplink. This enables the usage of U-APSD by applications which do not generate uplink traffic often enough to meet the downlink QoS application requirements.

Four Access Categories (AC) are defined in EDCA (AC_VO, AC_VI, AC_BE, and AC_BK) corresponding to the applications for which they are intended, i.e., voice, video, best effort and background. Each AC of a station can be configured separately to be delivery/trigger-enabled. When a station has an AC configured as *delivery-enabled*, the AP is allowed to use EDCA to deliver traffic from the AC to a station during an unscheduled SP triggered by a station. When a station AC is *trigger-enabled*, frames of subtype QoS Data and QoS Null from the station, that map to the AC, trigger an unscheduled SP if one is not in progress.

During a SP one or more data frames of delivery-enabled ACs might be delivered by the AP to a station up to the number of frames indicated in the *Maximum Service Period Length (Max_SP_Length)* following the rules of an acquired transmission opportunity. The maximum service period length is a field contained in the QoS Info field filled by the station at association. In each directed data or management frame associated with delivery-enabled ACs sent to a station, the More Data (MD) bit indicates that more frames are buffered for the delivery-enabled ACs. For all frames except for the final frame of the SP, the EOSP subfield of the QoS control field of the QoS data frame shall be set to 0 to indicate the continuation of the SP.

In order to guarantee backward compatibility of legacy stations that do not support APSD, the procedure of the AP to assemble the traffic indicator map (TIM) has been modified in such a way that, if at least one of the ACs is non delivery-enabled, it indicates the buffer status *only* of the ACs non delivery-enabled. Note that, in this case, it means that the Beacon will not indicate whether frames of ACs delivery-enabled are buffered. Only in the case that *All* ACs of a station are delivery-enabled the TIM indicates the buffer status of delivery-enabled ACs.

The configuration at the AP of the different ACs per station as delivery/trigger-enabled can be performed either at association time or through the usage of the Traffic Specification element info field (TSPEC) of the Add Traffic Stream frames (ADDTS). The option of configuring U-APSD at association has the advantage of not having to generate ADDTS frames each time a new communication starts for each of the ACs for which we would like to use U-APSD, but because the TIM in the Beacon might not indicate whether frames are buffered at the AP, it might require to periodically send QoS Null frames if no triggers are sent in order to learn the actual buffer status of the AP. On the other hand, if U-APSD is configured through ADDTSs each time a new application starts which requires its usage, then the stations can rely on the TIM indication of the Beacons to be informed about new traffic at the AP, enter into U-APSD mode through ADDTS when a communication starts and revert back to 802.11 power save mode when it ends.

2.2.2. *Example of U-APSD operation*

In Figure 2 an example illustrating the U-APSD procedure is provided. Several frame exchanges between the AP and a station are shown considering a station that has the following configuration: AC_VO and AC_VI configured as trigger- and delivery-enabled and AC_BE and AC_BK configured as neither delivery- nor trigger-enabled (i.e., 802.11 power save mode). The maximum service period length configured is *all* frames in order to retrieve from the AP all buffered frames in each service period.

In the first active period of the station, the station wakes up to receive the beacon sent by the AP and, since the TIM vector indicates that the AP has buffered packets for this station (1 AC_BE MSDU and 1 AC_BK MSDU), the station sends a PS-Poll in order to retrieve the buffered frames. Note that since some ACs are not configured to use U-APSD the TIM only refers to the ACs configured to use 802.11 power save mode, i.e., AC_BE and AC_BK in this particular case. As recommended in the 802.11e specification the EDCA rules to be followed to send a PS-Poll are the ones corresponding to the AC_BE category, therefore, the station performs the EDCA access to send the PS-Poll and then waits for an AC_BE or AC_BK frame to be delivered. After receiving the first frame, a new PS-Poll is sent because the the More Data bit indication signals that there are still buffered frames at the AP. Finally, the More Data bit indicates that there are no further frames

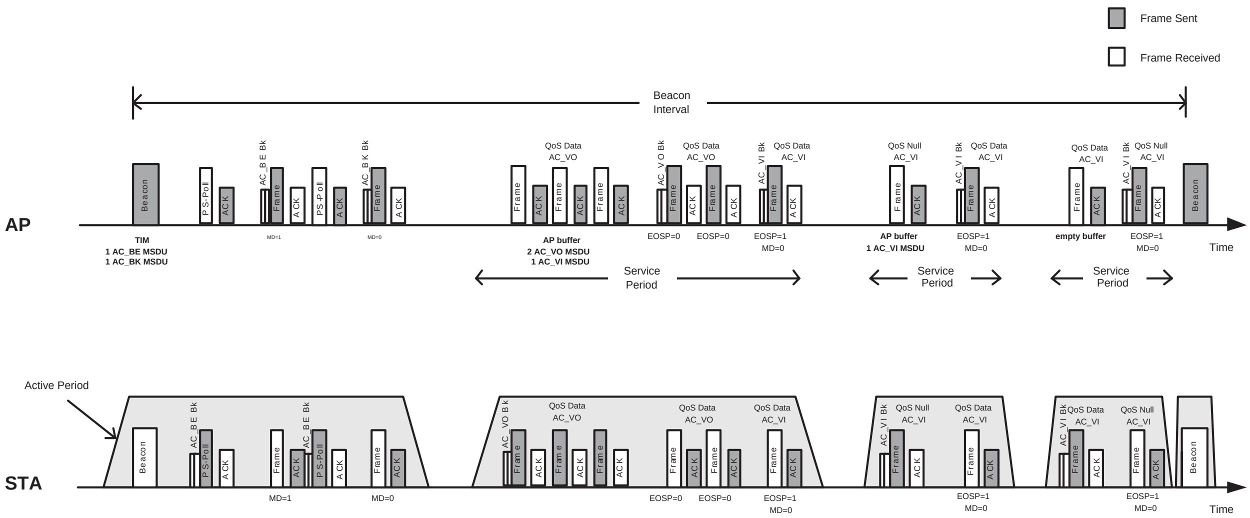

Fig. 2. Example of U-APSD operation

buffered at the AP and therefore the station goes back to sleep.

In the second active period, the station wakes up because it has to send an AC_VO QoS data frame in the uplink and transmits up to three frames as allowed by its TXOPLimit. The first AC_VO frame received by the AP acts as a trigger and starts an *unscheduled service period*. The AP starts then delivering to the station the two AC_VO data frames and the one AC_VI data frame it had buffered following the EDCA rules corresponding to the AC of the delivered frames. All the frames in the service period are sent with EOSP set to 0 except the last one that has EOSP set to 1 which ends the service period. Because the station configured the maximum service period length to be *all* frames the AP will deliver all the buffered frames during this service period.

In the third active period shown in the figure, the station wakes up because an internal U-APSD algorithm decides that a SP needs to be started in order to fulfill the QoS requirements of the downlink traffic. The example shows how the station sends an AC_VI QoS Null frame to trigger an unscheduled SP and receives an AC_VI frame from the AP.

In the fourth active period, the station wakes up because it has an AC_VI data frame to be sent and starts an unscheduled service period but the AP has no data to deliver and therefore the AP ends the service period by sending a QoS Null frame with EOSP set to 1.

Finally, the station wakes up to receive the beacon and since the TIM indicates that no packets are buffered at the AP for the ACs configured to use 802.11 power save mode, the station goes back to sleep again.

2.2.3. *Advantages of U-APSD with respect to 802.11 power save mode*

Three main advantages are introduced with U-APSD with respect to 802.11 power save mode. These advantages will be evaluated in Section 4.

- The first advantage is the possibility of providing different QoS treatment to the ACs with U-APSD enabled since, because of the possibility of generating triggers at any point of time for a specific AC, the delay introduced by U-APSD for an AC can be limited based on the expected QoS needs of this AC. Additionally, triggers get access to the channel according to the priority of the frames intended to be retrieved as compared to the PS-Polls which by default use AC_BE as indicated in [1].
- The second advantage is the reduction of the overhead required to retrieve frames from the AP thanks to the usage of data frames as triggers. This is specially relevant for symmetric applications like VoIP, because then less QoS Null frames are required and therefore, the PS-Poll overhead required by the 802.11 power save mode is significantly reduced.
- The third advantage is again an important reduction of the overhead required to retrieve frames from the AP depending on the maximum service period length (Max_SP_Length) per trigger. While for 802.11 power save mode a signaling frame (PS-Poll) is required for obtaining *each single* frame buffered at the AP this is not necessary for U-APSD which delivers up to Max_SP_Length frames per trigger.

3. Proposed U-APSD Implementation

The U-APSD specification provided in the IEEE 802.11e standard introduces different mechanisms to control the QoS and power saving provided to a station for each different AC. The specific implementation of these mechanisms to actually deliver the desired QoS is though, as usual, left open to allow differentiation between vendors. In the following we describe our proposed implementation of an U-APSD solution that we will refer as *Static* U-APSD (SU-APSD).

For the design of the U-APSD algorithm we assumed that the EDCA QoS parameters are properly set by a QoS algorithm at the AP to provide the required bandwidth and delay guarantees for the applications. Therefore, the objectives of our algorithm design are:

- Limit the MAC downlink delay introduced by the U-APSD mechanism to U-APSD-enabled ACs to a value acceptable for the expected QoS needs of the applications using such ACs
- Minimize the required signaling overhead introduced in the channel due to the U-APSD usage

Depending on the expected delay requirements of an AC with respect to the beacon generation interval of the AP we can differentiate between two cases that require different U-APSD solutions. The simplest case is the one where the expected delay requirement of the applications using a certain AC is above the beacon generation interval of the AP. Taking into account that the beacon interval usually configured is 100ms, this would typically be the case of the ACs AC_BE and AC_BK. In this case, if *all* the ACs of a station are configured as delivery-enabled, the station just needs to set the Listen Interval such that according to its value the expected delay requirements of the ACs whose delay requirements are above the beacon interval are fulfilled. By doing this, the default U-APSD functionality of the station will generate a QoS Null to start a SP after being informed by the TIM bit in the beacon that frames have been buffered at the AP and thus, the delay requirement of those ACs will be fulfilled. In the rest of the paper we will refer to this U-APSD mode as *U-APSD Mode Default*.

On the other hand, if the expected delay requirements of the applications using a certain AC are below the beacon generation interval of the AP, we need a way to guarantee that triggers will be generated to meet the delay requirements even if the application running at the station does not generate enough data frames (triggers).

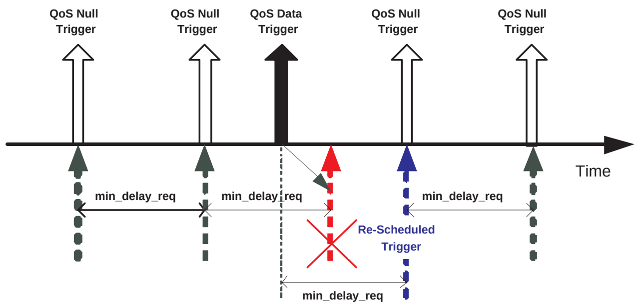

Fig. 3. U-APSD Re-scheduling example of operation

In order to limit the MAC downlink delay introduced by U-APSD (AP → station), our proposed solution is to schedule at fixed time intervals the generation of QoS Null frames to bound the maximum delay introduced by U-APSD. The time intervals could be set based on some information provided to the MAC layer depending on the application or simply fixed based on an assumed maximum delay requirement per AC. In the rest of the paper we will refer to this U-APSD mode as *U-APSD Mode QoS*.

In the U-APSD Mode QoS, to avoid the introduction of useless overhead in the channel due to the generation of unnecessary QoS Null frames, we propose to re-schedule the QoS Null frame generation of U-APSD trigger-enabled ACs each time that a new data frame of such an AC is prepared to be sent. The re-scheduling of the generation of QoS Null frames due to a QoS Data frame of a trigger-enabled AC is depicted in Figure 3. In this way, we guarantee that a QoS Null frame will be generated to trigger a SP only after reaching the maximum delay we would like to allow for U-APSD trigger-enabled ACs since the last transmission of a data frame of such an AC. Note that if for a certain station more than one AC is U-APSD trigger-enabled, we only need to schedule the transmission of QoS Null frames for the one with the minimum delay requirements because the requirements of other ones would be fulfilled by the most stringent one.

Once a trigger is received at the AP, we start delivering first the frames of the highest priority delivery-enabled AC and continue until all buffered frames of delivery-enabled ACs for a certain station have been delivered or the maximum service period length has been reached. Since, due to the maximum service period length, it could happen that a SP concludes before all the frames of a certain delivery-enabled AC had been delivered to a station, we use in our algorithm the More Data information provided in the data frames to decide whether a new SP has to be started after receiving the EOSP indication.

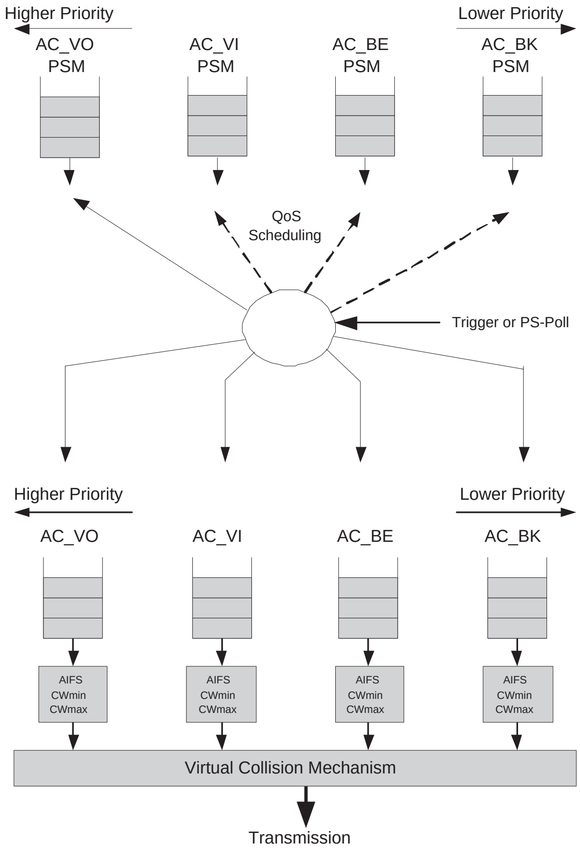

Fig. 4. U-APSD queueing architecture

The maximum service period length is another parameter that needs to be configured for U-APSD. In our proposed implementation we configure the maximum service period length (Max_SP_Length) such that it permits sending the maximum number of frames allowed by the 802.11e specification, i.e., all. We have chosen this option for three main reasons. First, because of power saving efficiency since the more frames sent in a row after getting access to the channel, the less contention for the channel required and therefore the SPs can complete faster. Second, because by retrieving the frames from the AP as soon as possible we maximize the chances that the traffic of power saving stations meets its delay requirements. Note that even if the Max_SP_Length has no limit, the maximum time that a station can keep control of the channel will continue to be bounded by the TXOP limit. Third, because the less SPs required to retrieve the frames from the AP the less signaling load introduced in the channel by U-APSD.

Finally, to avoid that high priority traffic gets dropped in congestion conditions at the AP due to the low retrieval rate of low priority one, we propose to have separate power save mode queues at the AP (physical or logical) for each AC to guarantee that no high priority traffic is dropped because of having the power save mode queue full of low priority traffic. Figure 4 depicts the resulting queueing architecture at the AP.

Based on the different considerations explained above we designed the Static algorithm for U-APSD (SU-APSD) shown in Algorithm 1 for the ACs in U-APSD Mode QoS.

Algorithm 1 Static U-APSD algorithm for the Mode QoS that soft bounds the MAC downlink delay to the minimum delay value configured for the trigger-enabled ACs of a station (SU-APSD)

Trigger interval selection
for All Trigger-Enabled ACs with an active session **do**
  if $delay\_req[AC_i] < min\_delay\_req$ **then**
    $min\_delay\_req = delay\_req[AC_i]$
  end if
end for

Start trigger generation
Generate the first trigger when an active session generating traffic of a trigger-enabled AC is detected
$trigger\_time = current\_time + min\_delay\_req$

Periodic trigger generation
if $trigger\_time = current\_time$ **then**
  $trigger\_time = trigger\_time + min\_delay\_req$
end if

Trigger re-scheduling
if A frame is sent in the uplink of a trigger-enabled AC **then**
  Cancel the pending Trigger and schedule a new one at $trigger\_time = current\_time + min\_delay\_req$
end if

---

4. Performance Evaluation & Discussion

In this section we evaluate the performance of our proposed algorithm for U-APSD as described in Section 3 with respect to 802.11 power save mode. The analysis has been performed via simulation. We extended the 802.11 libraries provided by OPNET v10.0 [12] to include the EDCA QoS mechanisms of 802.11e, our proposed U-APSD algorithm and 802.11 power save mode.

EDCA-TXOP durations are configured for all ACs to allow the transmission of one data frame after gaining access to the medium. The RTS/CTS mechanism has not been enabled to avoid its influence over the mechanisms being studied. The beacon interval used is 100ms and the listen interval configured for the power saving stations is 1. The maximum service period length (Max_SP_Length) configured allows to send all delivery-enabled frames for a certain station.

In the analysis, the performance of two different U-APSD configurations will be compared to the standard 802.11 power save mode (StdPSM) one. First, we consider a configuration where only the ACs categories which are supposed to require QoS guarantees, i.e., AC_VO and AC_VI, are configured to use U-APSD while AC_BE and AC_BK use 802.11 power save mode. This case will be referred as *UAPSD+StdPSM*. Second, we consider a configuration where *all* ACs are configured to use U-APSD but with AC_VO and AC_VI using U-APSD Mode QoS and AC_BE and AC_BK U-APSD Mode Default. This case will be referred as *All UAPSD*. The delay requirements set for the ACs using U-APSD Mode QoS are 40 and 60ms for AC_VO and AC_VI respectively. We have chosen the delay requirements for AC_VO and AC_VI 20ms above the frame generation interval of the traffic sources used (20 and 40 ms respectively) to model the fact that in general the trigger generation interval will not be matching perfectly the frame generation interval but will rather have a value to cover the most common range.

EDCA	**AIFS**	**CWmin**	**CWmax**
AC_VO	2	31	63
AC_VI	2	63	127
AC_BE	3	127	1023
AC_BK	7	127	1023

Table 1
EDCA configuration for the different ACs

Since our focus is on the differences in the performance between the power saving mechanisms when being used together with EDCA, we assume a fixed configuration of the 802.11e EDCA QoS parameters based on the 802.11e document recommendation [1]. The parameters used are detailed in Table 1.

The configuration of the applications used is detailed below:

– AC_VO: G.711 Voice codec with silence suppression. Data rate: 64kbps. Frame length: 20ms. Talk spurt exponential with mean 0.35s and silence spurt exponential with mean 0.65s.
– AC_VI: MPEG-4 real traces of the movie 'Star Trek: First Contact' obtained from [13]. Target rate: 64kbps. Frame generation interval: 40ms.
– AC_BE: Web traffic. Page interarrival time exponentially distributed with mean 60s. Page size 10KB plus 1 to 5 images of a size uniformly distributed between 10KB and 100KB.
– AC_BK: E-mail. Send interarrival time exponentially distributed with mean 120s. Receive interarrival time exponentially distributed with mean 60s. Size exponentially distributed with mean 100KB.

The length of the simulations performed is 300 seconds with a warm-up phase of 30 seconds. The number of seeds used to obtain each value in the graphs has been chosen such that the 95% confidence interval of a value in a certain point does not overlap with the 95% confidence interval of any other value.

In the evaluation we study the impact of increasing the number of stations over the MAC throughput and delay, the power saving efficiency and the resulting power saving costs. The experiment starts with a scenario of four wireless LAN stations where each station is configured to send and receive traffic from their corresponding pair in the wired domain of a different AC, i.e., one station sends and receives AC_VO traffic, a second one sends and receives AC_VI traffic and so on. Since the two power saving mechanisms subject of study degrade the performance mainly in the downlink direction we focus on the downlink results (AP $\rightarrow$ STA) which is the performance bottleneck of the system.

4.1. *QoS Differentiation*

Figure 5 shows the downlink data throughput performance of the different power saving configurations considered, *All UAPSD*, *UAPSD+StdPSM* and *StdPSM*, for each one of the four ACs. As expected, when there is no congestion in the wireless channel (below 7 stations per AC), similar results are obtained within each AC for all the power saving configurations in throughput terms since the signaling load introduced by the different mechanisms has no major impact. In this region, the differences between ACs are mainly due to the load generated by their applications and their EDCA QoS settings. However, as the number of stations increases the channel starts to get congested (above 8 stations per AC) and the difference in performance between the three power saving configurations becomes clearly noticeable.

The *All UAPSD* configuration (dotted line) is the one presenting the best performance for all ACs followed by

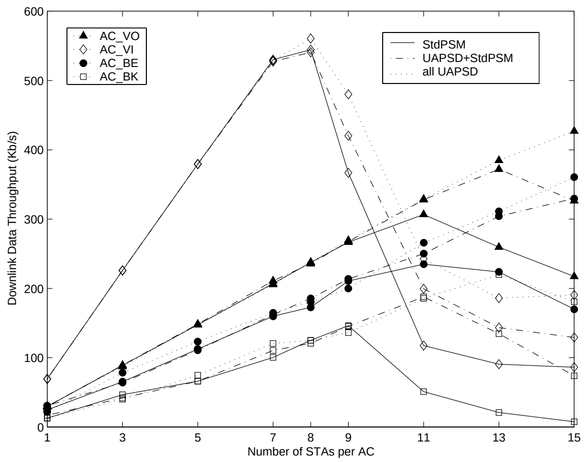

Fig. 5. Impact of the number of stations on the MAC downlink throughput, All UAPSD vs UAPSD+StdPSM vs StdPSM

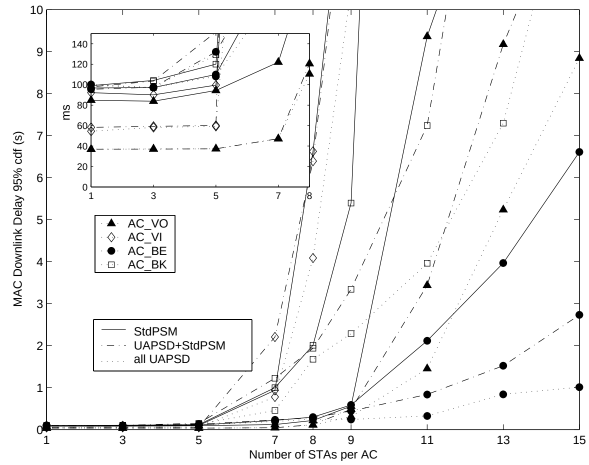

Fig. 6. Impact of the number of stations on the MAC downlink delay, All UAPSD vs UAPSD+StdPSM vs StdPSM

the *UAPSD+StdPSM* one (dot dashed line) and finally the *StdPSM* configuration (solid line). The reason for the performance enhancement observed with configurations using U-APSD with respect to the one using only StdPSM is twofold. On the one hand, the fact that with U-APSD the transmission of 'requests' (triggers) to get the frames buffered at the AP are not synchronized with the beacon reception reduces the probability of collision in the channel. On the other hand, the faster retrieving rate of frames buffered at the AP by the data frames sent in the uplink acting as triggers or scheduled QoS Nulls transmissions reduces the number of frames buffered at the AP and thus, increases the number of stations required to produce a buffer overflow. The *All UAPSD* configuration outperforms the *UAPSD+StdPSM* one because the ACs AC_BE and AC_BK can benefit also of the U-APSD enhanced power saving functionality if U-APSD Mode Default is used.

The best example to illustrate the benefits of the U-APSD functionality in our scenario is AC_VO (VoIP application) since U-APSD is specially suited for symmetric applications. In this case, because the frame generation rate is the same in the uplink and in the downlink, QoS Nulls need to be generated to retrieve the frames from the AP only when there is a silence period at the AC_VO stations and, as a result, a larger number of stations can be reached before the downlink data throughput starts to decrease. Note that our architecture mantains a different power save buffer for each AC. Therefore, the point where the power save buffer of each AC gets full is independent from the one of the other ACs.

The corresponding results with respect to the MAC downlink delay are shown in Figure 6. The results perfectly match the throughput ones in congestion conditions (above 5-7 stations per AC). *All UAPSD* and *UAPSD+SdtPSM* present a significant performance improvement as compared to StdPSM. A clear differentiation though is observed in no congestion conditions that did not appear before. The reason for the significant lower delay for AC_VO and AC_VI in the UAPSD configurations is that triggers are generated every 40ms and 60ms, respectively, as compared to 802.11 power save mode where the Beacon is sent every 100ms. Additionally, the lower probability of collision in the UAPSD configurations results also in an improvement for the AC_BE and AC_BK traffic. These results clearly show the differentiated downlink delay that can be provided with U-APSD based on the AC separation as compared to 802.11 power save mode. Therefore, the U-APSD power saving method is better suited to meet the QoS requirements of users operating mobile devices with Wireless LAN capabilities than 802.11 power save mode.

4.2. *Power Saving Efficiency*

The main objective of the two power saving mechanisms analyzed in this study is to reduce the power consumption of the stations. In the previous section we have analyzed which scheme is better to provide QoS differentiation according to each AC need. In this section we study the differences in power consumption between both methods and whether the QoS differentiation advantages obtained with the UAPSD configurations have a cost in power saving efficiency terms.

The power saving model used for the evaluation has been derived based on current WLAN cards/chipsets available on the market and consists of four states: Sleep, Listen, Reception and Transmission. The *Clear Channel Assesment* (CCA) function used in our analysis is CCA mode 1, or energy threshold. Using this CCA function when the station transitions from sleep to listen state it can realize whether there is an ongoing transmission or not by sensing the level of energy present in the medium.

Based on this model two performance metrics are obtained for the evaluation of the power saving methods. In order to obtain results independent of a specific WLAN card/chipset model, we compute the percentage of time spent during an active session in each state by the stations per AC. This generic metric allows us to derive conclusions regarding the relationship between the time spent in each different power state and its dependency with the chosen power saving method. On the other hand, in order to provide an example of the actual power required depending on the power method used, we particularize the results obtained with the generic metric by translating them into *mA* based on the information provided in the product datasheet of a common PCMCIA WLAN card [14]. The power consumption values used are shown in Table 2 [2].

Cisco Aironet^{TM}	**Sleep**	**Listen**	**Reception**	**Transmission**
Power (mA)	15	203	327	539

Table 2
Power consumption at each different state of a popular PCMCIA card

Figure 7 shows the results obtained with the generic metric, percentage of time that the stations of the different ACs spend in each power saving state, for the *All UAPSD*, *UAPSD+StdPSM* and *StdPSM* configurations. Several general trends can be observed valid for the results of the three different configurations. First, when the load in the Wireless LAN network is low the stations spend their time mostly in the sleep state while the rest of the time is spent in each of the listen, receive and transmit states according to the characteristics of the application used. For instance the AC_VO stations whose application is VoIP spend a similar time on the receive and transmit state while in the AC_VI case more time is spent in receiving rather than in transmitting since the application is a downlink video stream.

Second, when the congestion in the network increases the time spent in the sleep state decreases while the time spent in the listen state raises becoming the main contributor of the total power consumption. The reason for this is intrinsic to the CSMA/CA protocol, the larger the number of stations trying to access the channel the larger the time required to get access to it due to the contention process. As a result, the stations need a longer time to send a trigger and the Access Point in turn needs also more time to deliver the buffered frames. When this time gets above the interarrival time of the downlink traffic being received in the Access Point the stations never go back to the sleep state. In this particular experiment for example the AC_VI stations suffer from this effect when the number of stations of each AC gets above 7, from that point on the power consumption is the same that they would have if they would be continuosly active (active mode).

Third, the *All UAPSD* configuration outperforms the *UAPSD+StdPSM* one for all ACs and this one in turn outperforms the *StdPSM* configuration also for all ACs. The main reason for that, as it has been explained before, is the higher probability of collision in the channel when the 802.11 power save mode is used due to the synchronization of the transmission of the triggers (PS-Polls) by the stations after the reception of the beacon. Additionally, the reduction of the amount of triggers required by the U-APSD approaches to retrieve the frames buffered at the AP further improve the power saving performance achieved with the U-APSD method.

Fourth, the time spent in the sleep state by the AC_VO and AC_VI stations decreases considerably faster than for the AC_BE and AC_BK stations. The significantly larger number of frames transmitted and received by the AC_VO and AC_VI stations due to their applications and their QoS requirements is the reason for this, resulting in those stations being awake almost all the time in highly congested scenarios.

Finally, if we group the AC_VO and AC_VI stations as the ones with ”intensive” traffic and the AC_BE and AC_BK stations as the ones with ”low” traffic, we can clearly observe the effect of the higher priority QoS settings of AC_VO over AC_VI and of AC_BE over AC_BK with respect to the amount of time spent in the listen state. The larger the priority the smaller the time required to transmit or receive a frame and thus, the time required to be spent in the listen state becomes smaller which obviously results in a larger power saving.

If we now translate the results shown in Figure 7 to actual power consumption based on the values of a popular Wireless LAN card as indicated in Table 2 we obtain the results depicted in Figure 8. This figure summarizes the previous results by showing the power spent by the stations of each AC for each one of the power saving configurations under study. The values

[2] For the sleep mode we used the value of a previous model of a Cisco PCMCIA card (Cisco Aironet 350) since no information was available for the current one

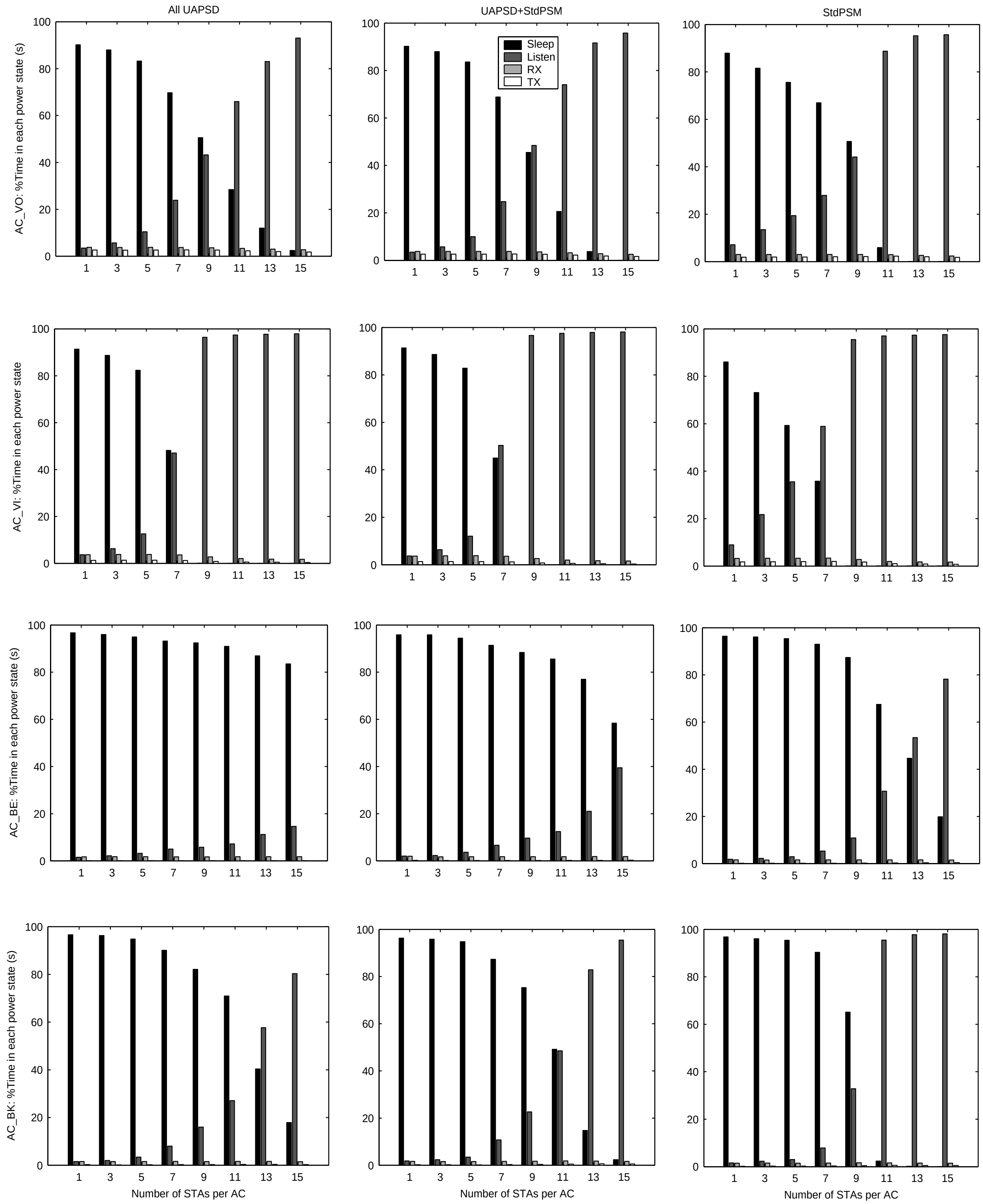

Fig. 7. Power saving states distribution depending on the number of stations per AC for the *All UAPSD*, *UAPSD+StdPSM* and *StdPSM* cases

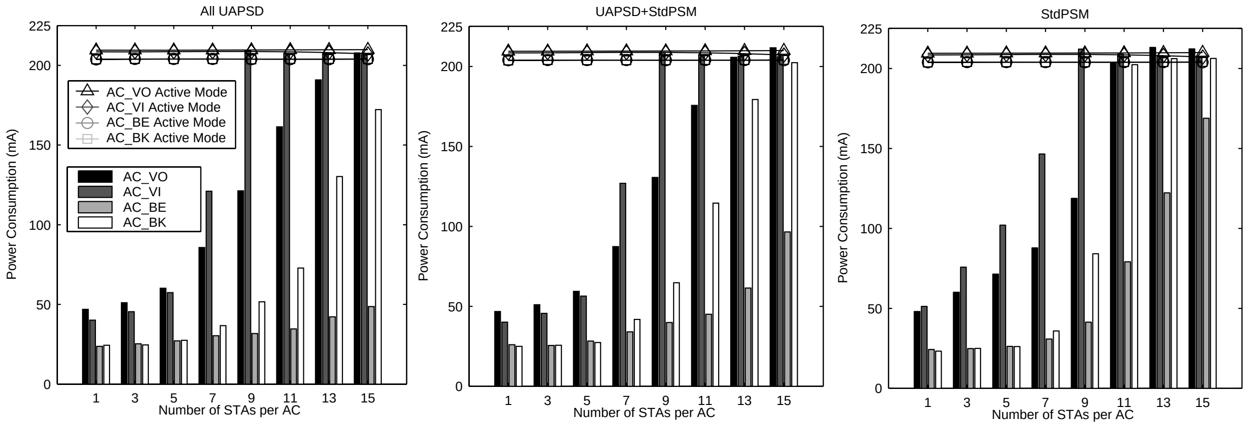

Fig. 8. Impact of the number of stations on the power consumption per AC for the *All UAPSD*, *UAPSD+StdPSM* and *StdPSM* cases

obtained confirm the conclusions drawn with the generic metric based on the amount of time spent in each power state and show that while the power consumption of the different power saving methods is highly dependent on the QoS settings and congestion in the network, it is fairly independent of these factors in the active mode case. The reason for this difference is that in the power saving methods case the power consumption in the sleep state is one order of magnitude lower than in any of the other states while in the active mode case all states are of the same order of magnitude. Since the amount of time spent in the sleep state depends on the amount of time required to be in the listen state, which varies significantly based one the QoS settings of each AC and the congestion in the network, the power consumption of the power saving methods is determined by these two factors.

These results clearly demonstrate that it is worth to use a power saving method during an active session instead of directly switching to active mode but the amount of power saving that can be achiveved strongly depends on the congestion of the network.

4.3. *Power Saving Cost*

In the previous sections we have seen the QoS and power saving improvements obtained with U-APSD as compared to 802.11 power save mode. In this section we study which are the costs of these performance enhancements in terms of required signaling load and Wireless LAN channel usage efficiency.

Figure 9 shows the signaling load introduced by the stations and AP in all considered configurations due to the power saving mechanisms, i.e., PS-Poll or QoS Null frames sent over the wireless channel. As in the power saving efficiency section, some general patterns can be identified depending on the power saving mechanism used by each station of a specific AC.

The first pattern corresponds to the QoS Null signaling generated by the AC_VO and AC_VI stations of the *All UAPSD* and *UAPSD+StdPSM* configurations which increases linearly according to the number of stations up to a maximum and then decreases again. The reason for such effect is that for these two configurations the AP delivers all the buffered frames of the AC_VO and AC_VI stations upon receiving a trigger from them based on the Service Period length we configured. Therefore, when the network gets congested, the time that the AP needs to deliver a frame in a Service Period becomes longer what in turn increases the chances that a new frame addressed to the station arrives at the AP within an ongoing service period. This results in an effective increase of the duration of the service periods, that obviously translates in an increase of the stations' listen time and in a reduction of the number of triggers generated. In the extreme case where a station is all the time active, no trigger is generated because the station is always in the same service period. Note that this is a very interesting property of the U-APSD mechanism since it adapts the amount of signaling introduced in the channel to the level of congestion in the network.

The second pattern is the one corresponding to the PS-Poll signaling generated by all the stations in the *StdPSM* configuration case and by the AC_BE and AC_BK stations of the *UAPSD+StdPSM* one. When using the 802.11 power save mode the stations have to send a PS-Poll for every single frame they want to retrieve from the AP and thus the number of PS-Poll transmissions increases linearly according to the num-

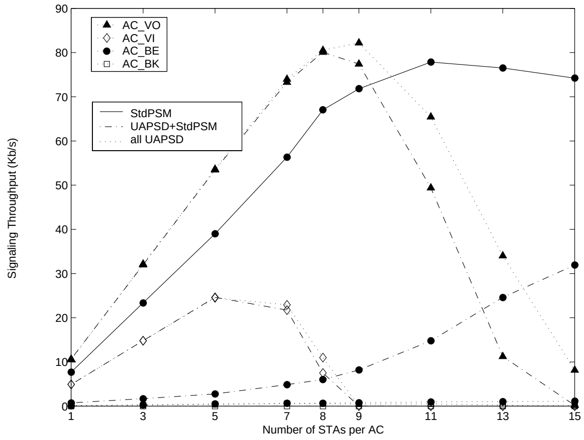

Fig. 9. Impact of the number of stations on the signaling load

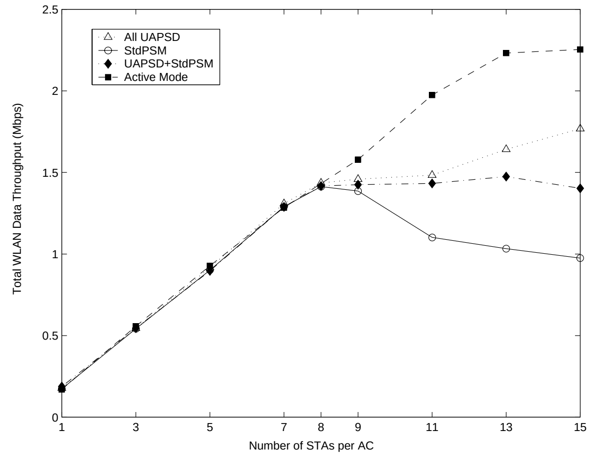

Fig. 10. Impact of the number of stations on the Wireless LAN channel

ber of stations using *StdPSM*. An important point to note in this case is the fact that while in the U-APSD case the stations and the AP use the AC QoS settings for the transmission of data and trigger frames (QoS Nulls) in the *StdPSM* one the stations send the PS-Poll frames using AC_BE as recommended in the 802.11e standard [1].

The last pattern corresponds to the QoS Null signaling generated by the AC_BE and AC_BK stations in the *All UAPSD* configuration case. These stations are configured to use the UAPSD mechanism in mode Default and thus, only generate a trigger upon the reception of a Beacon frame with a specific bit set in the TIM vector. This results in an very low signaling load generated because the web and mail traffic comes in bursts that can be retrieved from the AP using a single trigger.

Comparing the overall signaling load introduced in our experiments by the different configurations, *All UAPSD*, *UAPSD+StdPSM* and *StdPSM*, we can see that the U-APSD configurations introduce more signaling than StdPSM when the network is lightly loaded but less when the network gets congested. U-APSD introduces a higher signaling load when the network is not congested in this scenario mainly because we use for the AC_VO stations a voice codec with silence suppression. Therefore, when the voice traffic or QoS Nulls generated in the uplink by a station coincide with a silence interval in the downlink, there are no frames buffered in the AP for the station and the AP has to generate a downlink QoS Null in order to end the service period.

The same experiment corresponding to the results shown in Figure 9 for the *All UAPSD* case but without re-scheduling has been repeated in order to evaluate the benefit of the re-scheduling mechanism. We have compared the signaling load introduced by all ACs in both cases and computed the percentage of signaling saved with the re-scheduling mechanism until the channel gets congested, i.e., up to 8 stations per AC. The results summarized in Table 3 demonstrate that the signaling saving obtained with the mechanism depends on the congestion of the channel but still provides a significant improvement even in the case of the maximum congestion before the channel gets saturated (7 stations per AC in this experiment).

Number of STAs per AC	**1**	**3**	**5**	**7**	**8**
Signaling Load Saved (%)	25.2	23.7	21.6	14.4	9.0

Table 3
Percentage of signaling load saved with the re-scheduling mechanism in the *All UAPSD* case

In Figure 10 we show the results for the total data throughput succesfully sent over the Wireless LAN for the case of the different power saving configurations. Additionally, as a reference, we have included the same results for the case when the stations are in active mode to evaluate the channel resources that are consumed by each specific power saving method. As in the previous cases, the *All UAPSD* power saving configuration is the most efficient one followed by the *UAPSD+StdPSM* one and finally the *StdPSM* configuration presents the worst performance. The reasons for the U-APSD configurations to perform better than the *StdPSM* one is twofold. On the one hand, UAPSD reduces the collision probability as we have seen in section 4.1. On the other hand, the amount of signaling introduced under congestion conditions is smaller as previously seen in this section. With respect to the differences between the *All UAPSD* configuration and the *UAPSD+StdPSM* one, the *All UAPSD* configuration presents a better performance because the AC_BE and AC_BK stations need to introduce significantly less signaling when they use U-APSD mode Default.

As a summary, with the results analyzed in this section, we can conclude that the QoS and power saving enhancements observed in the previous sections obtained with the U-APSD configurations come at the cost of increased signaling load in non-congested situations as compared to 802.11 power save mode. However, this additional signaling load does not result in a worse channel usage efficiency for data since, when congestions appears, the UAPSD mechanism reduces the amount of signaling introduced in the channel resulting in an equal or better channel usage efficiency as compared to the 802.11 power save mode.

4.4. *UAPSD dependency on the PHY rate*

From the performance evaluation done in the previous sections we can conclude that the *All UAPSD* configuration is the most appropriate one for the upcoming mobile devices incorporating Wireless LAN functionality and requiring QoS and power saving. This is because it is the configuration that obtains the best QoS results while at the same time provides the largest power saving and channel usage efficiency. Therefore, in this section we focus on the *All UAPSD* configuration and study the dependency of the UAPSD performance improvements observed with the capacity available in the Wireless LAN channel. Three different physical layer operations are considered: 802.11b (11Mbps), 802.11g (54 Mbps) and the mixed operation of 802.11b and 802.11g (802.11b+g). The scenario used for this evaluation is the same as the one used in the previous sections but adapting the physical layer operation for all stations according to the 802.11b or 802.11g specification and by configuring in the 802.11b+g case half of the stations of each AC to use 802.11b and the other half to use 802.11g [3] .

The downlink data throughput obtained for the three different scenarios considered is shown in Figure 11. The results for the 802.11b and 802.11g stations in the 802.11b+g case are not shown separately since they are almost identical because they observe the same channel conditions. The main conclusion that can be drawn based on these results is that the QoS differentiation achieved with the U-APSD functionality in combination with EDCA is independent of the capacity in the channel. Regarding the increase in the number of stations to reach the congestion point per AC, as expected, an increase in the channel capacity available results in a larger number of stations required to suffer congestion. However, while the increase in the number of stations in the 802.11g case (dot dashed line) as compared to the 802.11b one (dotted line) is similar to the increase in the physical rate (11 Mbps $\rightarrow$ 54 Mbps $\approx$ 5 times) this is not the case for the 802.11b+g case (solid line) where the improvement is much smaller. The difference in the increase in number of stations achieved in the 802.11b+g case is due to the "overhead" required by the 802.11g stations to send a data frame to make sure that the 802.11b stations can update their NAV properly (CTS-to-self) and the requirement of using the 802.11b slot time (20μs) instead of the 802.11g one which is significantly shorter (9μs). This can be clearly seen in Table 4 where we provide for each of the three different scenarios the duration of the transmission in the physical layer of a typical data frame generated by the application using a specific AC and of a QoS Null. In the 802.11b case we consider the duration of the transmission of a frame plus the ACK, in the 802.11b+g case we consider CTS-to-self plus frame plus ACK and finally, in the 802.11g case, we consider frame plus ACK. The frames sizes considered for AC_VO, AC_VI, AC_BE and AC_BK and QoS Null are 1568 bits, 5048 bits, 12000 bits, 12000 bits and 240 bits, respectively. From the table, we can compute the reduction in the transmission duration in the 802.11b+g and 802.11g cases with respect to the 802.11b one which is below 50% in the 802.11b+g case and around 500%in the 802.11g one.

	PHY 802.11g	**PHY 802.11b+g**	**PHY 802.11g**
AC_VO	670μs	429μs	115μs
AC_VI	986μs	494μs	180μs
AC_BE	1618μs	622μs	308μs
AC_BK	1618μs	622μs	308μs
QoS Null	528μs	402μs	88μs

Table 4
Transmission duration of a data frame or a QoS Null for 802.11b, 802.11b+g and 802.11g.

Figure 12 shows the downlink delay results corresponding to the previously commented throughput ones. The same conclusions derived based on the throughput results hold in this case also. The QoS differentiation obtained when using the U-APSD and EDCA functionality together are independent of the channel capacity

[3] In the 802.11b+g scenarios the required CTS-to-self functionality has been included in the model

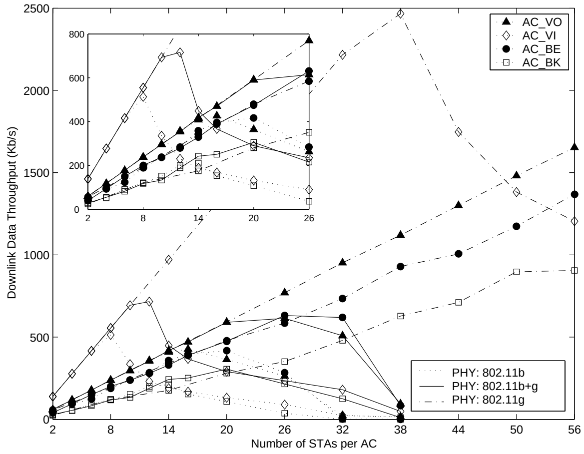

Fig. 11. EDCA+All UAPSD:: Impact of the number of stations on the MAC downlink throughput depending on the available channel capacity

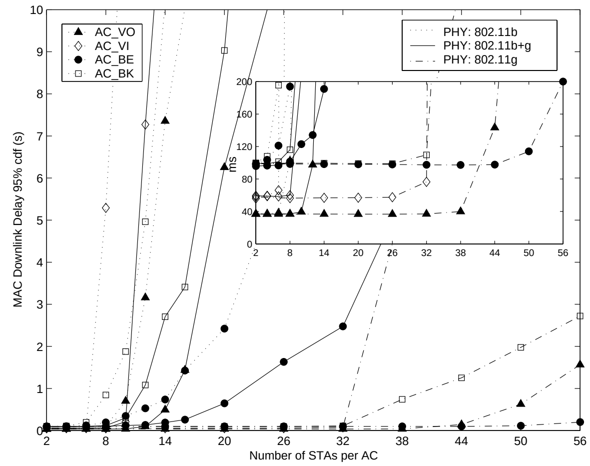

Fig. 12. EDCA+All UAPSD: Impact of the number of stations on the MAC downlink delay depending on the available channel capacity

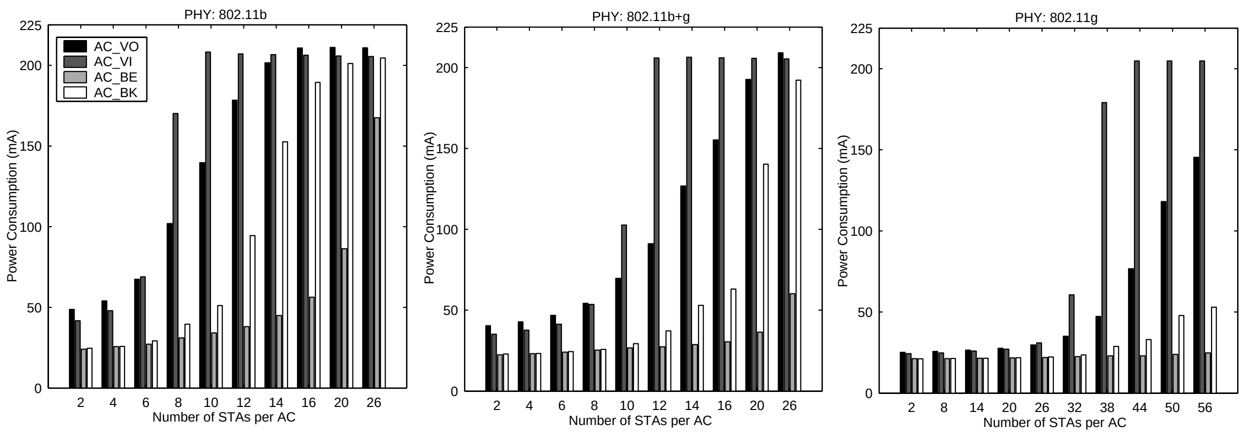

Fig. 13. EDCA+All UAPSD: Impact of the number of stations on the power consumption per AC for the *All UAPSD* depending on the available channel capacity

available and the increase in the number of stations that can be accepted in the network before the QoS suffers a degradation is proportional to the increase in channel capacity in the 802.11g case but much lower in the 802.11b+g one. For instance, if we focus on the AC_VO case and we assume for example a QoS requirement of a maximum downlink delay for the VoIP application of 100ms, the maximum number of AC_VO stations that could be accepted in the system in the 802.11b case would be 8 as compared to 10 in the 802.11b+g case and 38 in the 802.11g one.

Finally, we analyze in Figure 13 the dependency of the power consumption in the *All UAPSD* case based on the physical layer operation mode. We have opted for the actual power consumption metric in this case since it summarizes the information based on the power states in a single metric and thus, facilitates the comparison of results. As in the previous cases, the power consumption behavior is similar in all 3 cases with a different scaling based on the congestion point per AC which means that the power saving achieved with the UAPSD mechanism in combination with EDCA is independent of the channel capacity available. As expected, the larger the capacity of the channel the lower the power consumption for the same number of stations since the time to receive and transmit data frames and triggers is shorter. The relationship between the point where the power consumption increases significantly depending on the scenario is similar to the one observed in the throughput and delay results.

5. Summary & Conclusions

The forthcoming mobile devices including wireless LAN access capabilities introduce new technological challenges with respect to their QoS and power saving requirements that need to be addressed. We identified the combination of the 802.11e mechanisms EDCA and U-APSD as a promising candidate to fulfill the requirements of such specific mobile devices and evaluated the performance enhancements obtained with two different U-APSD configurations, *All UAPSD* and *UAPSD+StdPSM* as compared to simply using 802.11 power save mode.

In this paper, we provided first an overview of the 802.11 power save mode and 802.11e U-APSD mechanisms and their operation. Second, we described in detail our proposed implementation of a static U-APSD algorithm, SU-APSD, in order to meet the QoS and power saving needs of the mobile devices subject of interest. Finally, we studied via OPNET simulations the performance of the proposed U-APSD solution considering two different possible configurations as compared to the one obtained with 802.11 power save mode. Through this analysis, a deep insight on the performance differences of the U-APSD or 802.11 power save mode when combined with the EDCA QoS functionality and its causes was acquired. Therefore, the results of this study are twofold. First, we provided quantitative results of the performance enhancements to be expected when U-APSD is used as compared to 802.11 power save mode with respect to QoS perceived by the users, power saving efficiency, signaling load and effective channel usage. Second, we provided the reasoning behind the performance differences pointing out the elements to be taken into account when deciding which power saving mechanism is more appropriate for a certain device.

The main conclusions that can be drawn from our results are i) the two U-APSD configurations considered significantly outperform the 802.11 power save mode in all considered performance metrics ii) the performance enhancements obtained with U-APSD are independent of the available channel capacity and iii) the U-APSD configuration *All UAPSD* presents achieves the best results in all cases.

The study presented in this article has been performed within the framework of the product development of the 3G/WLAN mobile terminal N902iL.

6. Acknowledgments

The authors would like to thank their colleagues from the NEC 3G/WLAN dual-mode terminals team in Japan for fruitful discussions on this topic: Shinichi Morimoto, Youko Omori, Toshiyuki Sashihara and Yumi Hirano.

References

[1] IEEE 802.11 WG,*Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements*, IEEE Standard 802.11e, November 2005.

[2] S.Mangold and S.Choi and G.R.Hiertz and O.Klein and B.Walke, *Analysis of IEEE 802.11e for QoS Support in Wireless LANs*, IEEE Wireless Communications Magazine, December 2003.

[3] Wi-Fi Alliance, *Http://www.wi-fi.com*

[4] A. Banchs and X. Pérez-Costa, *An Assured Rate Service Extension for 802.11 Wireless LAN*, Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), Orlando FL, March 2002.

[5] R.Krashinsky and H.Balakrishnan, *Minimizing energy for wireless web access with bounded slowdown*, Proceedings of the eighth Annual International Conference on Mobile Computing and Networking (MOBICOM), September 2002.

[6] D.Qiao and K.G.Shin, *Smart Power Saving Mode for IEEE 802.11 Wireless LANs*, Proceedings of IEEE INFOCOM, March 2005.

[7] Y.Chen and S.Emeott, *Investigation into Packet Delivery Options for WLAN Acess Points Implementing Unscheduled Power Save Delivery*, Proceedings of IEEE Globecom, November 2005.

[8] X.Pérez-Costa, D.Camps-Mur and T.Sashihara *Analysis of the Integration of IEEE 802.11e Capabilities in Battery Limited Mobile Devices*, Wireless Communications Magazine, December 2005.

[9] X.Pérez-Costa and D.Camps-Mur, *APSM: Bounding the Downlink Delay for 802.11 Power Save Mode*, In Proceedings of IEEE International Conference on Communications (ICC), May 2005.

[10] X.Pérez-Costa and D.Camps-Mur, *AU-APSD: Adaptive IEEE 802.11e Unscheduled Automatic Power Save Delivery*, In Proceedings of IEEE International Conference on Communications (ICC), June 2006.

[11] IEEE 802.11 WG, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*,IEEE Standard 802.11, June 1999.

[12] OPNET Technologies, *Http://www.opnet.com*.

[13] F.H.P. Fitzek and M.Reisslein, *MPEG-4 and H.263 Video Traces for Network Performance Evaluation*, IEEE Network,Vol. 15,No. 6,pages 40-54, November/December 2001.

[14] Cisco Aironet 802.11a/b/g Wireless CardBus Adapter, *Http://www.cisco.com/*.