An IEEE 802.11 Energy Efficient Mechanism for Continuous Media Applications

Vitor Bernardo[a,*], Marilia Curado[a], Torsten Braun[b]

[a] *Center for Informatics and Systems, University of Coimbra, Coimbra, Portugal*
[b] *Institute for Computer Science and Applied Mathematics, University of Bern, Bern, Switzerland*

Abstract

The widespread deployment of wireless mobile communications enables an almost permanent usage of portable devices, which imposes high demands on the battery of these devices. Indeed, battery lifetime is becoming one the most critical factors on the end-users satisfaction when using wireless communications. In this work, the Optimized Power save Algorithm for continuous Media Applications (OPAMA) is proposed, aiming at enhancing the energy efficiency on end-users devices. By combining the application specific requirements with data aggregation techniques, OPAMA improves the standard IEEE 802.11 legacy Power Save Mode (PSM) performance. The algorithm uses the feedback on the end-user expected quality to establish a proper tradeoff between energy consumption and application performance. OPAMA was assessed in the OMNeT++ simulator, using real traces of variable bitrate video streaming applications, and in a real testbed employing a novel methodology intended to perform an accurate evaluation concerning video Quality of Experience (QoE) perceived by the end-users. The results revealed the OPAMA capability to enhance energy efficiency without degrading the end-user observed QoE, achieving savings up to 44% when compared with the IEEE 802.11 legacy PSM.

Keywords: IEEE 802.11, Energy Efficiency, Power Save Mode, Quality of Experience

1. Introduction

The opportunity to connect mobile equipment, sensors, actuators and other devices to the Internet, usually referred as Internet of Things (IoT) [1], raises new challenges in the deployment of those equipment. The battery lifetime is still one of the most relevant challenges, since it is directly affected by the device communication capabilities. Despite numerous efforts to create alternative low

*Corresponding author
Email addresses: vmbern@dei.uc.pt (Vitor Bernardo), marilia@dei.uc.pt (Marilia Curado), braun@iam.unibe.ch (Torsten Braun)

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

power radio technologies, IEEE 802.11 seems to be the *de facto* standard for wireless communications in most common scenarios. Therefore, it is crucial to investigate and propose mechanisms aimed at saving energy while providing Internet access through an IEEE 802.11 ready interface.

Furthermore, the massive deployment of high demand continuous media applications, namely Video on Demand (VoD) or Internet Protocol Television (IPTV), also enforces new requirements with respect to the equilibrium between energy efficiency and application performance. Besides specific application constraints, other aspects may be considered, such as end-user guidelines about whether or not energy saving is mandatory. For instance, the end-user configuration can be related with daily mobility or traveling patterns. As the end-user battery lifetime expectations are extremely hard to predict, the inclusion of end-user feedback in the optimization process will bring relevant benefits.

This work extends the Optimized Power save Algorithm for continuous Media Applications (OPAMA) [2]. OPAMA improves devices' energy consumption considering both end-user and application specific requirements, together with an optimized IEEE 802.11 power saving scheme and frame aggregation technique. Apart from using distinct application sources in the performance assessment, this paper also describes additional performance evaluation results concerning OPAMA algorithm parameters configuration. Additionally, a novel hybrid (simulation and testbed) Quality of Experience (QoE) measurement methodology is proposed, allowing the discussion about end-users' perceived quality along all studied scenarios.

The remaining sections of this paper are organized as follows. Section 2 discusses the related work, followed by the OPAMA proposal presentation in Section 3. The assessment of OPAMA performance, in the OMNeT++ simulator and using the developed hybrid video quality assessment methodology, is described in Section 4. Finally, Section 5 presents the conclusions.

2. Related Work

This section introduces the background of the proposed algorithm, and presents the most relevant related work concerning IEEE 802.11 energy efficiency improvements for continuous media applications employing power saving techniques.

An IEEE 802.11 station (STA) under Power Save Mode (PSM) [3] (also known as Legacy-PSM) is able to switch off the radio during a certain period, aimed at saving energy during that time. A STA must inform the Access Point (AP) about the current power management mode by defining the corresponding power management fields in the control frames. When the power saving mode is enabled for a STA, the AP buffers all the packets to that station. If the AP has packets buffered to a certain STA, it will send a notification using the Traffic Indication Map (TIM) field within the *Beacon* frames. In PSM, a STA must wake-up regularly to receive the *Beacon* frames. By performing this action, a STA that does not have any data buffered on the AP will be required to wake up recurrently, resulting in unnecessary energy consumption. To overcome this limitation, IEEE 802.11e [4] introduced the Unscheduled Automatic Power Save Delivery (U-APSD) algorithm. The main difference between the PSM and the U-APSD is related to the proactivity implemented in the U-APSD scheme. Unlike PSM, where only the Access Point (AP) is able to inform the station about pending packets, in U-APSD, the STA can itself ask the AP for new downlink messages pending in the queue. More recently, IEEE 802.11n [5] also announces two contributions to the power saving schemes, namely the Spatial Multiplexing (SM) Power Save and the Power Save Multi-Poll (PSMP) techniques.

Energy saving mechanisms for IEEE 802.11 can consider cooperation between the energy aware mechanisms at the lower (e.g. MAC layer aggregation) and upper layers. Camps-Mur et al. [6] have studied the impact of IEEE 802.11 MAC layer aggregation on both PSM and U-APSD schemes. The authors proposed a Congestion Aware-Delayed Frame Aggregation (CA-DFA) algorithm, which is divided into two logical parts: congestion estimation and dynamic aggregator. Congestion estimation is responsible for assessing the network capabilities and uses these values as near real-time input for dynamic aggregation. Being able to measure accurately network congestion, it allows the algorithm to dynamically adapt the maximum frame aggregation size when the network congestion goes below a certain limit. When compared with the IEEE 802.11 standard aggregation schemes, the CA-DFA performance is superior, particularly in terms of energy consumption. However, the CA-DFA algorithm does not support any end-user feedback.

Tan et al. [7] proposed a cross-layer mechanism based on the standard PSM, but using information provided by the upper layers. The algorithm, named PSM-throttling, aims at minimizing energy consumption for bulk data communications over IEEE 802.11. The PSM-throttling concept is based on the idea that there are already many Internet based applications performing bandwidth throttling and, as a result, there is an opportunity to improve energy efficiency at the client side. PSM-throttling uses the under-utilized bandwidth to improve the energy consumption of bandwidth throttling applications, such as video streaming. Nonetheless, it neither considers the inclusion of dynamic aggregation, nor the possibility that the end-user controls the maximum allowed delay. Ding et al. [8] also investigate the standard PSM capabilities and identify considerable differences between static and dynamic PSM approaches. By using preliminary results, the authors proposed a system named Percy, which uses the best of both static and dynamic methods. The Percy proposal is deployed as a transparent web proxy in the access point, and its main idea is to buffer the information in the local proxy, while the clients are running the PSM algorithm. The Percy solution does not consider the end-user feedback or frame aggregation, which could boost the 40% energy savings reported by the authors. Moreover, the trace-driven evaluation conducted does not assess the impact of the proposed mechanism on the end-users quality of experience while receiving the application data.

An adaptive-buffer power save mechanism (AB-PSM) for mobile multimedia streaming was proposed by Adams and Muntean [9] to maximize the STA sleep period. The proposal includes an application buffer, able to hide the frames from the Access Point and, consequently, to avoid the TIM reports with pending traffic indication. The authors argue that the amount of packets to store in that buffer could be dynamic, but they do not explain how to overcome this issue. Moreover, AB-PSM aims to be an application-based approach, but the mechanism to be used by the STA to provide feedback to the AP was not defined. Additionally, aggregation mechanisms were not employed and the testbed study is very limited, since only battery lifetime was analyzed. This is an important parameter, but it should always be correlated with the drawbacks introduced in the end-user application (e.g., extra delay or jitter). Another user-aware energy efficient streaming strategy for video application on smartphones was suggested by Shen and Qiu [10]. The system was modeled as stochastic process, and a Gaussian mixture model was built to forecast the end-user demands regarding the video playback time. The resulting predictive model enables a more efficient control of video download, allowing a superior control of the power states. The authors argue that energy savings of around 10% can be attained. Nevertheless, an important limitation regarding this stochastic approach is related to the need to have information about the actual user habits. Since the model uses end-user habits information as input, it is crucial that such historical information is always available. Additionally, the simulation study was performed using only a mathematical tool, where several network stack aspects are not modeled. Therefore, the obtained results are limited to the wasted energy, and do not include an analysis of network quality of service related parameters nor the end-users perceived quality.

According to Palit et al. [11] the feasibility of employing aggregation is strongly related with the scenario and/or application. In order to understand the typical packet distribution in a smartphone data communication, the authors have analyzed mobile device traffic. The main observations are that around 50% of the packets have a size less than 100 bytes and 40% have an inter-arrival time of 0.5ms or less. These conditions enable a good opportunity to perform aggregation. Using this motivation, the authors have studied the aggregation impact in the smartphones' energy consumption. The proposed aggregation scheme uses a buffering/queuing system in the AP together with PSM in the client side. The proposed packet aggregation mechanism, named Low Energy Data-packet Aggregation Scheme (LEDAS), receives packets from the different applications through the Logical Link Control sub-layer and performs the aggregation. This approach showed some good results, but application requirements, such as the maximum tolerable delay, were not taken into account. With the native support for frame aggregation in IEEE 802.11n [12], which includes two distinct approaches to perform MAC frame aggregation, named Aggregated MAC Service Data Unit (A-MSDU) and Aggregated Mac Protocol Data Unit (A-MPDU), various studies concerning aggregation performance have been done [13]. Kennedy et. al studied the adaptive energy optimization mechanism for multimedia centric wireless devices [14] and concluded that significant energy saving could be achieved when performing application-aware optimization. Pathak et al. [15] have proposed an application level energy consumption profil-

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

ing tool for mobile phones and reported issues concerning high energy usage in I/O operations. The software-based energy methodologies were early surveyed by Kshirasagar [16].

Although others in the literature [9][17] have also proposed energy optimization for continuous media applications, none takes advantage of all the key optimization parameters proposed in this work, as summarized in Table 1.

Table 1: Related work analysis summary

Parameter	[6]	[7]	[8]	[9]	[10]	[11]	[17]	This Work
Buffering techniques	✔	✔	✔	✔	✔	✔	✔	✔
Frame aggregation	✔	✘	✘	✘	✘	✔	✘	✔
End-user feedback	✘	✘	✘	✔	✔	✘	✘	✔
QoE assessment	✘	✘	✘	✘	✘	✘	✘	✔

Legend: ✔ = Yes ; ✘ = No

Therefore, this paper fulfills this gap by making an extensive study on power saving algorithm for continuous media applications, which combines the usage of buffering techniques and frame aggregation mechanisms, while using the end-user feedback to keep the application quality within the defined limits. The end-user perceived quality is investigated by employing a novel hybrid Quality of Experience assessment methodology. Additionally, although the novel power saving modes and aggregation schemes are available in more recent IEEE 802.11 standards, the Legacy PSM still is the *de facto* standard algorithm concerning PSM in IEEE 802.11, while the implementation of other algorithms is mainly optional. As a result, the algorithm is based on Legacy PSM and uses A-MSDU aggregation, which is already mandatory in the reception side of the IEEE 802.11n standard.

3. Optimized Power save Algorithm for continuous Media Applications (OPAMA)

This section describes Optimized Power save Algorithm for continuous Media Applications (OPAMA).

3.1. Motivation

Mobile end-user energy constraints are still one of the critical issues to be addressed in wireless communication protocols, particularly at the MAC Layer. IEEE 802.11, the most popular in real world equipment wireless technology, uses the Power Save Mode (PSM), usually referred in the literature as Legacy Power Save Mode (Legacy-PSM), to limit energy consumption. However, the Legacy-PSM does not bring considerable energy savings in the presence of continuous media applications (e.g., video or voice), due to protocol design limitations, as explained next.

Legacy-PSM buffers traffic at the Access Point (AP) to all the stations (STA) operating in PSM mode, which indicates that they are in a *doze* state. A

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

simplified Legacy-PSM operation example is depicted in Figure 1, where *STA-1* is operating in a *doze* state, while being served by the *AP-1*. *STA-1* must wake-up to receive the *Beacons* sent by the *AP-1* at the beginning of each *Beacon Interval*. When broadcasting a *Beacon*, an AP supporting PSM must look for pending packets for each STA in a *doze* state that is currently associated with the AP. If there is data pending for a certain STA, the AP reveals it through the Traffic Information Map (TIM) field present in the *Beacon* (e.g., *Beacon-2* indicates pending data for *STA-1*).

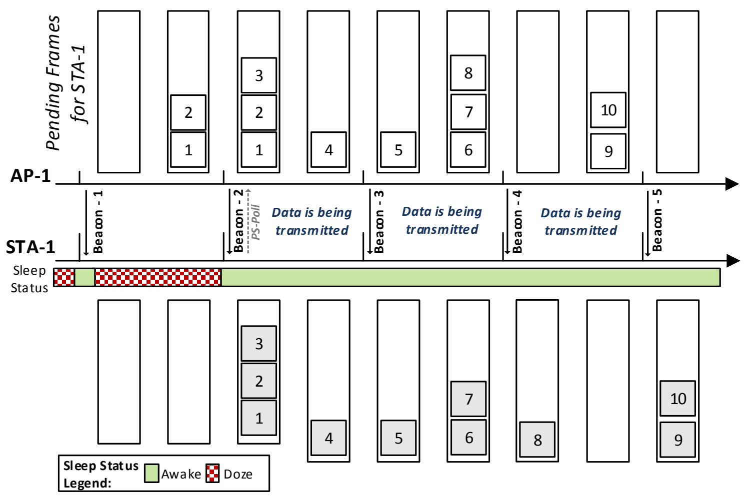

Figure 1: Legacy PSM algorithm operation example.

When receiving a *Beacon*, a STA analyzes the TIM to verify the pending information existing in the AP buffer. If there is pending data, the STA sends back a *PS-Poll* message to the AP asking for the data (e.g., *STA-1* asks for pending data upon receiving *Beacon-2*). The AP may reply with a single acknowledgement (ACK) or directly with the pending data frames. Then, the STA must stay awake while the *MoreData* flag is set. The AP will set this flag while there is data to be delivered, whereas the STA should send back a *PS-Poll* for each pending frame.

The obvious trade-off when employing Legagy-PSM is the extra delay introduced by the queuing mechanism. The delay of a certain frame, f, can be expressed as $d(f) = t_{PS-Poll}(f) - t_{Arrival}(f)$. In this equation $t_{PS-Poll}(f)$ represents the time when the *PS-Poll* to retrieve the frame f was received by the AP, and the $t_{Arrival}(f)$ indicates the arrival time of frame f to the AP queue.

If a STA is operating in sleep state, $t_{PS-Poll}(f)$ is directly related with the *Beacon Interval*, ΔBI, since, the AP announces pending data for a certain STA using the TIM field within the *Beacons*. Therefore, the minimum possible delay for the f-th frame can be expressed as $d_{min}(f) = \Delta BI - (t_{Arrival}(f) \bmod \Delta BI)$. The minimum delay represents the time between the frame arrival and the time until the next *Beacon*. This behavior is only possible assuming that the *PS-Poll* message can be sent to the AP by the STA upon receiving the *Beacon* containing pending data information.

The Legacy-PSM specification clearly indicates that a STA must stay awake when there is pending traffic to be delivered. When receiving data from a continuous media application (periodically sending data), the STA will not be able to stay in a *doze* state for long, as there will be almost always data to be received. As a result, even if the device battery is near a critical threshold, it will not be possible to save energy by employing Legacy-PSM. Such behavior is clearly shown in the Legacy-PSM operation illustration in Figure 1, since once the application starts continuously sending data, *AP-1* does not allow *STA-1* to go back into sleep, as it always has data to be received. A detailed discussion concerning PSM operation and buffer-related issues at the AP was performed by Zhu et al. [18].

OPAMA addresses these issues by introducing the expected end-user performance feedback in the process, allowing better control opportunities at the AP. The next subsection presents the OPAMA design and architecture.

3.2. Architecture

The main goal of OPAMA is to allow the end-user to save energy while keeping a desired quality at the application level. For instance, when the device battery is low, the end-user might like to have the possibility to slow down the transmission performance to a certain level in order to save energy. To accomplish this goal, the STA sleep periods must be maximized. Consequently, OPAMA will manage the AP buffer differently when compared to Legacy-PSM. While Legacy-PSM will always inform the STA about any pending data, OPAMA will employ an algorithm based on the end-user expectations for the application performance to decide when pending data information should be sent to the STA. As on Legacy-PSM, OPAMA pending packets will stay in the AP queue. As a result, this operation will not affect the Legacy-PSM standard protocol [9].

Figure 2 depicts a simplified operation scenario of OPAMA. *STA-1* is operating in a doze state, and it is being served by *AP-1*, which is then connected to the core network (not represented here).

OPAMA operates as follows: *STA-1* is left the *doze* state to receive *Beacon-1*. As there are no pending frames to be delivered, it just goes back into sleep mode. The first data for *STA-1* arrives at *AP-1* when the STA is sleeping, so is buffered. Again, *STA-1* becomes awake to receive *Beacon-2*. At this moment, there is already pending data for the STA. However, OPAMA will employ a specific algorithm (Algorithm 1) to determine whether *STA-1* should be informed about pending data. In the example of Figure 2, the algorithm returned false and the TIM of *Beacon-2* does not include information about pending data for *STA-1*. The pending data information is only sent within *Beacon-3*, followed by the data transmission start upon receiving the *PS-Poll* message. Later, in *Beacon-5* OPAMA decides again to queue the frames for a longer time, allowing *STA-1* to return into the *doze* state despite pending data being available.

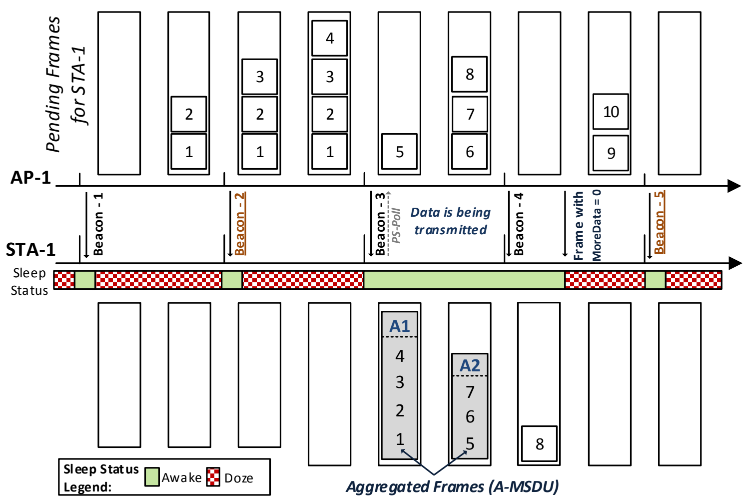

Figure 2: OPAMA algorithm simplified operation example.

When the frames stay longer in the AP queue there are more opportunities to perform aggregation, as represented in Figure 2. In this case, *Frame-1*, *Frame-2*, *Frame-3* and *Frame-4* were aggregated using the A-MSDU scheme into *Frame-A1* and *Frame-5*, *Frame-6* and *Frame-7* into *Frame-A2*. The number of frames present in each A-MSDU is dynamic and depends on the total amount of bytes to be sent. As a result, *Frame-A2* carries fewer frames than *Frame-A1*. *Frame-8* was sent without aggregation, since there is only a single frame to be sent.

Although OPAMA enables the possibility of receiving STA feedback in the AP, and uses such information to manage the pending frames accordingly, it does not specify any mechanism to control the values sent by each STA. This option will ensure a superior control at the end-user side, allowing each STA to determine what should be taken into account to define the maximum allowed delay. For instance, the maximum allowed delay can be configured by the end-user as a fixed value or dynamically calculated using application performance data.

The end-user feedback will be transmitted to the access point using two distinct messages, *PS-Poll* and *NullData*. The first message is used to request data from the AP, while the latter is an empty message used to inform the AP about shifts between two distinct power modes (e.g., going to sleep). Therefore, these message types are only transmitted from STA to AP and they do not carry payload data. OPAMA will add one extra byte field to the "Frame Body" of these messages, allowing the STAs to inform the AP about the maximum allowed delay.

The IEEE 802.11-2012 standard defines the *PS-Poll* frames as "Control" type, while the *NullData* frames belong to the "Data" type. The *subtype* value is 10 and 4, respectively, for *PS-Poll* and *NullData*. The combination of frame type and subtype allows unique identification of the frame function. Figure 3 depicts the IEEE 802.11 generic frame format and details the frame control field where frame type and subtype are presented.

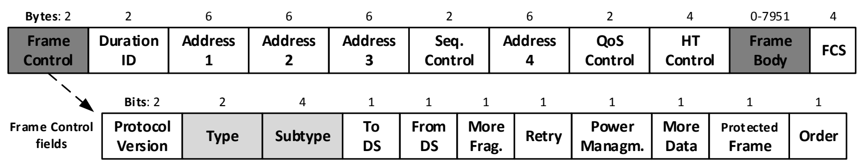

Figure 3: IEEE 802.11 MAC frame format (as in IEEE 802.11-2012 standard).

As OPAMA introduces an extra byte to transport STA Maximum Allowed Delay (STA-MAD) information, the standard frame content was changed. Therefore, to enable the proper deployment, while keeping the standard messages unaltered, there was a need to create a new unique identification for these two messages. The identification was done by selecting a frame subtype value not in use (usually named "Reserved" value). The extended *PS-Poll* and *NullData* frames were named, respectively, *OPAMA-PS-Poll* and *OPAMA-NullData*, and use the reserved frame subtypes 6 and 13.

By using the extra STA-MAD information byte, each station can send a value within the [0,255] interval. However, the AP will perform an operation to calculate the STA-MAD value to be employed. The *STA-MAD* in milliseconds for a station, s, is given by $STA\text{-}MAD(s) = CONFIGURED\text{-}STA\text{-}MAD(s) \times 10$, where $CONFIGURED\text{-}STA\text{-}MAD(s)$ is the last STA-MAD value for the station s received by the AP. This action is performed by STA-MAD variable refresh depicted in (*line 3*) of the algorithm.

The decision to determine whether pending data information should be sent is performed by the OPAMA algorithm, defined in Algorithm 1. First of all, OPAMA gets all the reference values needed to execute the algorithm, such as the maximum delay allowed by the STA or the aggregation limit support. Later, OPAMA analyzes the pending frames for the current STA, starting by verifying the delay related constraints (*lines 12 to 19*).

When analyzing each frame, OPAMA also updates the total pending bytes to be sent (*line 25*) and performs an application dependent assessment (*lines 20-24*). Actually, OPAMA provides specific mechanisms for video applications, where the main goal is to ensure that no more than a defined number of video key frames (α parameter in *line 21*) will be queued. The video key frames parameter is specific to video applications, but all the other mechanisms can be used with mixed traffic scenarios. The performance when handling combined application scenarios might depend on end-user preferences. For instance, the STA maximum allowed delay can be defined by the end-user using an algorithm designed to select the best parameter according to the end-user high level preferences for each application type.

As previously discussed, the delay for a frame, f, when employing the standard Legacy-PSM is expressed as $d(f) = t_{PS-Poll}(f) - t_{Arrival}(f)$. When using the OPAMA algorithm, the maximum delay, *OPAMA-Max-Delay*, for the f-th

Algorithm 1 Determine whether pending data information should be sent to a certain STA

```
1:  function Send_Pending_Data_To_STA_Decision(STA_MacAddress)
2:      ▷ Update the STA-MAD variable with information received from the STA in the
        OPAMA-PS-Poll or OPAMA-NullData frames.
3:      STA-MAD ← getConfiguredSTAMAD(STA_MacAddress) × 10
4:      ▷ Get the maximum aggregation defined by the IEEE 802.11 version
5:      Aggregation_Threshold ← getAggregationThreshold(STA_MacAddress)
6:      ▷ Get the time until sending next beacon
7:      TimeUntilNextBeacon ← getTimeUntilNextBeacon( )
8:      ▷ Get queued frames list for the STA
9:      STAFrameQueue ← getPendingFramesQueue(STA_MacAddress)
10:     TotalPendingBytes ← 0
11:     for each f in STAFrameQueue do
12:         ▷ Check if actual frame delay is greater or equal than the maximum delay defined by STA
13:         if getActualDelay(f) > STA-MAD then
14:             return TRUE
15:         end if
16:         ▷ Check the STA-MAD limits for the frame f
17:         if (getActualDelay(f) + TimeUntilNextBeacon) ≥ STA-MAD then
18:             return TRUE
19:         end if
20:         if f_MediaType == "video" and f_FrameType == "I" then
21:             if get_Total_Video_KeyFrames_Pending_To_STA(STA_MacAddress) > α then
22:                 return TRUE
23:             end if
24:         end if
25:         TotalPendingBytes ← TotalPendingBytes + sizeOf(f)
26:     end for
27:     if (TotalPendingBytes/Aggregation_Threshold) ≥ β then
28:         return TRUE
29:     end if
30:     return FALSE                         ▷ Pending data information will not be sent
31: end function
```

frame is given by $OPAMA\text{-}Max\text{-}Delay(f) = STA\text{-}MAD(s) - (STA\text{-}MAD(s) \bmod \Delta BI) - (t_{Arrival}(f) \bmod \Delta BI)$. $STA\text{-}MAD(s)$ represents the configured maximum allowed delay for the station s. The minimum delay for OPAMA is the same as the minimum delay for the Legacy-PSM. Concerning the algorithm complexity, OPAMA is similar to the Legacy-PSM, as the only difference of the proposed algorithm is the queuing management approach.

Additionally, the algorithm also analyzes the maximum allowed number of aggregated frames to be sent using the STA aggregation limit information ($Aggregation_{Threshold}$) and the total size of current pending data. The parameter β (*line 27*) controls the maximum number of aggregated frames, which can be queued for a certain STA. The configuration of this parameter might also be performed using dynamic approaches where, for instance, the network conditions or frames queuing time in lower layers (e.g. physical) are considered. The aggregation threshold information is associated with each STA (*line 5*), since the maximum feasible aggregation size is related to the STA Maximum

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

Transmission Unit (MTU).

The aggregation method employed in the OPAMA algorithm is the Aggregated MAC Service Data Unit (A-MSDU). This technique is defined in the IEEE 802.11 standard, and it is mandatory, at the receiver side, in all devices compliant with IEEE 802.11n. By using such approach OPAMA becomes fully compliant with the IEEE 802.11 standard aggregation techniques, which might enable a faster deployment of the proposed algorithm. The main goal of the A-MSDU technique is to allow multiple MAC Service Data Units (MSDUs) with the same source and destination to be sent in a single Mac Protocol Data Unit (MPDU). Figure 4 depicts the A-MSDU aggregation scheme.

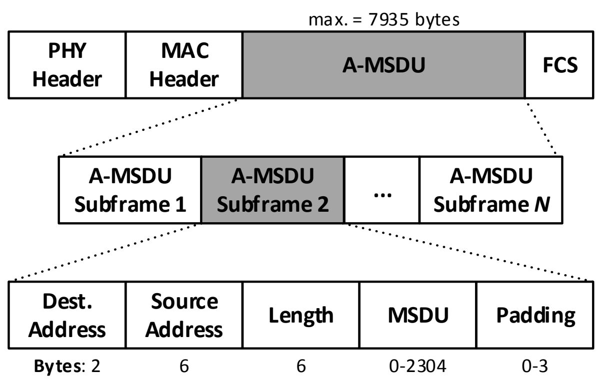

Figure 4: IEEE 802.11 A-MSDU aggregation scheme.

There is no limit of subframes to be included, but the maximum A-MSDU length defined in the standard is 7935 bytes. It should be highlighted that although the A-MSDU scheme is defined in IEEE 802.11, the standard only specifies messages format and types, to allow the standardization of encapsulation and decapsulation phases. However, the IEEE 802.11 standard does not establish an aggregation control policy. The aggregation policy is out of the scope of the standard, and must be defined afterwards as it has been done in OPAMA.

The following section presents detailed information concerning OPAMA performance when compared with Legacy-PSM and when No-PSM is used.

4. Performance Evaluation

This section shows the OPAMA evaluation performed in OMNeT++ and in a real testbed. First, a novel hybrid empirical methodology to assess video Quality of Experience within OMNeT++ simulator is presented, followed by the presentation of the simulation details and configuration parameters. Finally, a detailed OPAMA performance comparison analysis is done, including OPAMA performance against Legacy-PSM and No-PSM case, and a study concerning OPAMA key configurable parameters.

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

4.1. Video quality assessment in OMNeT++

The quality assessment is historically associated with the evaluation of performance parameters at the network layer. However, the common metrics associated with Quality of Service (QoS), namely the available bandwidth, delay or packet loss rate have as main drawback the inability to represent the real quality perceived by end-users at application level. To overcome this limitation the concept of Quality of Experience (QoE) has been developed. The QoE can also be related with the devices' energy consumption, since the end-users aim at saving energy, but keeping the quality level within acceptable bounds.

Therefore, it is important to assess the video Quality of Experience perceived by the end-users and establish a proper relationship with the energy needed to transmit such video data. As, according to the best of our knowledge, there is no application to be used within OMNeT++ able to perform video Quality of Experience assessment, the empirical video assessment methodology proposed in [19] was extended. All the specific features of video streaming, such as frame types and sending times, are collected with the aid of the Evalvid tool [20]. A similar proposal to integrate Evalvid in NS-2 simulator was performed by Ke et al. [21]. However, this proposal does not support the configuration of the number of repetitions that have to be performed, preventing a proper statistical analysis of the obtained results.

Figure 5 illustrates the proposed methodology to assess video quality within the OMNeT++ simulator. The video assessment methodology proposed in this work is a hybrid approach, meaning that it uses both OMNeT++ simulator and a real machine to assess the video quality. The simulator is used to transmit and capture both video and network information, while the real machine is employed to perform the remaining empirical evaluation procedures, such as *"Received Video File Reconstruction"* and *"Video Quality Assessment"*.

The video traffic generation is performed using a client / server logic, where there is a *"Video Server"* entity transmitting video streaming to a certain *"End-user Device"*. First, the raw video data compression must be performed, where a raw lossless YUV video is compressed to be sent to the end-user. The codec that will be used and all the compression tools should be selected in accordance with the specific requirements of the assessment goals. All the video-related procedures needed to prepare the video streaming are done in the real machine.

Apart from supplying a set of scripts to prepare the video compression, when used in a real environment, the methodology proposed in [19], also provides the tools and scripts required to start all the procedures associated with the video streaming. Here, as video streaming will be performed in OMNeT++, the methodology was extended to output a *"Video Trace File"* with all the video information, namely video frame type and size, packet fragmentation information, and start time. Additionally, a new OMNeT++ application module able to parse the *"Video Trace File"* was developed, allowing the emulated video streaming to be done.

The developed OMNeT++ video streaming application, installed in the *"Video Server"* entity, starts the transmission and simultaneously captures the

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

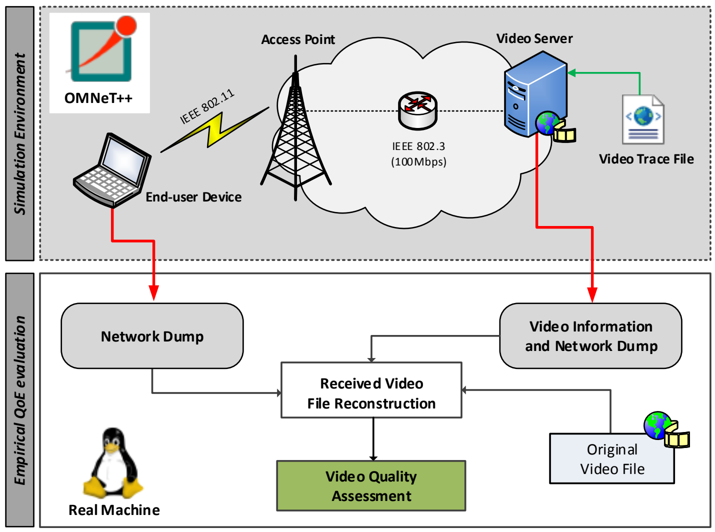

Figure 5: Video quality assessment methodology for OMNeT++ simulator.

information about the transmitted video. The same network capture is performed by the application at the receiver side until the video transmission ends. The network information about the transmitted packets is collected using the well known *libpcap* library format.

Once the OMNeT++ simulation ends, both video and network information collected from the sender and receiver are sent back to the real machine. There, the received video is reconstructed frame by frame using the captured information and the source video file. Thus, the reconstructed coded video is transformed back into raw YUV format, in order to perform the video quality comparison with the original *lossless* raw YUV video.

The basic video quality is assessed through packet or frame loss, end-to-end delay, rate information and Peak Signal Noise to Ratio (PSNR) [22]. Additionally, video quality is also assessed through QoE metrics, such as Structural Similarity Index (SSIM) [23]. SSIM is an objective and full reference image quality metric, which measures the similarity between two images. It is based on three different similarity components, namely the contrast, the luminance and the structural similarity. Unlike PSNR, which is incoherent with Human Visual System characteristics, such as human eye perception, SSIM takes into consideration human eye perception parameters, which improves the evaluation accuracy. Therefore, the resulting SSIM is a combination of the three similarity parameters into a single value between 0 and 1, where 0 means no correlation with the original image, and 1 means the exact same image.

Apart from the various assessment metrics, the employed assessment framework also allows the configuration of a playout buffer parameter. However, this playout buffer is just a parameter used to control the received video file reconstruction, as it only "converts" frame delay into loss. Additionally, this delay to frame loss "conversion" also takes into account the video group-of-picture (GOP) structure. For example, the excessive delay of a key-frame (i.e., I-frame) will make the reconstructing procedure to discard all correlated frames. This behavior represents the frame dependency of the video (de)coding process. Such behavior is very important to allow accurate testing of the OPAMA algorithm. The configured playout buffer parameter during all the following tests was 200 ms. This value was defined by the IEEE 802.11 working group usage model as the maximum delay for Internet video/audio streaming applications [24].

All the videos reconstructed using this framework will have the same total duration as the original video streamed. This is a strong limitation to study the impact of delayed video quality using the Temporal Quality Metric proposed in the ITU-T Rec. J.247 [25], which performs a superior analysis concerning the delay impact on the video quality. However, it allows a suitable quality comparison using video quality metrics such as SSIM, which can take losses into account but not delay or jitter.

The following subsection presents the OMNeT++ simulation scenario as well as the most relevant configuration parameters, including the video related information.

4.2. Simulation Scenario and Setup

The assessment of OPAMA was performed with two objectives. First, it aims to evaluate the impact of the proposed mechanism on energy consumption, delay and end-user attained Quality of Experience, when compared to Legacy-PSM and No-PSM scenarios. Second, it aims to assess the impact of OPAMA parameters configuration on the algorithm performance.

The tests were conducted in the OMNeT++ 4.2.2 [26] simulator together with the INET Framework 2.0.0. As one of the main goals of this work is to study energy consumption in the IEEE 802.11 interfaces, a multimeter like module, based on the existing INET Framework battery model, was created. This module can measure energy consumed in an IEEE 802.11 interface, by computing the time spent in each state. The simulation scenario used is illustrated in the upper half of Figure 5.

Table 2 illustrates the power values [6] used for each considered state in the IEEE 802.11 physical layer implementation and the key parameters defined for the simulation. Both Legacy-PSM and OPAMA were implemented using the OMNeT++ INET framework. The IEEE 802.11 radio Bit Error Rate (BER) used in this simulation study results from values obtained for various IEEE 802.11g physical modes, using a dedicated Orthogonal Frequency-Division Multiplexing (OFDM) physical layer simulator. The OPAMA related configuration is performed by defining both α and β parameters, used in Algorithm 1.

The assessment of OPAMA was performed using the freely and publicly available *"Elephants Dream"* raw sequence [27][28]. This sequence was encoded with H.264/MPEG-4 AVC codec using a Variable Bit Rate (VBR), and has

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

Table 2: OMNeT++ simulation parameters.

Parameter	**Value**
Total simulation time	660 seconds
Number of Runs	20
IEEE 802.11 - Operation mode	G
IEEE 802.11 - Beacon interval	100ms
IEEE 802.11 - Aggregation type	A-MSDU
Radio - Attenuation threshold	-110dBm
Radio - Maximum sending power	2.0mW
Radio - SNIR threshold	4dB
Radio - BER table	"per_table_80211g_Trivellato.dat"
Power while transmitting	2000mW
Power while receiving	1500mW
Power while idle	390mW
Power while sleeping	20mW
OPAMA α parameter	10
OPAMA β parameter	5

a resolution of 352x288, containing 14400 frames. All the movies were coded using a Group Of Pictures (GOP) of 12 frames with 24 Frames Per Second (FPS). All the video-related operations were performed using *ffmpeg* software [29] employing of the scripts and tools from [19]. The video is played for 10 minutes.

Three distinct video qualities were selected for the tests, as summarized in Table 3. The end-users' perceived Quality of Experience, given by the SSIM metric, is obtained in a real testbed by employing the hybrid video quality assessment methodology defined in Section 4.1.

Table 3: Parameters of compressed video sequences

Name	**CRF**	**Avg. Bitrate**	**Reference PSNR**	**Reference SSIM**
Video-Q1	26	340 kb/s	39.63±5.49	0.97±0.02
Video-Q2	22	539 kb/s	42.27±5.65	0.98±0.01
Video-Q3	18	845 kb/s	44.99±5.82	0.99±0.01

The video encoding was performed using the *x264* encoder and employing the Constant Rate Factor (CRF) or Quality Constant method, which is the default quality setting for this encoder. The CRF method keeps a constant quality along the video by compressing all the frames with the same quality, resulting in a variable bit rate movie. In this work three different CRFs (18, 22 and 26) were selected and mapped into three different video qualities, *Video-Q1*, *Video-Q2* and *Video-Q3*, respectively. The CRF scale ranges from 0 to 51, where 0 is lossless, 23 is the default compression and 51 represents the worst

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

possible quality.

All the results presented in the following sections include 20 runs using distinct random seed numbers with a confidence interval of 95%.

4.3. Simulation Results

This subsection presents the attained results regarding OPAMA performance assessment, compared with Legacy-PSM and No-PSM scenarios.

4.3.1. Validation of algorithm basics:

The analysis presented next has two main goals. First, it supports the validation of the OPAMA information exchange within the defined messages. Second, it aims at studying the algorithm on both network performance (i.e., delay) and energy consumption.

During these tests, to compare and validate, the information concerning STA Maximum Allowed Delay (STA-MAD) sent by the STA to the Access Point is always set equals to zero (i.e., STA-MAD = 0 ms). This configuration allows the implementation to be properly tested against Legacy-PSM and No-PSM.

Since with STA-MAD = 0 ms, OPAMA will not be able to queue frames, as it will also not be possible to perform aggregation. As it is important to validate the implemented A-MSDU aggregation mechanism, a common aggregation policy was defined for this scenario. Such policy encompasses the aggregation of all the packets arriving within a small interval ($\leq$ 5 ms). The maximum aggregation size was defined as 2272 bytes, which is the IEEE 802.11g MTU. This configuration will allow a proper validation against the Legacy-PSM.

With the stated configuration, OPAMA will perform similarly to Legacy-PSM, but using the A-MSDU aggregation scheme. Additionally, the extra byte to carry STA-MAD information is exchanged in all the messages defined to transport such data. Therefore, in order to perform a clear distinction between the usage of OPAMA with this restriction and the rest of the paper, this limited version of OPAMA with STA-MAD = 0 ms will be named as Legacy-PSM-Aggregation.

Figure 6 depicts a *boxplot* representing the end-to-end delay (in milliseconds) obtained for all the packets needed to stream each of the three distinct videos already presented (Table 3).

As expected, No-PSM shows a lower delay compared with both Legacy-PSM and Legacy-PSM-Aggregation. When assessing Legacy-PSM and Legacy-PSM-Aggregation performance it is noticeable that the delay is similar in both cases. The slightly higher maximum delay obtained with Legacy-PSM-Aggregation scenario is related with the employment of the A-MSDU aggregation technique.

The total energy consumed (in Joule) during the video transmission is illustrated in Figure 7.

The confidence interval limits are represented by the lines on the top of each bar. Although both Legacy-PSM and Legacy-PSM-Aggregation introduce extra delay, the energy savings are not significant. When employing Legacy-PSM the savings, compared with No-PSM scenario, are around 4.57%, 3.34% and 1.60%

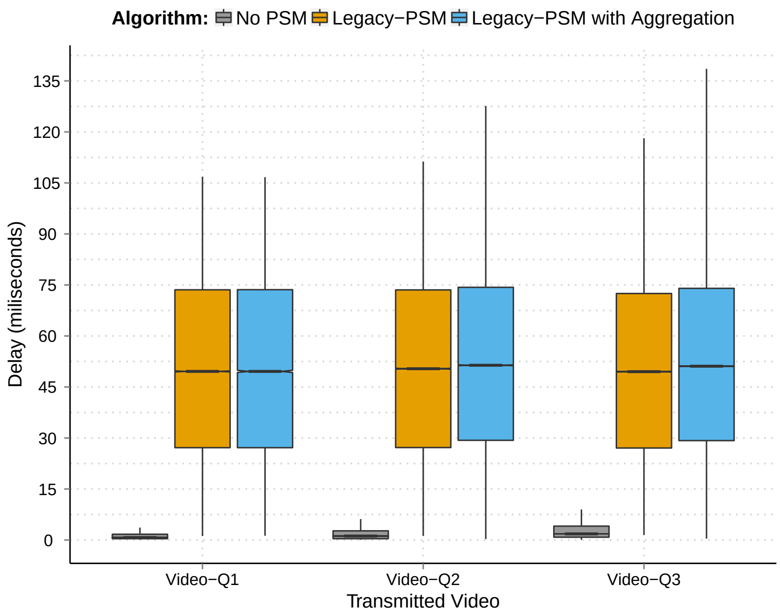

Figure 6: No-PSM, Legacy PSM and Legacy-PSM with Aggregation end-to-end delay.

for *Video-Q1*, *Video-Q2* and *Video-Q3*, respectively. The savings while using Legacy-PSM-Aggregation are 8.42%, 8.46% and 9.59%, respectively.

When analyzing the energy consumption of PSM-Legacy and PSM-Legacy-Aggregation, the latter achieves savings of 4.04%, 5.40% and 8.11%, respectively, for *Video-Q1*, *Video-Q2* and *Video-Q3*. These results clearly illustrate that solely employing aggregation is not able to solve all the energy related issues within continuous media applications. The lower energy consumption of Legacy-PSM-Aggregation depicts the benefits of aggregation in the overall energy consumption. Moreover, it is worth mentioning that the extra STA-MAD information byte does not have any impact on energy consumption.

Although these values do not constitute an optimal tradeoff between the extra delay introduced and the energy consumed, they showed the limitations of Legacy-PSM, particularly by depicting a performance degradation directly related with the video quality. As discussed previously, this behavior is mainly caused by the few sleeping opportunities of Legacy-PSM when using continuous media applications. Since those applications have almost always data pending to be transmitted, the possibilities for the STA to sleep are very limited. It must be highlighted that unlike Legacy-PSM-Aggregation, OPAMA will be able to control whether or not the pending data information should be broadcasted to the STA, allowing a better sleep period optimization.

The end-users' perceived Quality of Experience (QoE), assessed through the Structural Similarity (SSIM), shows that all the studied protocols can achieve

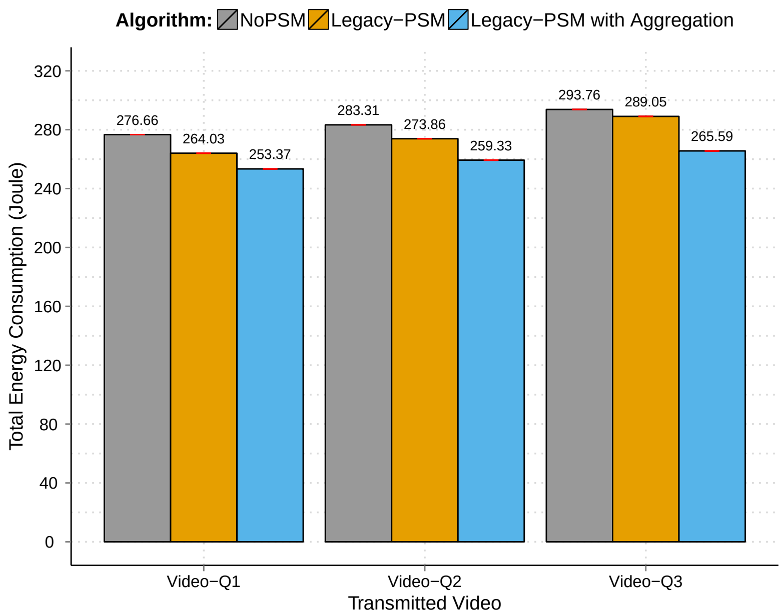

Figure 7: No-PSM, Legacy-PSM and Legacy-PSM with Aggregation energy consumption.

the maximum possible QoE, as shown in Table 3. Nonetheless, the SSIM values themselves do not reach the maximum (i.e., SSIM=1). The reason for this behavior is that the maximum possible SSIM for each sequence is directly related to the employed video data compression. The SSIM values illustrate their similarity compared with the corresponding lossless movies.

Apart from the discussed limitations and extra delay introduced both Legacy-PSM and Legacy-PSM-Aggregation are able to provide the same quality as No-PSM. Although marginal energy savings can be noticed when employing PSM-Legacy and Legacy-PSM-Aggregation, it is not possible to improve the energy/quality tradeoff or to include the end-user exceptions in the decision process. These limitations will be explored by the OPAMA algorithm.

The next subsection will study OPAMA performance when varying the Maximum Allowed Delay defined by the STA.

4.3.2. Impact of STA Maximum Allowed Delay on OPAMA performance:

This subsection studies the impact of the maximum allowed delay defined by the STA on the OPAMA performance. From now on, as the obtained results with the three distinct videos (*Video-Q1*, *Video-Q2* and *Video-Q3*) are similar, only *Video-Q2* will be used in the analysis. This sequence was selected since it represents a very good quality movie, being able to provide a very good quality perception to the end-users, achieving a SSIM=0.98 (Table 3). Figure 8 depicts a *boxplot* with the end-to-end delay (in milliseconds) in the y-axis. The x-axis represents the STA maximum allowed delay (in milliseconds). To allow a proper performance comparison, the maximum allowed delay defined by the STA was always kept constant in each test set.

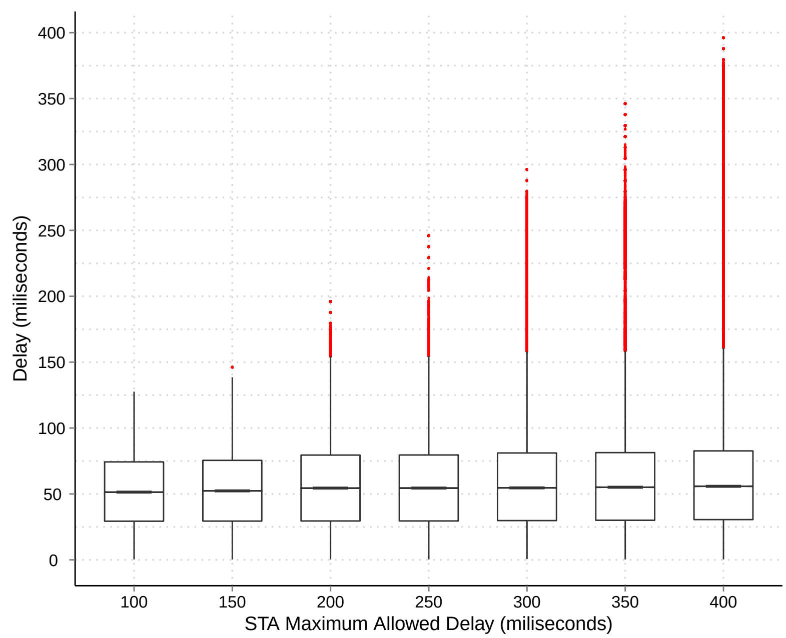

Figure 8: End-to-end delay for OPAMA with Maximum Allowed Delay defined by the STA.

The STA maximum allowed delay (STA-MAD) was never exceeded for all the test cases. By observing the *boxplots* mean values, it is possible to conclude that the end-to-end delay is around 50 ms in all the tested scenarios. The first quartile analysis shows that for 25% of the packets, the delay is about 30 ms, roughly the same as for both Legacy-PSM and OPAMA-NEF (see Figure 6). Additionally, it is also possible to observe that 75% (third quartile) of the delivered packets have only a delay of around 75 ms. The *outliers*, depicted as red points, reveal some limitations of employing OPAMA with this configuration (see Table 2). In optimal conditions the maximum delay should be near the STA-MAD, since it will allow the STA to sleep longer, while packets are queued in the AP. The study of OPAMA configuration parameters will be addressed later in this work. Nevertheless, this behavior can be explained by the strict delay control performed in conjunction with frame aggregation. OPAMA tries to maximize the number of frames sent in each A-MSDU frame, but always without exceeding the STA maximum allowed delay.

A comparison of the obtained energy savings regarding the employment of OPAMA, with both Legacy-PSM and No-PSM scenarios is shown in Figure 9. The y-axis represents the energy saved in percentage, while the maximum allowed delay (in milliseconds) defined by the STA is depicted in the x-axis.

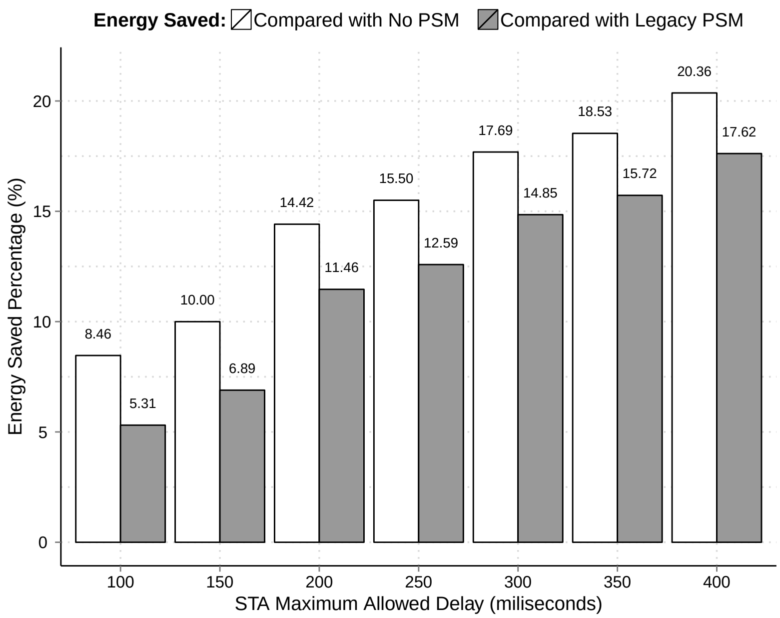

Figure 9: Energy savings with OPAMA, compared with Legacy-PSM and No-PSM scenarios.

The results show benefits of using OPAMA when the STA can accommodate some delay (e.g., by using local buffering techniques). The savings for STA-MAD = 100 ms when compared with the Legacy-PSM are around 5%, which in this particular case allows the end-user to play the video for almost 35 seconds longer using the same energy. The highest maximum allowed delays, such as 300 ms, boost the savings to around 15%. At a first glance, it might not seem interesting to employ such large delays. However, the STA can dynamically inform the OPAMA ready AP about the maximum expected delay to reflect the end-user behavior and this value might also be adjusted according to desired preferences or settings.

Figure 10 shows the SSIM (y-axis) obtained for each configured STA maximum allowed delay in milliseconds, depicted in the x-axis.

The SSIM for STA-MAD $\leq$ 200 ms is the maximum possible using the compression employed in *Video-Q2*. When compared to the scenario with STA-MAD = 250 ms, the SSIM drops from 0.9831$\pm$0.0078 to 0.9801$\pm$0.0126. In the highest maximum allowed delay scenarios there is already impact in the video quality perceived by the end-users. For instance, with a STA-MAD = 400 ms, the end-user perceived quality decreases to SSIM = 0.9273$\pm$0.0964. Notwithstanding the quality drop, the obtained SSIM values for the worst cases reveal OPAMA's capability to provide an acceptable video correlation to the end-users together with considerable energy savings.

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

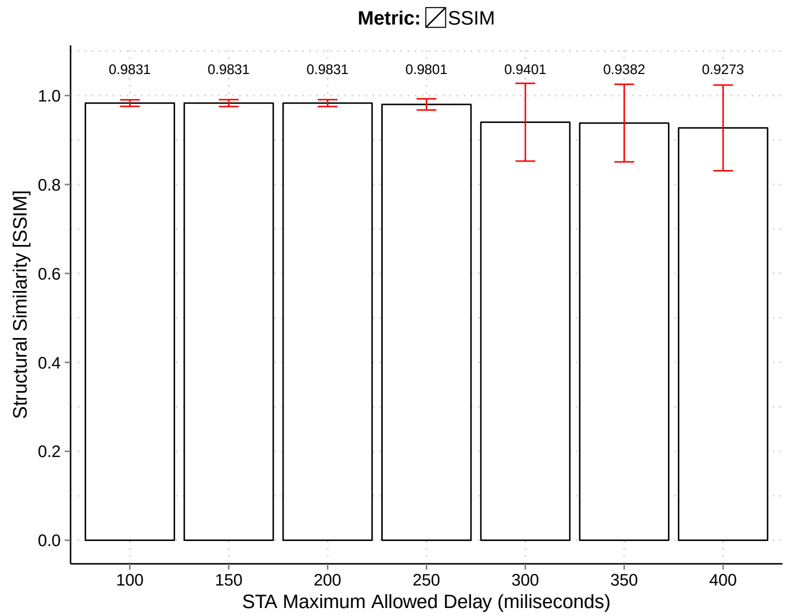

Figure 10: SSIM for OPAMA with Maximum Allowed Delay defined by the STA.

4.3.3. Impact of α and β parameters in OPAMA performance:

This subsection studies the impact of OPAMA core parameters (i.e., α and β) on the overall algorithm performance. As described during OPAMA introduction in Section 3.2, the α parameter defines the maximum number of queued data frames containing video key frames, while β controls the maximum allowed number of aggregated frames (A-MSDU) in the AP queue.

The relationship between the STA maximum allowed delay (STA-MAD) and the β value (ranging from $\beta = 1$ to $\beta = 30$) is depicted in the following figures. As presented in Table 2, the base configuration has $\beta = 5$.

Figure 11 depicts a *boxplot* for the end-to-end delay (in milliseconds) in the y-axis, while x-axis represents β configuration for each of the STA maximum allowed delay.

In a STA allowing a maximum delay of 100 ms, the β variation has no impact on the delay. Such behavior is explained by the strict control of the maximum allowed delay defined in the OPAMA algorithm. Before allowing frames to be queued, OPAMA checks whether there are (or will be) frames exceeding the defined STA-MAD. Therefore, once the delay restriction is violated, there are no benefits in allowing more frames to be queued by using a larger β. The impact of β on the achieved delay is noticeable when STA-MAD $\geq$ 150 ms, as for these scenarios the queuing opportunities, generated by the bigger allowed delays, also increase.

By observing the β behavior along the various studied STA-MAD values, it

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

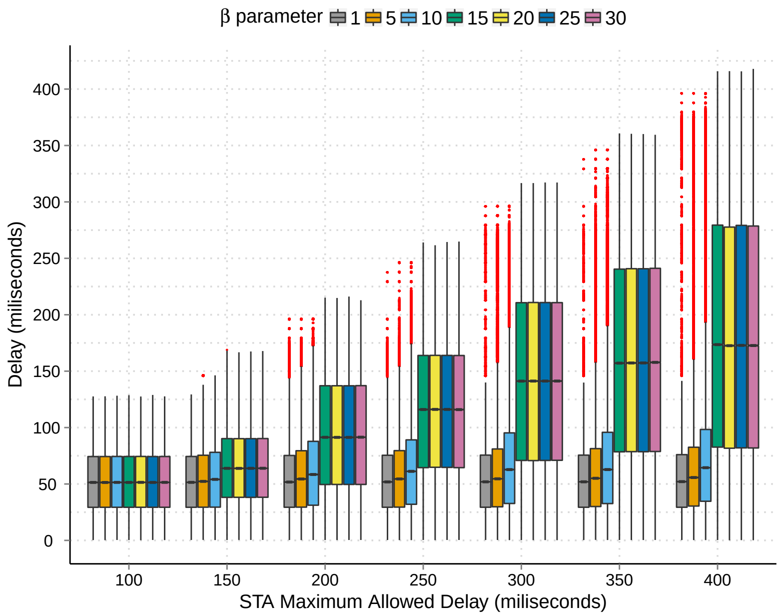

Figure 11: Delay for distinct β configurations and Maximum Allowed Delay defined by the STA.

is possible to notice that the lower β values (≤ 10) do not allow a vast percentage of the packets to be delayed by more than 100 ms. For instance, when STA-MAD = 200 ms and $\beta = 10$, there are 75% (third quartile) of the packets with a delay lower or equal than 80 ms, while the maximum delay (excluding the *outliers*) is near to 150 ms. A similar behavior can be noticed for all the cases with STA-MAD $\geq$ 200 ms. Since the obtained delay is related to the capability of a STA to sleep longer, it is crucial to maximize the delay up to the bounds defined by each STA as the maximum allowed.

When parameter $\beta \geq 15$, unless for STA-MAD = 100 ms, one can observe that the mean delay is higher, when compared to cases with $\beta < 15$, and there are also no *outliers*. Both lower and upper quartiles depict higher values, and there is a bigger interquartile range (difference between the upper and lower quartiles), meaning that 50% of packets arrived within such delay interval.

The maximum delay obtained when $\beta \geq 15$ and STA-MAD $\geq$ 150 ms exceeds marginally the maximum delay allowed by the STA. This is mainly due to the time needed for the AP to process and send all the queued frames to the network. Thus, packets inside the latest frames sent to the network might suffer some extra delay. Furthermore, there is almost no difference between the *boxplots* when $\beta \geq 15$, since the aggregation opportunities are first limited by the OPAMA delay control procedure. Consequently, as revealed by the obtained results, in this scenario it is not significant to have β values greater than 15.

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

The total energy consumption (in Joule) and the end-user perceived Quality of Experience (given by the SSIM metric), for the previously discussed scenarios, are illustrated in Figures 12 and 13, respectively.

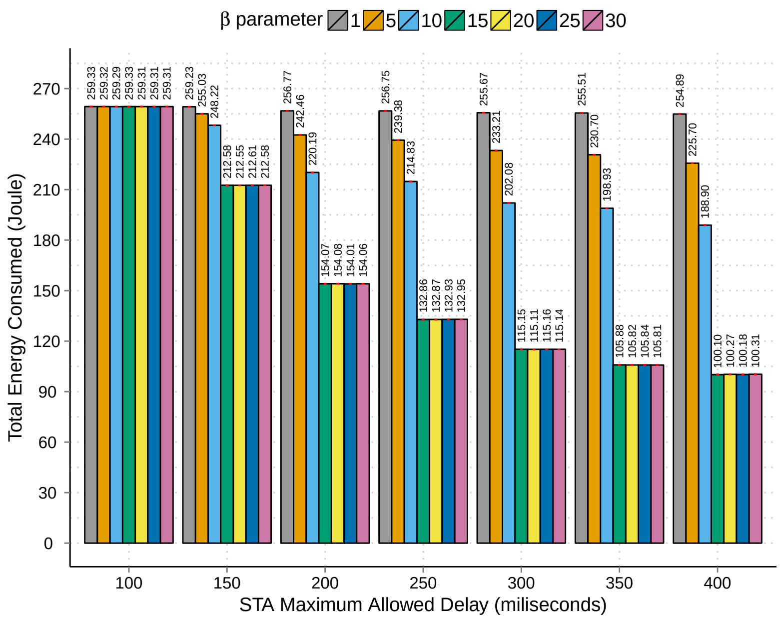

Figure 12: Energy consumption for distinct β configurations and Maximum Allowed Delay defined by the STA.

The depicted energy results show a clear relationship between the β configuration and the total energy consumption, unless when STA-MAD = 100 ms, as discussed previously. The energy consumption analysis reveals that the β configuration can considerably enhance the energy savings. Using the default setup with β=5, versus a configuration with β=15, improves energy savings from around 17% with STA-MAD = 150 ms to 36% and 56%, respectively, for STA-MAD equal to 200 ms and 400 ms. For the same scenarios, the energy savings compared with Legacy-PSM are 44% and 63%, respectively.

Nevertheless, the energy savings should be analyzed together with the obtained Quality of Experience (Figure 13), since the end-users are not only concerned about the used energy. The higher β values ($\geq$ 15) have a direct impact on the quality perceived by the end-users providing a maximum delay greater or equal to 250 ms.

Even though the perceived quality is affected by both STA-MAD and β configuration, it still achieves an acceptable quality level for the worst cases. As an example, the scenarios with β=15 and STA-MAD = 400 ms (energy saving of 56%, compared with β=5 setup) obtain a SSIM = 0.6844, which still an acceptable correlation with the original video, while in the STA-MAD = 200 ms

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

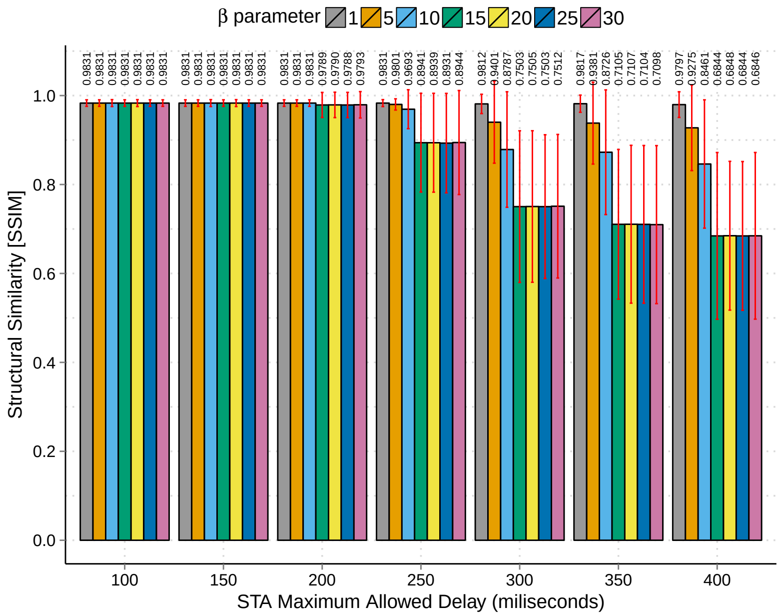

Figure 13: SSIM for distinct β configurations and Maximum Allowed Delay defined by the STA.

(energy saving of 36%) the quality is roughly the same (SSIM = 0.9789) as the reference video. By establishing a proper cost/benefit tradeoff between the energy savings and the obtained quality, OPAMA gives end-users the opportunity to select the best configuration according to their preferences or energy levels.

A similar study, using the same parameters range, was performed for the α parameter. The study includes also two distinct β configurations ($\beta = 5$ (default) and $\beta = 15$). In the first case, the α parameter shows no influence on the results, mainly because other algorithm restrictions (delay and maximum allowed number of aggregated frames) were not satisfied. With $\beta = 15$, the study reveals a minor impact with $\alpha = 25$ and 30, introducing a negligible delay and an insignificant impact on both energy consumption and perceived quality. Nevertheless, it should be highlighted that the α parameter impact is closely related with the video characteristics and it might have a different impact in other videos where, for instance, the key frames are bigger.

4.3.4. OPAMA performance with larger MTU:

OPAMA uses MAC layer aggregation (A-MSDU) as one of the algorithm components. Until now, all the tests were performed using a Maximum Transmission Unit (MTU) of 2272 bytes, which is the default for IEEE 802.11g. Nevertheless, in IEEE 802.11n, where aggregation at the receiver side is already mandatory, the MTU can be up to 7935 bytes. Therefore, this section investigates the OPAMA behavior with these two distinct MTUs. Additionally, the β parameter influence on OPAMA's performance, using those two MTUs, was also studied.

Figure 14 shows the end-to-end delay for MTU of 2272 and 7935 bytes, with two distinct values of β. The x-axis represents the maximum allowed delay by the STA (STA-MAD) in milliseconds.

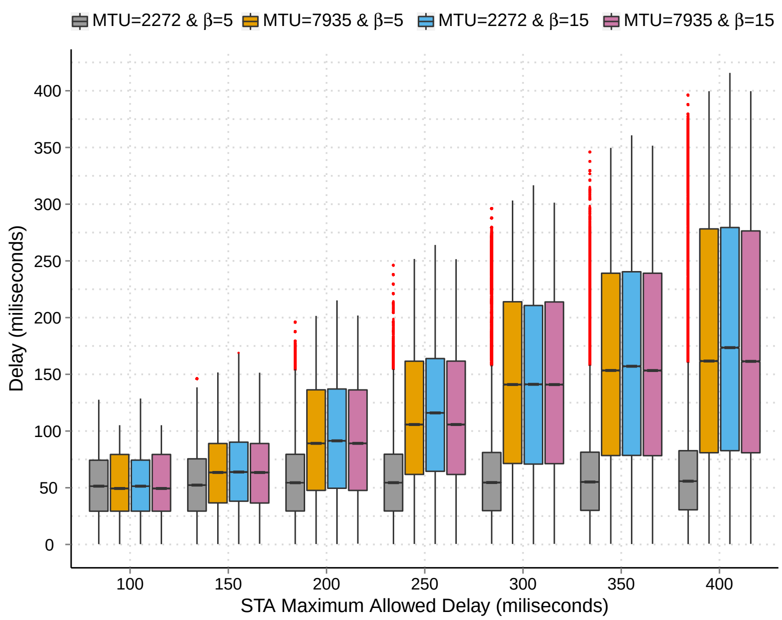

Figure 14: MTU impact on OPAMA delay.

When comparing the two MTU scenarios with $\beta = 5$ it is noticeable that the one with smallest MTU shows a lower mean delay, independent of the defined STA-MAD. This behavior is related to the OPAMA algorithm delay restrictions, as previously discussed in Subsection 4.3.2. Using the IEEE 802.11n default MTU (i.e., 7935 bytes) the mean delay starts to increase along with the STA-MAD, while the interquartile range also grows in the same proportion. Thus, in this scenario OPAMA introduces additional delay for almost all the delivered packets, but the maximum allowed delay defined by the STA is never exceeded.

For the two MTU scenarios with $\beta = 15$ both configurations present a similar delay pattern, depicting a clear increase when compared with a scenario with β=5 and MTU = 2272 bytes. Nevertheless, unlike both scenarios with MTU = 7935 bytes, the scenario with MTU = 2272 bytes and $\beta = 15$ exceed the maximum allowed delay for all the tested scenarios.

If only the end-to-end delay is analyzed, this might reveal poor OPAMA performance in scenarios with large maximum transmission units. However, the information regarding energy consumption, shown in Figure 15, highlights the opposite.

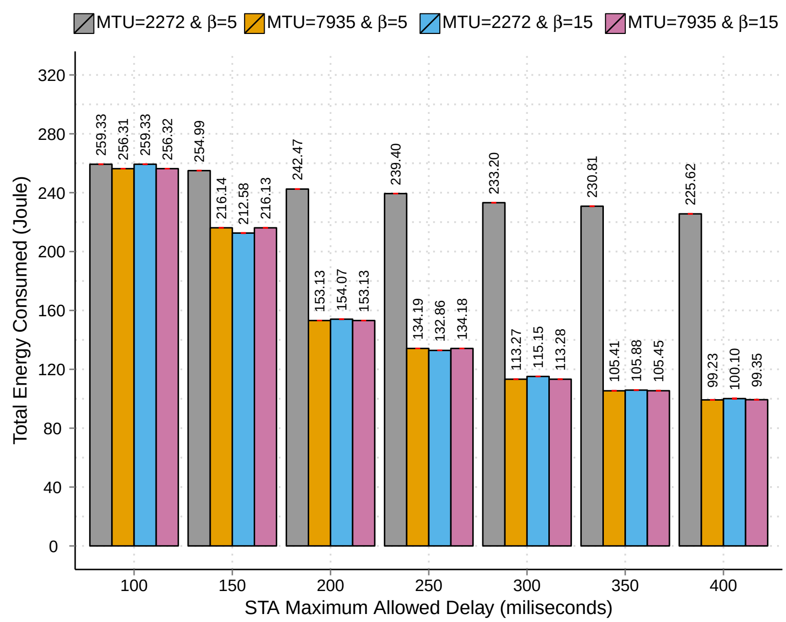

Figure 15: MTU impact on OPAMA energy consumption.

With a larger MTU (i.e., 7935 bytes), OPAMA introduces additional delay for almost all the delivered packets, but it also decreases significantly the energy consumption. A similar energy performance is achieved when using MTU = 2272 bytes and $\beta = 15$. This behavior can be explained by the higher aggregation opportunities created by the β configuration. The slightly higher energy consumption under these conditions is related with the energy costs of sending more frames to the network, when compared with the MTU = 7935 bytes case. Additionally, the usage of larger frames also reduces the number of MAC layer acknowledgments in the network, which reduces the global network contention and maximizes the STA sleep time.

In the scenarios with β=5, the usage of 7935 bytes as maximum transmission unit, when STA-MAD = 100 ms, is able to achieve savings only of around 2%. This behavior can be explained by the characteristics of the video used during these tests, which has a packet inter-departure time that does not allow OPAMA to perform better within lower STA delay restrictions. However, the savings for a STA with a maximum allowed delay of 250 ms and 400 ms are 44% and 56%, respectively. Similar savings can be obtained using MTU = 2272 bytes with $\beta = 15$. However, a different behavior can be observed in the Quality of Experience perceived by the end-users, as illustrated in Figure 16. The STA maximum allowed delay is plotted in the x-axis, while the SSIM is depicted in the y-axis.

NOTICE: this is the authors version of a work that was accepted for publication in Sustainable Computing: Informatics and Systems. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Sustainable Computing: Informatics and Systems, Volume 4, Issue 2, June 2014, Pages 106-117, ISSN 2210-5379, http://dx.doi.org/10.1016/j.suscom.2014.04.001

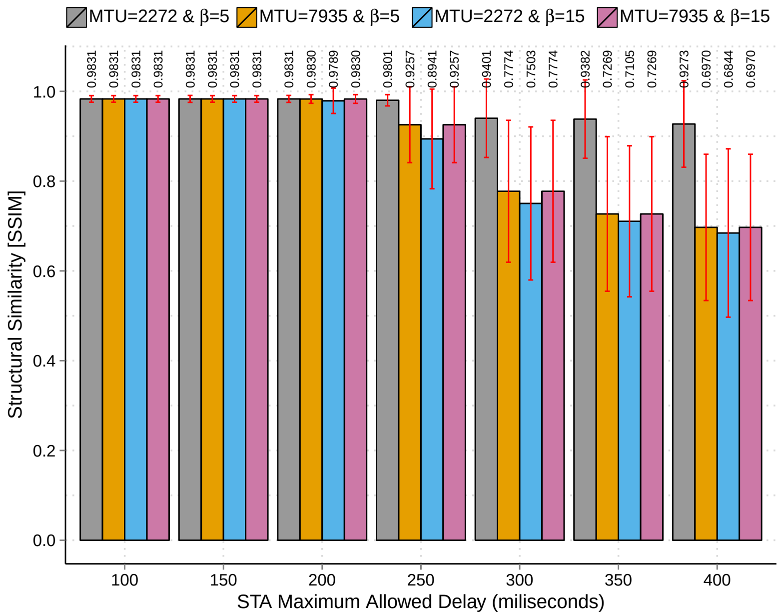

Figure 16: MTU impact on end-users' perceived quality.

When compared with Legacy-PSM case, a STA with a maximum allowed delay of 200 ms and using a MTU of 7935 bytes can achieve energy savings of around 44% without degrading the video quality perceived by the end-users. Similar energy savings can be obtained by configure OPAMA with $\beta = 15$ and using an MTU of 2272 bytes, although there is already a slight impact on the end-user quality of experience.

The results show OPAMA capabilities to benefit from larger MTUs, but also the possibility to achieve similar performance with smaller MTU, if a correct parametrization is performed. Therefore, OPAMA is able to improve the cost/benefit tradeoff between energy consumption and Quality of Experience, while keeping end-user satisfaction at a desired level.

5. Conclusions

The energy efficiency in end-user IEEE 802.11 ready devices is still an important factor towards a fast and global deployment of the future mobile communication scenarios, since battery lifetime is one of the most critical factors in a daily usage. This paper investigates and proposes a mechanism aiming at saving energy while supporting continuous media applications with a certain quality. The proposed power saving algorithm for IEEE 802.11 networks, named OPAMA, was designed to enhance the energy consumption by extending the IEEE 802.11 legacy PSM in order to accommodate the end-user feedback, and using Aggregated MAC Service Data Unit (A-MSDU) to deliver the data frames. Additionally, a novel hybrid (testbed and simulation) methodology able to evaluate the end-user perceived Quality of Experience (QoE) was also specified.

OPAMA performance assessment showed capabilities to improve energy efficiency in several scenarios, while keeping the end-user quality of experience at an acceptable level. When using IEEE 802.11g default MTU (2272 bytes) in the presence of optimal algorithm configuration, OPAMA can achieve energy savings up to 63% in a high tolerance to delay scenario and 44% for a scenario where the STA can only accommodate a maximum delay of 200 ms. The Quality of Experience assessment also highlights the relationship between the energy saved and the obtained quality, with noticeable quality drops in the highest maximum allowed delay scenarios. Nevertheless, the usage of OPAMA allows end-users to select the best tradeoff between quality and energy consumption.

The impact of MTU configuration in the proposed algorithm performance was also noticeable, showing benefits of using larger MTU whenever possible. When employing the IEEE 802.11n default MTU (7935 bytes), OPAMA performance with default configuration is similar to the optimal setup using the smaller MTU tested. Apart from the impact on the energy consumption, the usage of larger MTUs also showed benefits concerning the quality perceived by the end-users.

Acknowledgements

This work was partially supported by the COST framework, under Actions IC0804 and IC0906, as well as by the iCIS project (CENTRO-07-ST24-FEDER-002003), co-financed by QREN, in the scope of the Mais Centro Program and European Union's FEDER. The first author was also supported by the Portuguese National Foundation for Science and Technology (FCT) through a Doctoral Grant (SFRH/BD/66181/2009).

References

[1] S. Tozlu, M. Senel, W. Mao, A. Keshavarzian, Wi-fi enabled sensors for internet of things: A practical approach, Communications Magazine, IEEE 50 (6) (2012) 134 –143. doi:10.1109/MCOM.2012.6211498.

[2] V. Bernardo, M. Curado, T. Braun, Enhancing ieee 802.11 energy efficiency for continuous media applications, in: EE-LSDS 2013, Energy Efficiency in Large Scale Distributed Systems conference, Vol. 8046, Lecture Notes in Computer Science, Vol. 8046, n/a, 2013, pp. 1–15.

[3] IEEE, ANSI/IEEE Std 802.11, 1999 Edition (r2003) (2003) i –513doi:10.1109/IEEESTD.2003.95617.

[4] IEEE, IEEE std 802.11e-2005 (amendment to ieee std 802.11-1999 (2005) 0–189doi:10.1109/IEEESTD.2005.97890.

[5] IEEE, IEEE std 802.11n-2009 (amendment to ieee std 802.11-2007) (2009) 1 –565doi:10.1109/IEEESTD.2009.5307322.

[6] D. Camps-Mur, M. D. Gomony, X. PéRez-Costa, S. Sallent-Ribes, Leveraging 802.11n frame aggregation to enhance qos and power consumption in wi-fi networks, Comput. Netw. 56 (12) (2012) 2896–2911. doi:10.1016/j.comnet.2012.05.004.
URL http://dx.doi.org/10.1016/j.comnet.2012.05.004

[7] E. Tan, L. Guo, S. Chen, X. Zhang, Psm-throttling: Minimizing energy consumption for bulk data communications in wlans, in: Network Protocols, 2007. ICNP 2007. IEEE International Conference on, 2007, pp. 123 –132. doi:10.1109/ICNP.2007.4375843.

[8] N. Ding, A. Pathak, D. Koutsonikolas, C. Shepard, Y. Hu, L. Zhong, Realizing the full potential of psm using proxying, in: INFOCOM, 2012 Proceedings IEEE, 2012, pp. 2821–2825. doi:10.1109/INFCOM.2012.6195708.

[9] J. Adams, G.-M. Muntean, Adaptive-buffer power save mechanism for mobile multimedia streaming, in: IEEE International Conference on Communications, 2007. ICC '07, 2007, pp. 4548 –4553. doi:10.1109/ICC.2007.751.

[10] H. Shen, Q. Qiu, User-aware energy efficient streaming strategy for smartphone based video playback applications, in: Design, Automation Test in Europe Conference Exhibition (DATE), 2013, 2013, pp. 258–261. doi:10.7873/DATE.2013.065.

[11] R. Palit, K. Naik, A. Singh, Impact of packet aggregation on energy consumption in smartphones, in: Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International, 2011, pp. 589 –594. doi:10.1109/IWCMC.2011.5982599.

[12] D. Skordoulis, Q. Ni, H.-H. Chen, A. Stephens, C. Liu, A. Jamalipour, Ieee 802.11n mac frame aggregation mechanisms for next-generation high-throughput wlans, Wireless Communications, IEEE 15 (1) (2008) 40 –47. doi:10.1109/MWC.2008.4454703.

[13] J. Lorchat, T. Noel, Reducing power consumption in ieee 802.11 networks, in: Wireless And Mobile Computing, Networking And Communications, 2005. (WiMob'2005), IEEE International Conference on, 2005, pp. 106 – 110 Vol. 2. doi:10.1109/WIMOB.2005.1512858.

[14] M. Kennedy, A. Ksentini, Y. Hadjadj-Aoul, G.-M. Muntean, Adaptive energy optimization in multimedia-centric wireless devices: A survey, Communications Surveys Tutorials, IEEE PP (99) (2012) 1 –19. doi:10.1109/SURV.2012.072412.00115.

[15] A. Pathak, Y. C. Hu, M. Zhang, Where is the energy spent inside my app?: fine grained energy accounting on smartphones with eprof, in: Proceedings of the 7th ACM european conference on Computer Systems, EuroSys '12, ACM, New York, NY, USA, 2012, pp. 29–42. doi:10.1145/2168836.2168841.
URL http://doi.acm.org/10.1145/2168836.2168841

[16] K. Naik, A survey of software based energy saving methodologies for handheld wireless communication devices, Tech. Report No. 2010-13, Dept. of ECE, University of Waterloo.

[17] F. R. Dogar, P. Steenkiste, K. Papagiannaki, Catnap: exploiting high bandwidth wireless interfaces to save energy for mobile devices, in: Proceedings of the 8th international conference on Mobile systems, applications, and services, MobiSys '10, ACM, New York, NY, USA, 2010, pp. 107–122. doi:10.1145/1814433.1814446.
URL http://doi.acm.org/10.1145/1814433.1814446

[18] Y. Zhu, H. Lu, V. Leung, Access point buffer management for power saving in ieee 802.11 wlans, Network and Service Management, IEEE Transactions on 9 (4) (2012) 473 –486. doi:10.1109/TNSM.2012.062512.110188.

[19] V. Bernardo, M. Curado, A methodology for assessing video transmission energy consumption and quality, in: Communications (ICC), 2012 IEEE International Conference on, 2012, pp. 6308–6313. doi:10.1109/ICC.2012.6364945.

[20] J. Klaue, B. Rathke, A. Wolisz, EvalVid - A Framework for Video Transmission and Quality Evaluation, in: Proceedings of the 13th Conference on Modelling Techniques and Tools for Computer Performance Evaluation, Lecture Notes in Computer Science, Springer, 2003, pp. 255–272. doi:10.1007/b12028.
URL http://www.tkn.tu-berlin.de/publications/papers/evalvid.pdfANDhttp://citeseer.ist.psu.edu/klaue03evalvid.html

[21] hih Heng Ke, C.-K. Shieh, W.-S. Hwang, A. Ziviani, An evaluation framework for more realistic simulations of MPEG video transmission, Journal of Information Science and Engineering 24 (2) (2008) 425–440.

[22] Q. Huynh-Thu, M. Ghanbari, Scope of validity of psnr in image/video quality assessment, Electronics Letters 44 (13) (2008) 800–801. doi:10.1049/el:20080522.

[23] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: from error visibility to structural similarity, Image Processing, IEEE Transactions on 13 (4) (2004) 600 –612. doi:10.1109/TIP.2003.819861.

[24] A. P. Stephens et al., IEEE p802.11 wireless lans: Usage models. technical report, IEEE 802.11n working document 802.11-03/802r23.

[25] ITU-T, Objective perceptual multimedia video quality measurement in the presence of a full reference, Recommendation J.247, International Telecommunication Union, Geneva (August 2008).

[26] A. Varga, R. Hornig, An overview of the omnet++ simulation environment, in: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, Simutools '08, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, 2008, pp. 60:1–60:10.
URL http://dl.acm.org/citation.cfm?id=1416222.1416290

[27] Blender Foundation / Netherlands Media Art Institute, Elephants Dream movie, http://www.elephantsdream.org, Last visit: 14 August 2013 (2013).

[28] G. Van der Auwera, P. David, M. Reisslein, Traffic and quality characterization of single-layer video streams encoded with the h.264/mpeg-4 advanced video coding standard and scalable video coding extension, Broadcasting, IEEE Transactions on 54 (3) (2008) 698 –718. doi:10.1109/TBC.2008.2000422.

[29] S. Tomar, Converting video formats with ffmpeg, Linux J. 2006 (2006) 10–.
URL http://dl.acm.org/citation.cfm?id=1134782.1134792