Adaptive-Buffer Power Save Mechanism for Mobile Multimedia Streaming

Janet Adams
Performance Engineering Laboratory
School of Electronic Engineering
RINCE, Dublin City University
Ireland
Email: adamsj@eeng.dcu.ie

Gabriel-Miro Muntean
Performance Engineering Laboratory
School of Electronic Engineering
RINCE, Dublin City University
Ireland
Email: munteang@eeng.dcu.ie

***Abstract*— Wireless networks are becoming a part of everyday life for many people. When a mobile device has wireless LAN capability, multimedia content can be streamed over a wireless network to that device. However, a major disadvantage of all mobile devices is their limited battery lifetime. Multimedia streaming puts extra pressure on the battery, causing it to discharge faster. In some cases, streaming tasks cannot be completed purely because the battery of the device becomes fully discharged, which causes significant user dissatisfaction. Consequently, it is necessary to devise mechanisms to enable longer battery lifetime in order to support complex applications such as mobile multimedia streaming.**

This paper proposes an Adaptive-Buffer Power Save mechanism (AB-PSM) for increasing the battery life of mobile devices during multimedia streaming. This increase is achieved by controlling how and when data is sent over a wireless LAN. AB-PSM introduces an additional buffer which hides data from the station it is intended for, allowing it to return to sleep and consequently saving power. Data is eventually delivered in one of the station's following attempts to receive it. Tests involving AB-PSM have been performed and show good results in terms of significant increases in battery lifetime. The comparison between AB-PSM and the IEEE 802.11 legacy power save mechanism shows important increases in battery lifetime of more than 100%.

I. Introduction

Mobile devices are becoming increasingly popular for delivering multimedia content, particularly by means of streaming. The main disadvantage of these devices is their limited battery life. Unfortunately, streaming of multimedia content causes the battery of the device to discharge very fast, often causing the battery to deplete before the streaming task finishes, resulting in user dissatisfaction. It is generally not possible to charge the device while on the go as electricity socket and charger are required. Therefore, to avoid this user dissatisfaction, it is necessary to find ways to prolong the battery lifetime and to support the completion of the multimedia streaming tasks.

A typical architecture for mobile multimedia streaming is presented in Figure 1. In this architecture, a wired server streams multimedia content over a wireless IP network (e.g. IEEE 802.11b / g [1] [2] [3]) to a number of client devices. These devices could be PDAs, smartphones or any other mobile device with 802.11 connectivity.

In relation to possible power savings, the multimedia streaming process can be described as consisting of three stages: reception, decoding and playing [4]. Other researchers have shown that energy savings can be made in each stage, for example by using pre-buffering in the reception stage, feedback control during decoding and backlight adjustment for playing. However, it is not a common practice to combine energy savings in the three stages in order to achieve the best overall savings. Due to the large amount of power used by the network interface card, the reception stage is the largest consumer of the battery [5] [6].

This paper proposes the Adaptive-Buffer Power Save Mechanism (AB-PSM) that provides significant power savings in the reception stage, and hence to the overall battery life. AB-PSM introduces an additional buffer which hides data from the station it is intended for, allowing it to return to sleep and consequently save power. Data is eventually delivered in one of the station's following attempts to receive it. The proposed AB-PSM improves the existing Power Save Mechanism (PSM) without making any modifications to it.

The following section of this paper describes the related works proposed to provide power savings for each stage of the multimedia streaming process. The legacy PSM in the 802.11 standard is then described and its weaknesses identified. The proposed AB-PSM is described in detail and

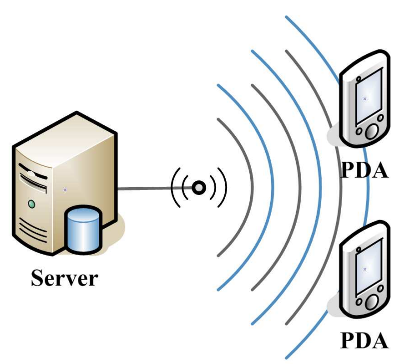

Fig. 1. Mobile Multimedia Streaming over Wireless LAN Architecture

tests and results are presented which show its benefits over the legacy PSM. Conclusion and proposed future work sections are then presented.

II. Related Works

A large amount of research has been carried out in both multimedia streaming and power saving techniques in wireless communications. However, few researchers have considered power saving in multimedia streaming and even fewer have considered the streaming process as a whole. Instead they have concentrated on only one of the three stages: reception, decoding or playing. In the following sections, some of these research findings are described.

A. Reception Stage

Chandra and Vahdat [5] propose an application-specific server side traffic shaping mechanism that can offer energy savings by allowing the client to sleep for longer periods of time. The system architecture consists of a client side proxy and a server side proxy. The server side proxy informs the client side proxy of the next data arrival. It is then the responsibility of the client side proxy to transition the client to a low power sleep state. The client can then sleep between data transfers. Although this scheme looks promising, it is not compatible with the 802.11 standard as it ignores the beacon interval which is the basis of the standard power-save mechanism.

Another scheme for power saving in the reception stage is proposed by Bae et al [7]. The authors describe a Buffer-based Energy efficient CPU scheduler for mobile devices, particularly those that run real time multimedia applications. To save power, the pre-buffering method for multimedia output is used, where output frames of real-time multimedia applications are temporarily stored in buffers. The proposed algorithm monitors the buffer occupancy and adjusts the CPU frequency accordingly. Although good results are shown to be achieved in simulations, this scheme relies on a low power hardware technique, such as Dynamic Voltage Scaling and on the pre-buffering method, which makes it difficult to be implemented on a real system or device.

B. Decoding Stage

To save power in the decoding stage of the multimedia streaming process, Pakdeepaiboonpol and Kittitornkun propose two schemes in [8]. The authors describe two high level power-saving techniques which are based on reducing the number of memory/bus accesses by high level language optimization. These solutions are aimed specifically at ARM-based devices which limits their applicability. The high level nature of this solution makes it problematic to be carried out adaptively.

Lu et al [9] propose to reduce the decoding power of multimedia by using feedback control. A controller adjusts the decoder's speed to keep constant the occupancy of the buffer between the decoder and the display, effectively matching the average decoding rate to the display rate without the need for off line profiling. The advantage of this scheme is that no pre-playback or server-side profiling is required and that slack can be reclaimed across frame boundaries. There are no real tests included in the paper and the results presented do not directly relate to the battery life of the device, making it difficult to see the actual energy savings.

C. Playing Stage

The majority of the research which has been proposed in relation to the playing stage of the multimedia streaming process is related to the display of the device, in particular the back light. Pascrich et al [10] propose an adaptive middleware-based approach to optimize back light power consumption for mobile handheld devices when streaming MPEG-1 video. Another back light power management scheme is proposed by Shim, Chang and Pedram [11]. In this case, a back light power management framework for color TFT LCD panels is proposed. The authors extend Dynamic Luminance Scaling (DLS) to cope with transflective LCD panels, which operate both with and without a back light, depending on the remaining battery energy and the ambient luminance. The scheme, known as Extended DLS or EDLS, compensates for loss of brightness when there is a rich or moderated power budget and compensates for loss of contrast when the power budget is low.

D. General Power Saving in Multimedia Streaming

Acquaviva, Benini and Ricco [12] propose a software-controlled approach for adaptively minimizing energy in embedded systems for real time multimedia processing. Energy is optimized by modifying the clock speed settings. This is a very low level solution involving hardware and it is not independent of the platform. Korhonen and Wang [6] study the impact of the burst length and peak transmission rate for observed packet loss and delay characteristics. They then implement an adaptive burst length mechanism which provides an improved trade off between power efficiency and congestion tolerance. Anastasi et al [13] address energy saving by including periodic transmission interruptions in the schedule of audio frames at the server. In this way the Network Interface Card (NIC) at the server can be set to low power state, achieving power savings. Mohapatra et al [14] propose an integrated power management approach that unifies low level architectural optimizations, OS power saving mechanisms and adaptive middleware techniques. Zhu and Cao [15] present a power conserving service model for streaming applications over wireless networks. They use a scheduling algorithm called rate-based bulk scheduling. Although both of these schemes achieve power savings, neither is similar to the power saving scheme proposed in this paper.

III. Factors that Influence Battery Power in Multimedia Streaming Process

The multimedia streaming process can be seen as comprising of three distinct stages. The reception stage refers to all of the network related tasks in the multimedia streaming process. The decoding stage involves the received media being

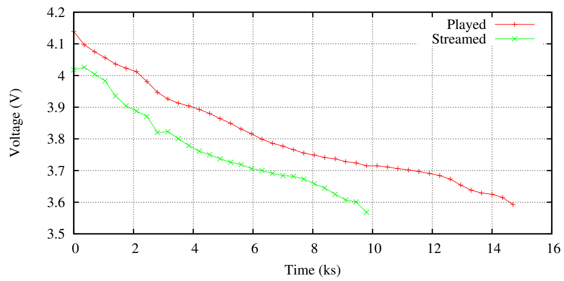

Fig. 2. MPEG 1 Layer 3 - Playing versus Streaming of Same Media Clip

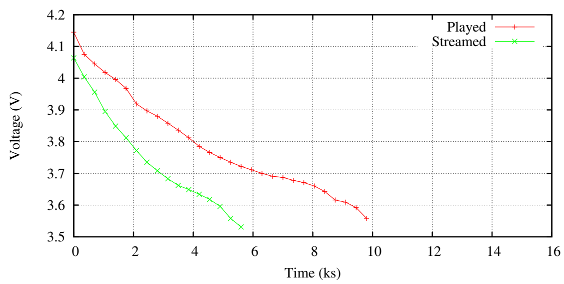

Fig. 3. MPEG 4 - Playing versus Streaming of Same Media Clip

decoded based on the encoding scheme used. The playing stage relates to the decoded media being played to the user, usually involving speakers and a screen.

Tests have been performed in order to investigate the effect of each stage on the battery power level [4]. These tests involved a media clip being played and then streamed in a continuous loop from the time the battery was full until it had depleted completely. The amount of time that the battery lasted was recorded.

Two types of media were used, an MPEG 1 Layer 3 audio clip and an MPEG 4 video clip. The MPEG 1 Layer 3 encoded audio sequence had a sampling rate of 44kHz and an average bit rate of 192kbps. The MPEG 4 encoded clip had a resolution of 352 x 288 pixels and a frame rate of 25fps. First the clips were played and then they were streamed to the same device, an Dell Axim PDA with an 802.11b network connection. Any differences in testing results could be attributed to the reception stage and a significant difference in the battery life between the played and the streamed cases was noticed. For the tests involving the MPEG 1 Layer 3 clip, results of which are shown in Figure 2, there is a difference of approximately five thousand seconds, representing a 50% increase in battery life when the media is played as opposed to streamed. The results in the MPEG 4 clip case shown in Figure 3, show that the battery lasts about four thousand seconds longer for playing than it does for streaming, accounting for a 66% increase in battery life.

Due to the extreme differences in the battery life between the playing and the streaming, the conclusion can be drawn that the reception stage plays a significant role in the problems with the battery life in the multimedia streaming process. Therefore, it is expected that most savings in terms of battery power can be made in the reception stage.

As expected, for the decoding stage, the higher the bit rate, the faster the battery is consumed. Various tests were performed in the playing stage. The test results show that both the brightness level of the screen and the volume of the speakers have a significant effect on the battery consumption rate. These tests, including details of the setup and the results, can be found in [4].

IV. Legacy Power Save Mechanism in 802.11

The reception stage is the most significant power drainer in the streaming process due to the fact that the Network Interface Card (NIC) consumes a large amount of energy in a mobile device. For this reason, methods to save battery power during this stage are being devised.

Within the 802.11 standard, there is a built in power-save mechanism (PSM). A station informs the access point whether or not it is using power management by setting the Power Management bit within the Frame Control field of the transmitted frames. This bit is set to 1 if power management is being used and to 0 if it is not. When using power management, a station can enter a low power sleep state when it is not receiving traffic. This is how the station saves power. If the access point receives packets for a station that is in sleep mode, then it will buffer these packets.

A beacon is sent to all stations at a specified interval, usually 100ms, and all stations, including those that are sleeping, wake up to receive the beacon. There is a Listen Interval which determines whether or not the station wakes for every beacon or whether it sleeps through some of them. The most significant disadvantages of using this Listen Interval are that it must be set on association with the access point and it is rarely changed from the default value, which sets the station to wake for every beacon.

Within a beacon, there is a Traffic Indication Map (TIM). This indicates whether or not there is traffic waiting for a station. If there is traffic waiting, the station will stay awake to receive it; if not it can return to the low power sleep state. Although this PSM does save power in mobile stations, the savings are minimal. This is because the Listen Interval is not used in the most power effective way.

V. Adaptive-Buffer Power Save Mechanism

This paper proposes a novel power save scheme, known as Adaptive-Buffer Power Save Mechanism (AB-PSM). AB-PSM introduces a second buffer, in addition to the data buffer that is included at the Access Point. The new buffer, called the Application Buffer, effectively hides packets from the default Access Point Buffer. When a beacon is received, the TIM only reports traffic which is waiting in the Access Point Buffer and is not influenced by the data that is in the Application Buffer.

Assuming that the Listen Interval is set to one, as is generally the case, the station will wake up every beacon interval to receive the beacon. If the TIM indicates traffic, the station will stay awake to receive it, otherwise it will return to the low power sleep mode. The Application Buffer stores

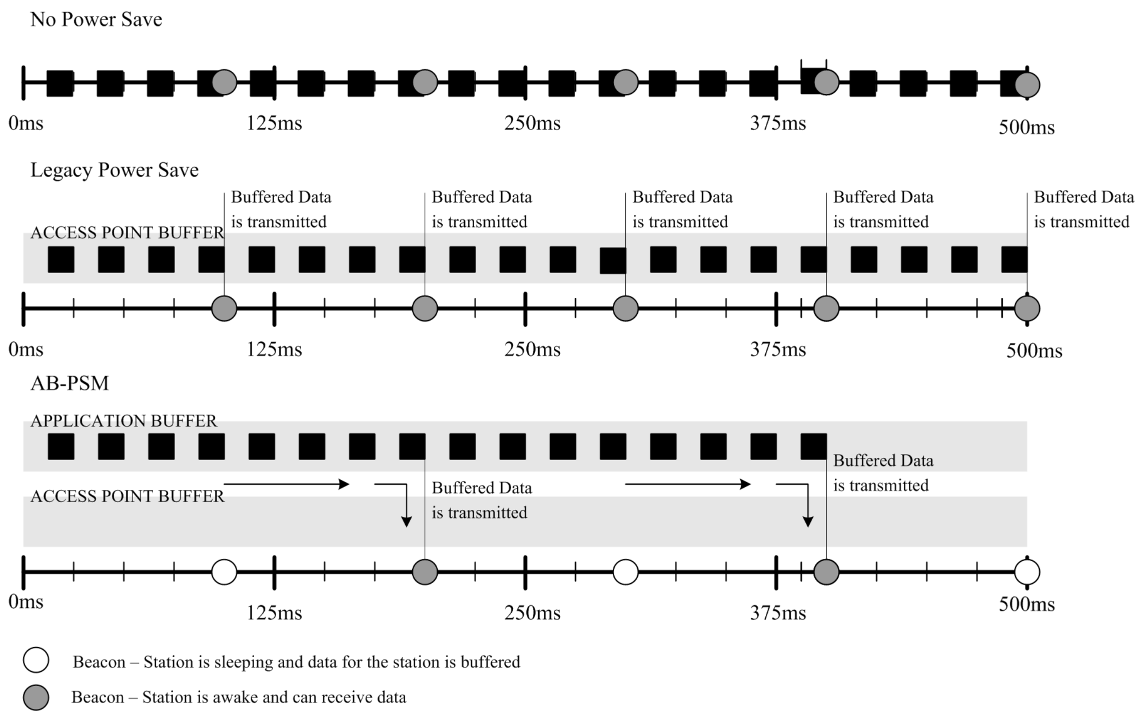

Fig. 4. Timelines for 802.11 with no power save (top), 802.11 with legacy power save (centre) and 802.11 with proposed AB-PSM (bottom).

the packets so that when the beacon is sent, the TIM reports no traffic. The station can then return to the low power sleep mode. Once the beacon interval has passed, the Application Buffer will allow the packets to move to the Access Point Buffer so that, at the next beacon interval the station will see that there is traffic waiting for it and stay awake to receive it.

The amount of time that the packets are stored in the Application Buffer can be varied. For example, the packets could be held so as to skip two beacon intervals, allowing the station to sleep for longer and the battery to last longer. However, the longer the packets are held in the Application Buffer, the higher the possibility of delay in the streaming task. For this reason, it is necessary to set a threshold time for which the packets can be hidden in the Application Buffer. This time is a multiple of the beacon interval so that 2 will refer to being allowed to skip two beacon intervals, 3 refers to being allowed to skip three beacon intervals, etc.

One of the main benefits of the AB-PSM scheme is that it requires no changes to the 802.11 standard. The station will still wake to receive beacons and in this way, will still be able to receive any broadcast or multicast packets sent on the network. AB-PSM can be controlled adaptively and the amount of time that the Application Buffer stores the packets can be adjusted based on the battery power level remaining in the device.

The scheme is described graphically in Figure 4. In this diagram, there are three timelines. The top one represents 802.11 with the power save mechanism disabled. The middle one shows 802.11 with the legacy power save mechanism enabled. The final one shows 802.11 with AB-PSM enabled. Notice the introduction of the Application Buffer in AB-PSM and the fact that the station only receives traffic every second beacon, on the others it can return to sleep.

AB-PSM saves power by allowing the station to sleep for a longer amount of time. It achieves this while still allowing the station to receive all beacons and to behave as defined in the 802.11 standard. The fact that the station can still receive all beacons means that, from the network point of view, the station is behaving identically whether using AB-PSM or not. As AB-PSM is an application-based mechanism, it will behave differently depending on the application running at that particular time, and also depending on the battery life remaining in the mobile device.

VI. Tests and Results

To examine the effectiveness of AB-PSM, tests were performed which compared it to the case when streaming is performed over 802.11 with no power save mechanism employed and with the legacy PSM, respectively.

A. Test Setup & Scenario

For the tests, a 3GHz Pentium 4 desktop computer with 1GB of RAM was used as the server. The client was a Personal Digital Assistant (PDA), with a 520MHz CPU, 64MB RAM and running Microsoft Windows Mobile 5 operating system. The multimedia content was sent from the server, to an 802.11b access point and then via the wireless network to the client. The tests involved continuously sending multimedia packets to the client. The interval and packet size were adjusted based

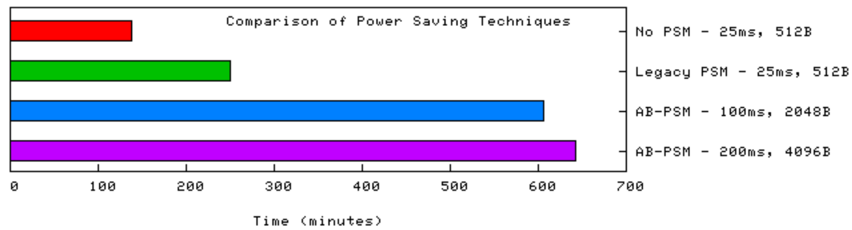

Fig. 5. Results of Tests

on the scheme being used. As the main idea of AB-PSM is to send a larger amount of data less frequently, the tests involve increasing the packet size by the same factor as the interval, hence keeping the average data bit rate constant. The beacon interval is set to the default value of 100ms in all tests.

B. Analysis of Results

The results are presented in Figure 5. The interval between sent packets is represented in terms of milliseconds and ranges from 25ms to 200ms. The packet sizes, or more specifically the size of the multimedia data chunks before lower layer fragmentation, are also shown and range from 512B(ytes) to 4096B. The graph shows four bars, each representing a different power saving scheme. The top bar refers to the 802.11 standard with the PSM switched off, with a 25ms interval and a packet size of 512B. The next bar shows the legacy 802.11 PSM with the same settings. The last two bars correspond to the proposed AB-PSM. The first has an inter-packet sending interval of 100ms and a packet size of 2048B and the second has an interval of 200ms and a packet size of 4096B. It should be noted that an identical amount of data is sent in all cases. The results show significant increases in the battery life when AB-PSM is used in comparison with both other cases: when no power saving and when the legacy PSM is employed respectively. There is an increase of 140% in the battery life when first AB-PSM scenario was considered and an increase of 160% battery life in the second AB-PSM test, both in comparison with the same legacy IEEE 802.11 PSM.

Further tests were performed to investigate the effect of the interval between sent packets and the packet size on the battery power consumption rate. In the above tests, with the results shown in Figure 5, in each of the cases average sending data rate was the same. However, in the following tests, the amount of data being sent varies and they mainly aim to investigate the effect of the packet size and inter-packet sending interval have on battery power level.

C. Investigation into the Effect of Inter Packet Sending Interval

The first example of AB-PSM from Figure 5 was used as a base for comparison. Firstly, the packet size was kept constant at 2048B, and the sending interval between two data packets varied from 50ms to 200ms. The results, which are shown in Figure 6, show that by adjusting the interval in 50ms increments, the battery lifetime increases. Between the first and second case, there is an increase in battery lifetime of over 150 minutes. Between the second and the third and the third and the fourth cases respectively, there is an increase of approximately 60 minutes per each 50ms increment. The large difference in the first case can be attributed to the fact that this is the only case where the interval is less than the beacon interval, which is set to 100ms.

D. Investigation into the Effect of Packet Size

The next set of tests studied the effect of the packet size and involved keeping the inter-packet sending interval constant at 100ms and varying the packet size. The results, presented in Figure 7, show that varying the packet size appears to have a lesser effect on the battery than varying the interval. This indicates that during AB-PSM, the large increase in battery life can be attributed to the inter packet sending interval. However, in order for AB-PSM to work it is essential to send data in large chunks so as to force the station to return to the low power sleep mode as often as possible.

Conclusion

The multimedia streaming process is divided into three stages: reception, decoding and playing. As the reception stage is the biggest consumer of power, the largest power savings can be obtained in this stage.

To obtain power saving in the reception stage, the Adaptive-Buffer Power Save Mechanism (AB-PSM) is proposed. This is a mechanism that makes use of an extra Application Buffer in order to ”hide” packets from the station they are intended for, hence allowing it to sleep for longer periods and save power. Real-world tests show significant increases in battery life when AB-PSM is used. For example, tests presented in this paper show that when AB-PSM is used, the battery life increased by 160% in comparison to the legacy power save mode described in the IEEE 802.11 standard.

Further tests were carried out to investigate the effect on the battery of packet size and interval between packets. These tests showed that the interval had a more significant effect on the battery consumption rate. However, in order for AB-PSM not to influence the data transmission rate, the packet size must be increased by the same factor as the interval. In this way, the amount of data being sent remains the same.

The results in the tests of AB-PSM, particularly when compared with the legacy PSM, are extremely good. The increase in battery life is substantial and the benefits of not

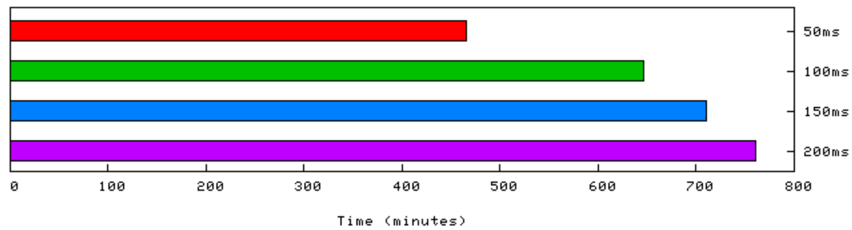

Fig. 6. Constant Packet Size, Varied Interval

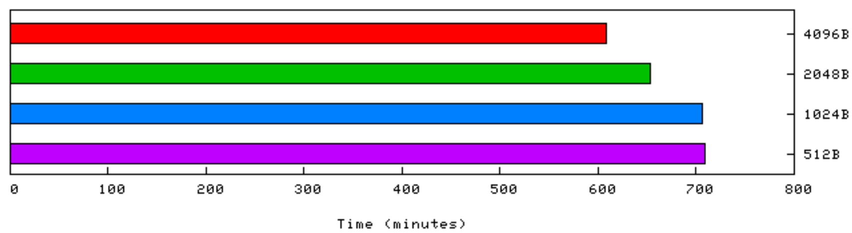

Fig. 7. Varied Packet Size, Constant Interval

having to change the standard and being able to control it adaptively make AB-PSM a very good form of power saving in mobile devices.

Future Work

Tests will be performed to investigate the effect that adapting the streamed content in order to save battery power will have on the user satisfaction. From these tests, thresholds will be defined to ensure user satisfaction. Tests will also be carried out to incorporate more than one client and also to take background traffic into account. More realistic scenarios, such as running a number of different applications throughout the lifetime of the battery, will be tested. Other types of mobile devices and battery types will be used.

The schemes for saving power in the decoding and playing stages will be combined with AB-PSM. Prioritization of the different power save mechanisms will be defined and tested.

Acknowledgment

The support of Irish Research Council for Science Engineering and Technology (IRCSET) Embark Initiative and the Enterprise Ireland Commercialisation Fund is gratefully acknowledged.

References

[1] "IEEE 802.11: Wireless lan medium access control (MAC) and physical layer (PHY) specification," 2000.

[2] "IEEE 802.11b: Higher-speed physical layer (PHY) extension in the 2.4ghz band," 2001.

[3] "IEEE 802.11g: Further higher-speed physical layer extension in the 2.4 ghz band," 2003.

[4] J. Adams and G.-M. Muntean, "Power-dependent adaptation algorithm for mobile multimedia networking," *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting*, April 2006.

[5] S. Chandra and A. Vahdat, "Application-specific network management for energy-aware streaming of popular multimedia formats," *Proceedings of the General Track. 2002 USENIX Annual Technical Conference*, pp. 329 – 42, 2002.

[6] J. Korhonen and Y. Wang, "Power-efficient streaming for mobile terminals," *Proceedings of the 15th International Workshop on Network and Operating Systems Support for Digital Audio and Video. NOSSDAV 2005*, pp. 39 – 44, 2005.

[7] G. Bae, J. Kim, D. Kim, and D. Park, "Low-power multimedia scheduling using output pre-buffering," *13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, pp. 389 – 96, 2005.

[8] P. Pakdeepaiboonpol and S. Kittitornkun, "Energy optimization for mobile MPEG-4 video decoder," pp. 1–6, November 2005.

[9] Z. Lu, J. Lach, M. Stan, and K. Skadron, "Reducing multimedia decode power using feedback control," *Proceedings 21st International Conference on Computer Design*, pp. 489 – 96, 2003.

[10] S. Pasricha, S. Mohapatra, M. Luthra, N. Dutt, and N. Venkatasubramanian, "Reducing backlight power consumption for streaming video applications on mobile handheld devices," 2003.

[11] H. Shim, N. Chang, and M. Pedram, "A backlight power management framework for battery-operated multimedia systems," *IEEE Design and Test of Computers*, vol. 21, no. 5, pp. 388 – 396, 2004.

[12] A. Acquaviva, L. Benini, and B. Ricco, "Software-controlled processor speed setting for low-power streaming multimedia," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, pp. 1283 – 92, November 2001.

[13] G. Anastasi, A. Passarella, M. Conti, E. Gregori, and L. Pelusi, "A power-aware multimedia streaming protocol for mobile users," *Proceedings. International Conference on Pervasive Services 2005*, pp. 371 – 80, 2005.

[14] S. Mohapatra, R. Cornea, N. Dutt, A. Nicolau, and N. Venkatasubramanian, "Integrated power management for video streaming to mobile handheld devices," pp. 582–591, 2003.

[15] H. Zhu and G. Cao, "On supporting power-efficient streaming applications in wireless environments," *IEEE Transactions on Mobile Computing*, vol. 4, pp. 391 – 403, July 2005.