

**Adaptive Buffer Power Save
Mechanism
for
Mobile Multimedia Streaming**

Janet Adams
B.Eng. Hon

Thesis presented for the degree of
Masters in Engineering
to the
School of Electronic Engineering
Faculty of Computing and Engineering
Dublin City University
Ireland

Research Supervisor
Dr. Gabriel-Miro Muntean

July 2007

fff

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Masters of Engineering is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed:

(Candidate) ID No.:

Date:

Acknowledgement

The support of the Irish Research Council for Science, Engineering and Technology is greatly appreciated.

I would like to thank my supervisor, Dr. Gabriel-Miro Muntean, for his support and encouragement in both my undergraduate and postgraduate studies.

I would also like to extend my thanks to the other staff members of the Performance Engineering Laboratory. Also to the other students past and present, particularly Jimmy, Hubert and Olga, who have given me invaluable advice and guidance as well as friendship and laughter.

A special thanks to my parents, Noel and Helen, who have supported me unquestionably at all times. I appreciate their patience and understanding and especially how proud they are of everything that I do.

Finally, I would like to thank Ed. His support has been unwavering and his encouragement has gotten me through some difficult times. He never stopped believing I would get to this point and for that I will be forever grateful.

Abstract

With the proliferation of wireless networks, the use of mobile devices to stream multimedia is growing in popularity. Although the devices are improving in that they are becoming smaller, more complex and capable of running more applications than ever before, there is one aspect of them that is lagging behind. Batteries have seen little development, even though they are one of the most important parts of the devices.

Multimedia streaming puts extra pressure on batteries, causing them to discharge faster. This often means that streaming tasks can not be completed, resulting in significant user dissatisfaction. Consequently, effort is required to devise mechanisms to enable and increase in battery life while streaming multimedia.

In this context, this thesis presents a novel algorithm to save power in mobile devices during the streaming of multimedia content. The proposed Adaptive-Buffer Power Save Mechanism (AB-PSM) controls how the data is sent over wireless networks, achieving significant power savings. There is little or no effect on the user and the algorithm is very simple to implement. The thesis describes tests which show the effectiveness of AB-PSM in comparison with the legacy power save mechanism present in IEEE 802.11.

The thesis also presents a detailed overview of the IEEE 802.11 protocols and an in-depth literature review in the area of power saving during multimedia streaming. A novel analysis of how the battery of a mobile device is affected by multimedia streaming in its different stages is given. A total-power-save algorithm is then described as a possible extension to the Adaptive-Buffer Power Save Mechanism.

Acronyms

AB-PSM	Adaptive-Buffer Power Save Mechanism
ACK	Acknowledgement
AM	Active Mode
AP	Access Point
APSD	Automatic Power Save Delivery
BSS	Basic Service Set
CFP	Contention Free Period
CF-Poll	Contention-Free Poll
CP	Contention Period
CPU	Central Processing Unit
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS	Clear to Send
DCF	Distributed Coordination Function
DLS	Dynamic Luminance Scaling
DS	Distribution System
DSL	Digital Subscriber Line
DTIM	Delivery TIM

DVD	Digital Versatile Disc
DVS	Dynamic Voltage Scaling
EDCA	Enhanced DCF Channel Access
EDLS	Extended Dynamic Luminance Scaling
EOSP	End of Service Period
ESS	Extended Service Set
EU	European Union
HCCA	HCF Controlled Channel Access
HCF	Hybrid Coordination Function
HDTV	High Definition Television
IBSS	Independent BSS
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ITU	International Telecommunications Union
ITU-T	ITU Telecommunication Standardization
LAN	Local Area Network
MAC	Media Access Control
MIMO	Multiple-Input Multiple-Output
MPEG	Moving Pictures Expert Group
NAV	Network Allocation Vector
NIC	Network Interface Card
OS	Operating System
OSI	Open Systems Interconnection

PC	Point Coordinator
PCF	Point Coordination Function
PDA	Personal Digital Assistant
PS	Power Save
PSM	Power Save Mechanism
QoS	Quality of Service
RTCP	The Real-time Control Protocol
RTP	Real-time Transport Protocol
RTS	Request to Send
RTSP	Real Time Streaming Protocol
S-APSD	Scheduled APSD
SIM	Subscriber Identity Module
SP	Service Period
STA	Station
TCP	Transmission Control Protocol
TIM	Traffic Indication Map
TPSA	Total Power Save Algorithm
TXOP	Transmit Opportunity
U-APSD	Unscheduled APSD
UDP	User Datagram Protocol
USB	Universal Serial Bus
UWB	Ultra Wideband
VoIP	Voice over IP

WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WNIC	Wireless Network Interface Card
WPAN	Wireless Personal Area Network
WUSB	Wireless Universal Serial Bus

Contents

1	Introduction	1
1.1	Overview	1
1.2	Motivation	1
1.3	Contributions	3
1.4	Layout of Thesis	4
2	Wireless Multimedia Streaming	6
2.1	Mobile Devices	6
2.2	Multimedia Streaming Process	7
2.2.1	System Architecture	9
2.2.2	Reception	11
2.2.3	Decoding	12
2.2.4	Playing	15
2.3	Wireless Solutions	17
2.3.1	WiFi	17
2.3.2	WiMAX	18
2.3.3	Bluetooth	19
2.3.4	Ultra Wideband	20
2.4	Summary of Chapter	21
3	IEEE 802.11 Overview	22
3.1	IEEE 802.11 Architecture	22
3.2	Point Coordination Function	24
3.3	Distributed Coordination Function	24
3.4	Amendments to the Standard	26
3.4.1	IEEE 802.11b	26
3.4.2	IEEE 802.11a	27
3.4.3	IEEE 802.11g	28

CONTENTS

3.4.4	IEEE 802.11n	28
3.4.5	IEEE 802.11e	29
3.5	Power Saving in IEEE 802.11 Networks	30
3.6	Power Saving in IEEE 802.11e	34
3.7	Summary of Chapter	35
4	Literature Review	38
4.1	Power Saving in Multimedia Streaming	38
4.2	Power Saving in the Reception Stage	42
4.3	Power Saving in the Decoding Stage	43
4.4	Power Saving in the Playing Stage	45
4.5	Summary of Chapter	46
5	Power Investigation during Multimedia Streaming	47
5.1	Preliminary Testing	47
5.1.1	Reception Stage	49
5.1.2	Decoding Stage	51
5.1.3	Playing Stage	51
5.2	Analysis of Results	54
5.3	Total Power Save Algorithm	55
5.3.1	Overview	55
5.4	Chapter Summary	61
6	Adaptive Buffer Power Save Mechanism	62
6.1	Proposed Solution	62
6.2	Adaptive-Buffer Power Save Mechanism	63
6.3	Scalability Issues and Solutions	67
6.4	Advantages of AB-PSM	68
6.5	Summary of Chapter	70
7	Experimental Tests	71
7.1	Test Setup and Scenarios	71
7.2	Preliminary Tests	72
7.2.1	Analysis of Results	73
7.2.2	Investigation into the Effect of Inter Packet Sending Interval	74
7.2.3	Investigation into the Effect of Packet Size	75

CONTENTS

7.3	Investigation into Effectiveness of AB-PSM in Multimedia Streaming	75
7.4	Summary of Results	84
7.5	Summary of Chapter	84
8	Conclusions & Future Work	85
8.1	Conclusion	85
8.2	Contributions	86
8.3	Future Work	88
	Bibliography	90

List of Figures

2.1	European Mobile Penetration - December 2006	8
2.2	Multimedia Streaming Process	9
2.3	Client Server System Architecture	10
2.4	System Architecture - Block Diagram	10
2.5	Typical UWB WPAN	21
3.1	802.11 Ad Hoc Architecture	23
3.2	802.11 Infrastructure Architecture	24
3.3	Distributed Coordination Function	25
3.4	802.11b Decreasing Signal Strength & Bandwidth	27
3.5	802.11 Frame Formats	31
3.6	802.11 TIMS and DTIMS - Example with DTIM sent at every three TIM intervals	31
3.7	IEEE 802.11 PSM	33
5.1	MP3: Comparing Playing and Streaming	50
5.2	MPEG-4: Comparing Playing and Streaming	50
5.3	MP3: Comparing Multimedia Encoded at Different Bitrates	52
5.4	MPEG-4: Comparing Multimedia Encoded at Different Bitrates	52
5.5	MP3: Comparing Sound On and Sound Off	53
5.6	MPEG 4: Comparing Different Brightness Levels	53
5.7	Total Power Save Algorithm	58
5.8	TPSA: State Diagram of Decision Algorithm	59
6.1	AB-PSM: Block Diagram	63
6.2	AB-PSM Examples with One and Two Skipped Beacon Intervals	65
6.3	Graphical Comparison IEEE 802.11 Transmission with: No Power Save, Legacy Power Save and AB-PSM	66

LIST OF FIGURES

6.4	Scalability Issues of AB-PSM	68
7.1	Test Setup	72
7.2	Results of AB-PSM Tests with Non-Video Packets	74
7.3	Constant Packet Size, Varied Interval	76
7.4	Varied Packet Size, Constant Interval	76
7.5	Analysis of Packets, No AB-PSM, Legacy 802.11 PSM disabled	78
7.6	Analysis of Packets, No AB-PSM, Legacy 802.11 PSM enabled	79
7.7	Analysis of Packets, AB-PSM with One Skipped Beacon, Legacy 802.11 PSM enabled	81
7.8	Analysis of Packets, AB-PSM Level with Two Skipped Bea- cond, Legacy 802.11 PSM enabled	82
7.9	Battery Lifetime for Different Power Save Mechanisms	83

List of Tables

3.1	802.11 Family	37
3.2	802.11 Amendments	37
5.1	Stages Affected by Each Test: Reception (R), Decoding (D) or Playing (P)	48
5.2	Scaling of Power Save Mechanisms	59

LIST OF TABLES

fff

Chapter 1

Introduction

This chapter provides an introduction to the thesis. The idea of a mobile device is discussed followed by a description of the motivation behind the work and the contributions provided by the thesis. The publications that have come out of this work are then described. Finally, an overview of the layout of the thesis is given.

1.1 Overview

This thesis offers two principal contributions. The first is the Adaptive-Buffer Power Save Mechanism. This is an algorithm that works with Institute of Electrical and Electronics Engineers (IEEE) 802.11 wireless networks to give extensive power savings for mobile devices that are performing multimedia tasks. The second contribution is an investigation into how these multimedia streaming tasks affect the battery of the device.

1.2 Motivation

The primary focus of this thesis is wireless multimedia streaming to mobile devices. This work is motivated by the fact that batteries in mobile de-

vices do not perform well when the strain of multimedia streaming is placed upon them. The combination of the multimedia tasks, which use mainly the screen and speakers as well as Central Processing Unit (CPU) and memory for decoding, and the streaming of the multimedia content, which uses the Wireless Network Interface Card (WNIC), consumes the power in the battery very quickly, causing it to deplete before the task is completed. In particular, the WNIC has been shown to be one of the most power hungry parts of a mobile device [1], [2], using as much as 50% of the total system power in some cases [3]. Mobile devices have not been designed to maintain such power intensive tasks for a long period of time [4] and if the battery dies before the task is complete, the user satisfaction is significantly decreased.

Current research has proposed many schemes for power saving in IEEE 802.11 networks. Much of the proposed research has been based on the Power Save Mechanism (PSM) that was proposed in the IEEE 802.11 standard [5]. Some have tried to improve it, others have tried to change it. There has been limited work performed in the area of power saving during multimedia streaming as the majority focus on either saving power or improving how the multimedia content is streamed and do not combine the two.

This work focuses on improving the way that multimedia content is streamed in order to make it as battery efficient as possible. The primary aim of the work is to increase the battery lifetime in a mobile device, that is to increase the amount of time that the device is active and is capable of performing tasks. This is achieved by changing how the data is sent and by focusing on the streaming of the data. The IEEE 802.11 wireless networks are the main focus of the work with the concentration being on improving the current, standards based, PSM while changing it as little as possible. By ensuring that the changes made to the standard remain minimal, the proposed scheme will be implementable on current 802.11 devices and can be easily added to current 802.11 networks.

1.3 Contributions

This thesis offers two principal contributions. The first is the **Adaptive-Buffer Power Save Mechanism (AB-PSM)**. This is an algorithm that works with IEEE 802.11 to give extensive power savings for mobile devices that are performing multimedia tasks. The algorithm improves upon the legacy PSM that is presented in the IEEE 802.11 standard. The algorithm gives significant power savings enabling the battery of the device to last longer and hence offering the possibility to complete a task that could not have been completed had the battery run out. By using AB-PSM, power savings of up to 50% can be obtained for streaming video to a mobile device. When non-video packets are used, the savings increase to over 100%, offering significant extra battery lifetime. The proposed solution works in conjunction with the standards-based PSM, making it easy to implement on current 802.11 devices.

The second contribution is an **investigation into how these multimedia streaming task affect the battery of the device**. The multimedia streaming process is divided into three stages: reception, decoding and playing. Tests are performed to examine exactly how each of these stages influence the battery of the device.

A number of publications have arisen from this research. An investigation into how the battery of a mobile device is affected by multimedia streaming tasks is described in [6]. The multimedia streaming task is divided into three stages and the effect that each stage had on the battery was shown. In [7], a total power save algorithm that could be used to save power in mobile devices while performing multimedia streaming tasks is discussed. The Adaptive-Buffer Power Save Mechanism is described in [8]. The algorithm offers significant power savings for mobile devices performing multimedia streaming tasks over an IEEE 802.11 wireless network.

1.4 Layout of Thesis

This thesis is laid out as follows. Chapter 2 presents the wireless multimedia streaming process, in particular how it is approached throughout this thesis. The system architecture used in this work is then described. Next, some of the more common protocols related to encoding and transmission are discussed followed by an overview of a number of wireless solutions.

Chapter 3 gives an overview of IEEE 802.11, with particular emphasis on the power save mechanisms within it. All tests in this thesis have been performed using an 802.11b [9] network, so the chapter focuses primarily on this. However, in order to understand 802.11b, it is first necessary to explore the legacy 802.11 standard, on which 802.11b is based. Other amendments to the standard are also described.

Chapter 4 is an overview of related work performed by other researchers. In designing the AB-PSM algorithm, it was essential to examine the current solutions that have been proposed and their effectiveness in relation to power saving. Similar algorithms must be comparatively explored and it was also necessary to ensure that a similar algorithm has not already been developed.

Chapter 5 discusses the preliminary work which was performed to investigate the areas of the multimedia streaming process that were most power intensive and the possible benefits of power save in each of these areas. In order to decide which area of the multimedia streaming process would benefit most from power saving mechanisms, a number of tests were performed. These tests are presented and their results analysed. A suggested Total Power Save Algorithm (TPSA) is also briefly described.

Chapter 6 offers a theoretical description and analysis of the newly proposed, novel power save scheme, AB-PSM. The algorithm is depicted in detail, its benefits in the field of power saving for multimedia streaming are given and some of its weaknesses are highlighted.

Chapter 7 presents the tests and results that were performed using AB-PSM. The test setup is described first, including the equipment that was

used. Then each of the tests are presented and their results analysed.

The final chapter describes conclusions and future work related to power save solutions in wireless multimedia streaming.

Chapter 2

Wireless Multimedia Streaming

This chapter describes wireless multimedia streaming with particular focus on streaming to mobile devices. This includes a description of a mobile device and a detailed discussion of the multimedia streaming process, as it is seen in this thesis. The system architecture is also discussed followed by a description of a number of protocols related to multimedia streaming in the areas of encoding and transmission. Possible wireless solutions are then presented.

2.1 Mobile Devices

Mobile devices are becoming smaller and more advanced. They are now capable of running increasingly complex applications. These devices can now handle high quality multimedia tasks due to improved screens and speakers. Additionally, many of these devices can connect to a Wireless Local Area Network (WLAN), such as those included in the IEEE 802.11 family.

In this thesis, a mobile device is classified as a device that is portable, relatively small and with a screen capable of showing video content. The device will have connectivity to a wireless network. It will be battery powered and can be recharged once the battery depletes. Examples are Personal

Digital Assistants (PDAs) and mobile phones. In this thesis, mobile device does not refer to very high-speed (for example vehicle-mounted) devices but rather to a portable device that will work both when moving at a reasonable speed and while at a fixed location. The term mobile device refers to devices which can be carried while operating and communicating.

Mobile devices are also becoming more popular than ever before. According to the Commission for Communications Regulation ¹, in their Irish Communications Market quarterly key data report [10], mobile penetration in Ireland is now at 111% based on a population of 4.235 million. These figures are based on the number of active Subscriber Identity Module (SIM) cards. The results indicate that some users have more than one active SIM card, possibly one for work and another for personal use. This has increased steadily since the last quarter of 2002, when it was at 79% penetration. It reached 100% penetration in the third quarter of 2005 and has continued to increase since. When compared with other European countries, Ireland is at the European Union (EU) average. This is shown in the graph presented in Figure 2.1.

2.2 Multimedia Streaming Process

Throughout this thesis, the multimedia streaming process is assumed to have three stages. The purpose of dividing the streaming process into three stages is so that the highest possible power savings can be achieved. By investigating the effect of each stage on the battery life of the device, it is possible to ascertain which stage is the most battery intensive and hence, to know where it will be most effective to make changes. The three stages of the multimedia streaming process are as follows:

- **Reception**

¹www.comreg.ie

- **Decoding**
- **Playing**

The reception stage involves the media being sent across the network and being received by the client. It is a combination of all the network related tasks involved in the streaming process.

The decoding stage comprises the media being decoded by the client device once it has been received. There will be two encoding schemes tested, the first is MPEG-1 Part 3 and the second is MPEG-4. These will be described in more detail later.

The playing stage occurs once the media has been received and decoded. The nature of this stage will depend on the media type which is received. For video, the screen and speakers will be involved, for music only the speakers and for images or silent video, it will just be the screen.

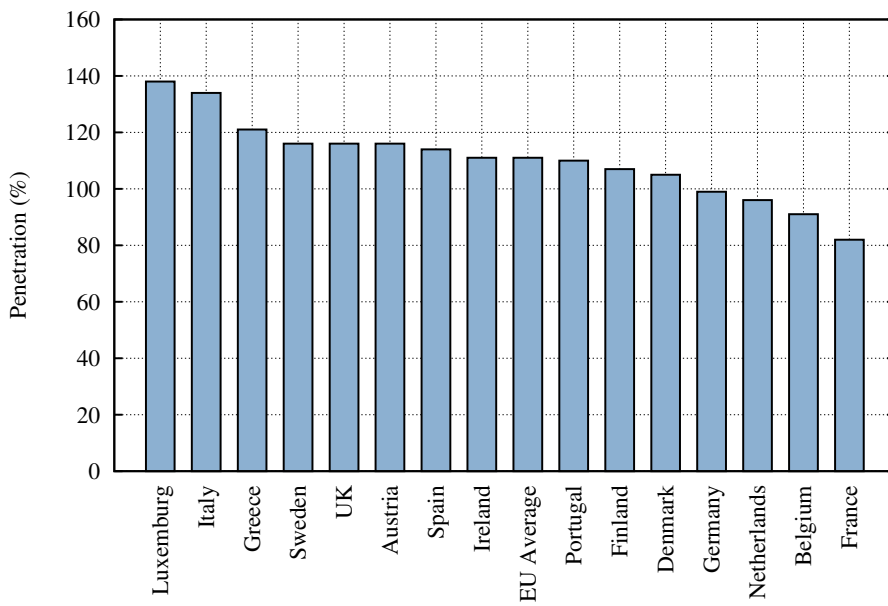


Figure 2.1: European Mobile Penetration - December 2006

The multimedia streaming process is shown graphically in Figure 2.2. The diagram shows that the reception stage is between the server and the client, via the Access Point (AP). The decoding stage is focused primarily on the client device, although the encoding scheme used at the server will have a direct impact on this stage. The playing stage is an interaction between the user and the client device.

2.2.1 System Architecture

The system architecture used throughout this thesis is a client-server based one. The client and server communicate over a wireless network, namely an IEEE 802.11b network. The client is a wireless device such as a PDA or a mobile phone. An overview of the system architecture is given in Figure 2.3, showing a server communicating with a number of wireless clients through a wireless AP. A more detailed, block diagram is illustrated in Figure 2.4,

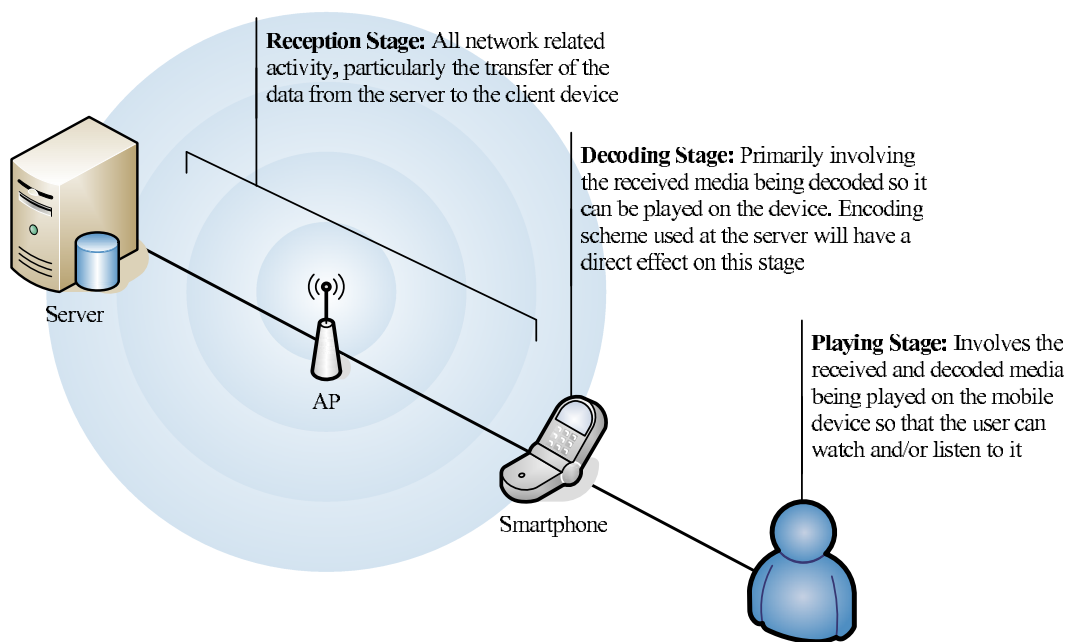


Figure 2.2: Multimedia Streaming Process

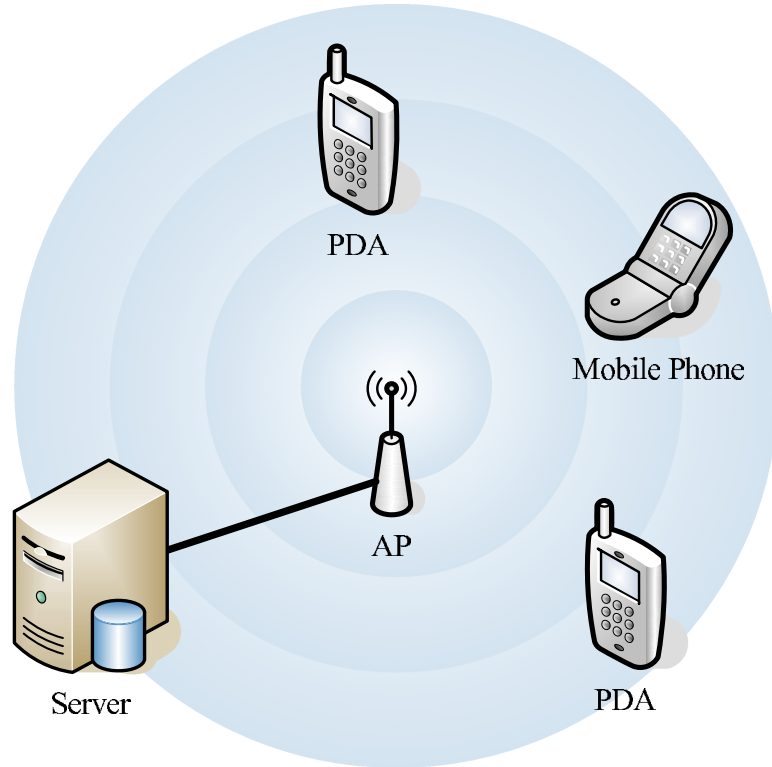


Figure 2.3: Client Server System Architecture

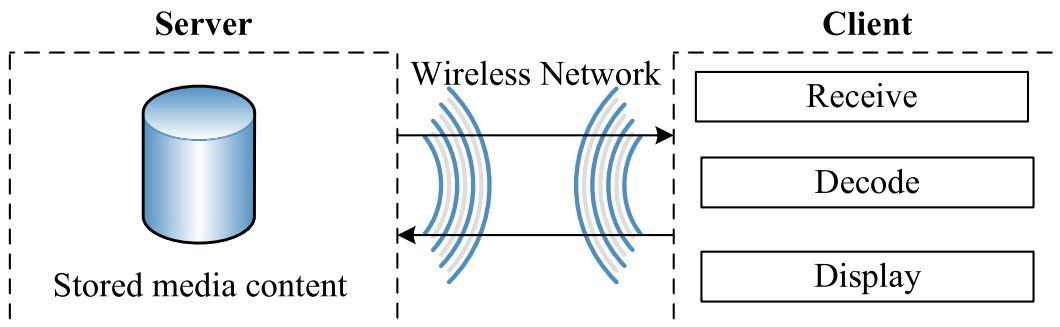


Figure 2.4: System Architecture - Block Diagram

providing details of what is contained within the client and server.

2.2.2 Reception

A number of transmission protocols exist that could be used during the reception stage of the multimedia streaming process. These are described below.

User Datagram Protocol

The User Datagram Protocol (UDP) [11] allows networked computers to send short messages to each other. These messages are known as datagrams. UDP is an unreliable protocol and if packets are dropped or arrive in an incorrect order, the transmission still continues. It is unreliable in that packets are not reordered and no acknowledgements are sent. This makes UDP very efficient and suitable for the streaming of multimedia content [12] and other time-sensitive applications.

For the tests performed in this thesis, UDP was used in sending the data from the server to the client via the AP. UDP was chosen because of its suitability for mobile multimedia streaming.

Real-time Transport Protocol

The Real-time Transport Protocol (RTP) [13] is a standardised protocol for delivering audio and video content over the Internet. In general, it is used on top of the UDP protocol but the standard does not state this as being compulsory. Originally developed in 1996, the standard was replaced in 2003 with an improved version. It is very commonly used in the streaming of real-time multimedia content, in conjunction with the Real Time Streaming Protocol (RTSP) [14]. RTP provides identification of payload type, time stamping, delivery monitoring and sequence monitoring. It does not provide any mechanism to ensure timely delivery or Quality of Service (QoS).

However, RTP enables a mechanism to verify that the packets are received in the correct order. The Real-time Control Protocol (RTCP), which is a network control protocol defined within the RTP specification, allows information about the quality of reception and this information can then be used to make any necessary adjustments.

Transmission Control Protocol

The Transmission Control Protocol (TCP) [15] allows for reliable exchange of packets between networked computers. Reliability is ensured by the receiver knowing what packets to expect and ordering them correctly. Acknowledgements are also used, adding to the reliability of the protocol. Due to its inherent inefficiency, which is a side effect of the reliability it provides, it is generally accepted that TCP is not the most suitable protocol for the streaming of real-time data [16]. UDP is generally preferred.

2.2.3 Decoding

In referring to the decoding stage, it is important to understand that the encoding scheme used will affect the power required to decode it. There are a variety of encoding schemes that can be used for the multimedia content. Some of the most common ones are briefly described below.

MPEG-1

The Moving Pictures Expert Group (MPEG) defined the MPEG-1 standard [17] in 1996, although development had been going on for many years before that. MPEG-1 refers to a group of coding and compression standards for audio and video.

The most commonly used audio of the MPEG-1 group of standards is MPEG-1 Layer 3, more commonly known as MP3. Throughout this thesis it will be referred to as MP3. This is a digital audio format that greatly

reduces the amount of data required to represent the digital audio. This means that, due to the small size of an MP3 clip, users can easily transfer them to their portable audio players and can fit a lot of music on one disk. A large number of bitrates are specified for MP3, allowing users to choose the quality of their media clip. This is especially relevant for downloading and storage, as a higher bitrate piece will take longer to download and will require more space to store it.

In MPEG-1 video, the major disadvantage is that it is limited to supporting only progressive pictures. These disadvantages were among the factors that prompted the development of MPEG-2.

For the audio tests performed in this thesis, the chosen encoding scheme was MP3. This is because it is extremely popular and very widely used, particularly on mobile devices which may have space limitations. By using what is popular and commonly used, the tests are very realistic and as close to real-life situations as possible.

MPEG-2

MPEG-2 [18] was standardised in 2000. The title of the standard is "Generic coding of moving pictures and associated audio information", which describes the idea behind MPEG-2. It is a combination of audio and video compression methods that allows storage and transmission of the media. MPEG-2 is very widely used. Some of the more common applications are terrestrial and cable television, direct broadcast satellite television systems such as Sky ² and a variant of MPEG-2 is used for the media distributed on Digital Versatile Discs (DVDs).

The video part of MPEG-2 improves upon that defined in MPEG-1, allowing for interlaced video streams. This is used in High Definition Television (HDTV), although some adjustments are made to it first. The audio part of MPEG-2 enhances that defined in MPEG-1 by allowing for audio with more

²www.sky.com

than two audio channels to be coded.

MPEG-4

MPEG-4 [19], defined in 1998 and standardised in 2002, is used to compress digital audio and video data. The main uses for it are Internet media streaming and broadcast television. These applications benefit from the combined audio and video stream.

To this day, MPEG-4 is still being developed. However, some of the technologies that it provides and who they benefit will now be discussed. For developers, MPEG-4 allows more stringent control of content and better security for copyrighted material. Network providers can use MPEG-4 for data transparency, making it easier to use signals with any available network. End-users can make use of the wide range of interaction possibilities.

MPEG-4 is capable of streaming audio and video data from files, cameras, microphones etc. It is capable of multiplexing and synchronising the audio and video data and allows interaction on the side of the receiver with the audio-visual scene. MPEG-4 is extremely flexible, providing a number of different profiles, which means that the entire standard need never be implemented.

MPEG-4 was used for the video tests in this thesis. The reason for choosing this particular video encoding scheme is that it is popular and widely used and is suitable for wireless streaming to mobile devices.

H.263

H.263 [20], is a video codec that was designed around 1996 and standardised in 2002 by the ITU Telecommunication Standardization (ITU-T) of the International Telecommunications Union (ITU). It is a low bitrate, compressed format, originally designed for video-conferencing. It is part of the H.26x family of the Video Coding Experts Group (VCEG). Most flash video

content websites such as YouTube ³ and Google Video ⁴, use this format. With the development of H.264, H.263 is now really considered to be the legacy standard, with H.264 improving upon it.

H.264

H.264 [21], is a video coding standard that achieves very high levels of data compression. The first version of the standard was finalised in 2003. It is technically identical to Part 10 of the MPEG-4 standard, and they are jointly maintained to keep them the same. The original goal was to create a video coding standard that could give good quality at low bitrates and without being too difficult to implement. It was also a goal that the standard could be easily applied to a large range of applications, networks and systems. H.264 allows video to be compressed far more effectively than older standards, such as MPEG-1 and MPEG-2. It also provides flexibility that allows it to work in a wide variety of environments.

2.2.4 Playing

For the playing stage of the multimedia streaming process, there are a number of media players that could be used. Some of the main ones are described below.

VLC Media Player

The VideoLAN project ⁵ developed the free, open source, software media player known as VLC media player ⁶. The main benefit of this media player is that is very portable, supporting many codecs and file formats, such as MPEG-1, MPEG-2 and MPEG-4. It is a player, an encoder and a streamer.

³www.youtube.com

⁴video.google.com

⁵www.videolan.org

⁶www.videolan.org/vlc

It can stream over networks and also has the ability to transcode multimedia files to different formats. It can be ported to any operating system, including those found on mobile devices.

VLC was the media player used for the tests which were performed in this thesis. It was chosen for its portability (it was used on both the client and the server) and its support of the relevant media encoding formats.

Windows Media Player

Microsoft's Windows Media Player ⁷ can be used to play audio, video or images in a number of formats including but not limited to MPEG-1, MPEG-2 and MPEG-4. It can only be run on computers using the Microsoft Windows operating system and on Windows based mobile devices.

RealPlayer

RealPlayer ⁸ is a media player designed by RealNetworks ⁹ that allows the playing of a number of different media formats such as some of the MPEG standards, on a number of different platforms like Microsoft Windows, Apple Mac and many more.

Quicktime Media Player

Apple's media player is called Quicktime ¹⁰. Quicktime is capable of handling a number of formats for digital video, media clips, sound, text, animation, music and a number of interactive panoramic images types. It is used in conjunction with the Apple iTunes music player ¹¹. It can run on the Apple Mac operating system and on Microsoft Windows.

⁷www.microsoft.com/windows/windowsmedia

⁸europe.real.com

⁹www.realnetworks.com

¹⁰www.quicktime.com

¹¹www.apple.com/itunes/

WinAmp Media Player

The WinAmp media player ¹² is a proprietary media player. It can be used with multiple media formats. It is the second most actively used media player, with Windows Media Player being the first.

2.3 Wireless Solutions

Wireless communication has also grown in popularity in recent years. Ireland has the most access points per head of population at 18.3 per 100,000 people ¹³. It is developing so that eventually, entire cities will be covered by wireless access points, and eventually seamless handovers may be provided from one to the next so that uninterrupted wireless access can be achieved. This will in turn facilitate the growth and popularity of wireless multimedia streaming.

Many wireless communication solutions over which mobile devices can connect are available. The next sections will discuss some of the most common of these solutions, with a brief technical description of each and details of their applications.

2.3.1 WiFi

WiFi is a generic term, used to describe the interface used by mobile devices to connect to a WLAN. Originally however, the term referred more specifically to the WLAN's of the IEEE 802.11 standard [5] [9] [22] [23] [24] [25]. These standards were developed to allow wireless access between compatible devices. The first standard was ratified in 1997 and since then, the growth and development has been significant. The majority of laptops come equipped with WiFi capability. Most PDAs and some mobile phones also have WiFi connectivity.

¹²www.winamp.com

¹³www.ofcom.org.uk/media/news/2006/11/nr.20061129

It has now become quite common for regular home broadband users to have a WiFi network set up in their home. This has developed with the increased penetration of broadband Internet access in homes. With an increasing number of people having laptops and PDAs, the WiFi network allows them to have Internet access on any WiFi compatible device anywhere in their home. This has become so popular that some broadband service providers, such as Eircom ¹⁴ in Ireland, offer a free wireless access point with a broadband subscription.

WiFi has also gained popularity with gaming. Game consoles such as the Playstation Portable ¹⁵ and the Nintendo DS ¹⁶ have WiFi capability. Sometimes, this allows playing of games using WiFi, other times it allows the user to connect to their home WiFi network and have Internet access.

WiFi has also become widely used in educational institutions. Many universities and schools offer complete WiFi access. This means that wherever a user goes within the campus, they can connect to the local WiFi service and have Internet access. Similar setups can also be found in large businesses. The use of WiFi hotspots has also grown. This is when WiFi is offered in a public place such as a café and a user can connect to it using their mobile device, usually for a small cost.

For the tests in this thesis, IEEE 802.11b was used. This wireless standard is described in greater detail in the following chapter. The reason that it was chosen is that there are a large number of compatible devices making tests realistic and a good estimation of what a user might really use.

2.3.2 WiMAX

Worldwide Interoperability for Microwave Access (WiMAX), or IEEE 802.16 [26], was developed to provide wireless data over long distances in a number

¹⁴www.eircom.ie

¹⁵ie.playstation.com/psp

¹⁶www.nintendo.com/systemsds

of different ways. WiMAX operates at frequencies between 2GHz and 66GHz, although recent revisions offer more stringent guidelines. It is a long range technology, with a possible range of up to five miles. It can achieve data rates of about 100Mbps. Originally WiMAX was a fixed wireless technology, however in late 2004, IEEE 802.16e [27] was introduced, which allows for pedestrian mobility and regional roaming.

Due to its high bandwidth and large range, WiMAX has many possible uses. It can connect Wi-Fi hotspots to each other or connect them to other parts of the Internet. It can provide an alternative to Digital Subscriber Line (DSL) or cable for broadband access, particularly for the last part of the connection, between the broadband exchange and the users home. WiMAX can provide high-speed data and telecommunications services for both personal and business use.

2.3.3 Bluetooth

Bluetooth ¹⁷ is a wireless technology intended to replace cables between two communicating devices while maintaining a high level of security. Also known as IEEE 802.15.1 [28], Bluetooth is a globally accepted technology and a device with bluetooth capability can connect quite simply to virtually any other device with bluetooth connectivity. Bluetooth can handle voice and data transmissions and this has been very instrumental in its success.

Bluetooth operates in the 2.4GHz unlicensed spectrum. It can achieve a range of 10 meters, or up to 100 meters for higher classes. The highest data rate achievable is 3Mbps. It can penetrate walls and other solid objects, is omni-directional, does not require line of sight positioning and allows for three levels of security. It is also relatively inexpensive.

Due to all the advantages mentioned above, Bluetooth has a wide number of applications. It is used to connect wireless earpieces to mobile phones, for wireless peripherals such as mice, keyboards and printers, to send data be-

¹⁷www.bluetooth.com

tween two computers and is also now being used for game console controllers, such as those with the Nintendo Wii ¹⁸.

2.3.4 Ultra Wideband

Ultra Wideband (UWB) [29] is a Wireless Personal Area Network (WPAN) that enables wireless connectivity across multiple devices and computers, while maintaining a consistently high data rate. It was created in order to provide for emerging devices and the want of the user to be fully connected. The typical user will want all of their personal devices, i.e. music players, mobile phone, digital cameras, etc. to be able to connect to each other within a specified area, for example, their home. This can not be achieved by the current Local Area Network (LAN)'s and WLAN's.

UWB uses a very wide band of the radio frequency spectrum, making it different from other wireless technologies. UWB uses frequencies between 3.1GHz and 10.6GHz. UWB is intended for short range, indoor use and the potential data rate is based on the channel bandwidth used. Each radio channel can have a potential bandwidth of 500MHz but broadcast power restrictions are in place to avoid interference.

Some of the applications of UWB include replacing firewire cables in personal multimedia devices with wireless connectivity, enabling high-speed Wireless Universal Serial Bus (WUSB) connectivity to replace current Universal Serial Bus (USB) connections on peripherals such as printers and keyboards and creating an ad-hoc, high bit-rate wireless network to allow for a WPAN among all personal devices in a small area, such as a house, similar to that shown in Figure 2.5.

¹⁸www.wii.com

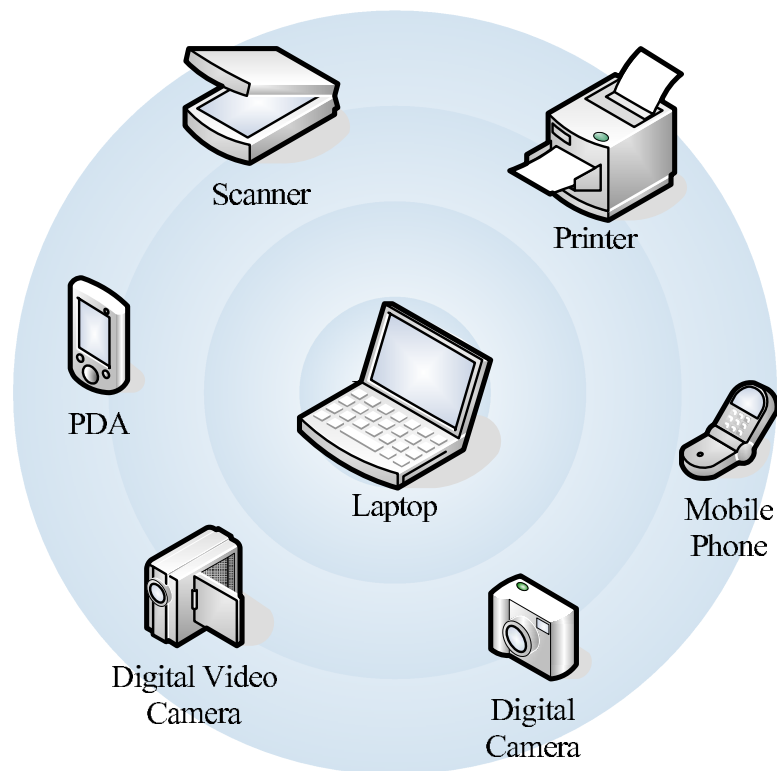


Figure 2.5: Typical UWB WPAN

2.4 Summary of Chapter

This chapter presented the background information that is required to understand the remainder of the thesis. The idea of the multimedia streaming process and its stages, which will be used throughout the thesis, was explained. The most popular encoding schemes, transmission protocols, media players and wireless solutions were then discussed and those chosen for use in this thesis were highlighted.

Chapter 3

IEEE 802.11 Overview

IEEE 802.11 refers to a set of WLAN standards developed by Working Group 11 of the IEEE LAN/MAN Standards Committee (IEEE 802). This chapter briefly describes the IEEE 802.11 WLAN standard family [5]. A brief overview of the original 802.11 standard is given first, including all power related issues and solutions. Next, brief descriptions of each of the protocols that extend the original proposal are presented. Later chapters present work that use 802.11 technology.

3.1 IEEE 802.11 Architecture

The basic building block of the IEEE 802.11 architecture is a Basic Service Set (BSS), which is defined as a set of Stations (STAs) controlled by a single coordination function, which could be a Point Coordination Function (PCF) or a Distributed Coordination Function (DCF). A coordination function determines when a STA can transmit within the BSS. In DCF, the same coordination function logic is active in every STA of the BSS whereas in PCF, the coordination function logic is only ever active in one STA of the BSS at any one time. Both PCF and DCF can operate at the same time.

The 802.11 architecture allows for two types of operational modes: Inde-

pendent BSS (IBSS) and infrastructure. The most basic type of 802.11 LAN is an IBSS, which is sometimes referred to as an ad-hoc network. This mode, shown in Figure 3.1, enables direct communication between STAs and is often formed without being pre-planned and for temporary use. In an IBSS any STA can communicate with any other STA without the need for an AP.

In the infrastructure mode, shown in Figure 3.2, an AP exists and all communication goes through this AP. The work in this thesis will use the infrastructure mode. Infrastructure networks can be extended, thus providing an increased range, by integrating multiple BSS's using a common Distribution System (DS). This is known as an Extended Service Set (ESS).

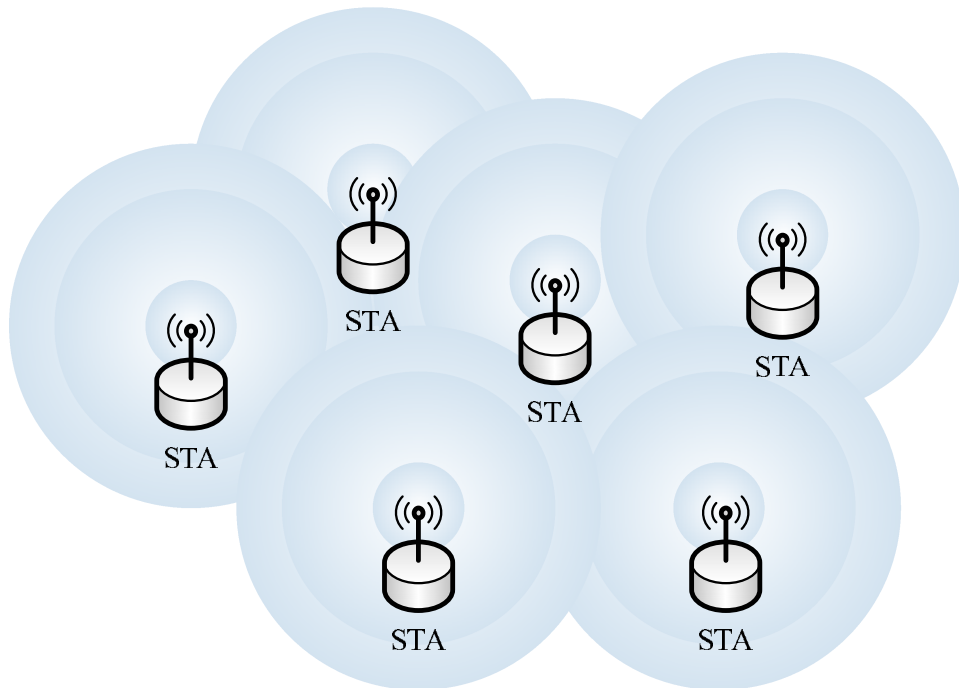


Figure 3.1: 802.11 Ad Hoc Architecture

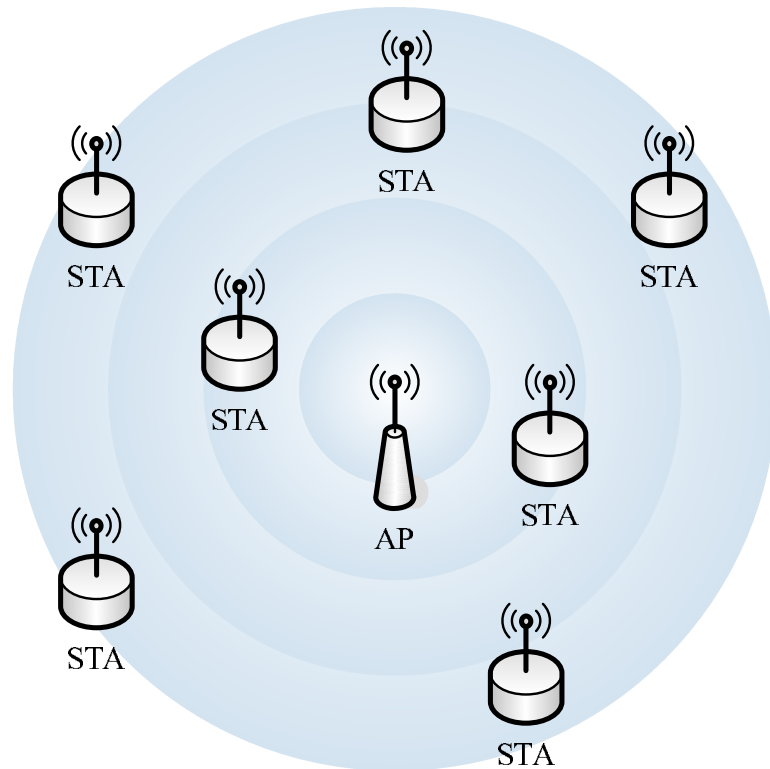


Figure 3.2: 802.11 Infrastructure Architecture

3.2 Point Coordination Function

PCF provides contention-free data transfer. In PCF, a Point Coordinator (PC) is present in the AP and this PC controls when STAs can transmit at any given time. During a Contention Free Period (CFP), the PC will allow a STA operating in PCF mode to transmit. While a STA is transmitting, no other STA can transmit. Once the STA finishes, the next one can start.

3.3 Distributed Coordination Function

In 802.11, PCF is optional and rarely implemented. This section will explain in detail how DCF works. The principle DCF uses in 802.11 is Carrier Sense

Multiple Access with Collision Avoidance (CSMA/CA). This is implemented in all STAs within both types of configurations i.e. ad-hoc and infrastructure. When a STA wants to transmit, it first senses the medium to see if another STA is transmitting. If the medium is free then the STA can commence its transmission. CSMA/CA specifies that there must be a minimum duration between adjacent frame sequences. The STA will wait for this amount of time after deeming the medium idle before commencing transmission. If the medium is busy, the STA will wait until the current transmission is complete, then select a random back off interval which will be decremented while the medium is idle.

Further to this, it is possible to also use short control frames, known as Request to Send (RTS) and Clear to Send (CTS), after determining that the medium is idle and after any waits or back offs. The RTS/CTS frames contain a duration/ID field which defines the time required to transmit the data frame and the returning Acknowledgement (ACK) frame. The STA sends an RTS which is broadcast to all STAs in the range and contains a field which specifies the intended recipient of the data packet. When the intended recipient receives the RTS it responds with a CTS and the STA can proceed with the transmission. To further prevent collisions, Network Allocation Vectors (NAVs) are used. When all STAs in the BSS see the RTS, they set a NAV which is basically a time in which they cannot send. This means that they will not collide with the transmission. All the STAs also see

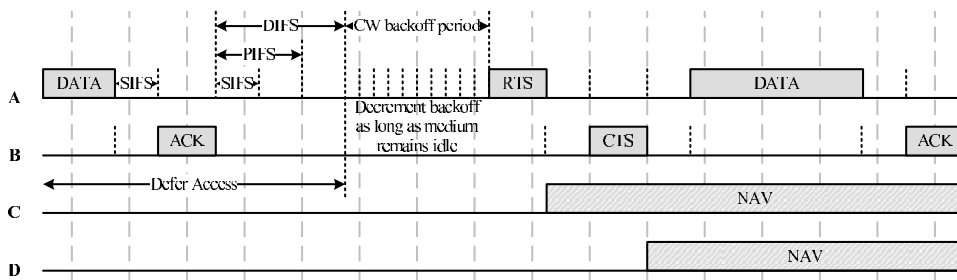


Figure 3.3: Distributed Coordination Function

the CTS and set a NAV then too. This is to allow for any STA that did not see the RTS to avoid collision. A timing diagram representing DCF is shown in Figure 3.3.

3.4 Amendments to the Standard

The original standard needed improving as it had low data rates (approximately 2Mbit/s with only about 0.9Mbit/s throughput) and a short range. Since the release of IEEE 802.11 there have been various amendments made to it. Some of them increase speed or range while others aim specifically at various aspects such as security, voice etc. In this thesis, 802.11b will be the focus. Some of the amendments increased the speed, range or both. 802.11b, a, g and n are all examples of this. 802.11e provided for quality of service. Other amendments were also included and are listed in Table 3.2. For this work, 802.11b was chosen. However, the algorithm could easily be implemented on any of the amended standards, once power saving is handled in the same way.

3.4.1 IEEE 802.11b

The 802.11b [9] amendment to the original standard was ratified in 1999 . 802.11b has a maximum raw data rate of 11 Mbit/s and like the legacy standard, uses CSMA/CA. Due to the overhead incurred by CSMA/CA, the actual achievable throughput is quite lower, about 5.9 Mbit/s using TCP and 7.1 Mbit/s using UDP. 802.11b works in the 2.4GHz band, which due to the popularity of this frequency band, can sometimes lead to interference problems.

Due to the similarity that 802.11b had with the original standard, the equipment was inexpensive and easy to produce and therefore was released to the market relatively quickly. As 802.11b had a high throughput (compared to original standard) and was inexpensive to implement, it became popular

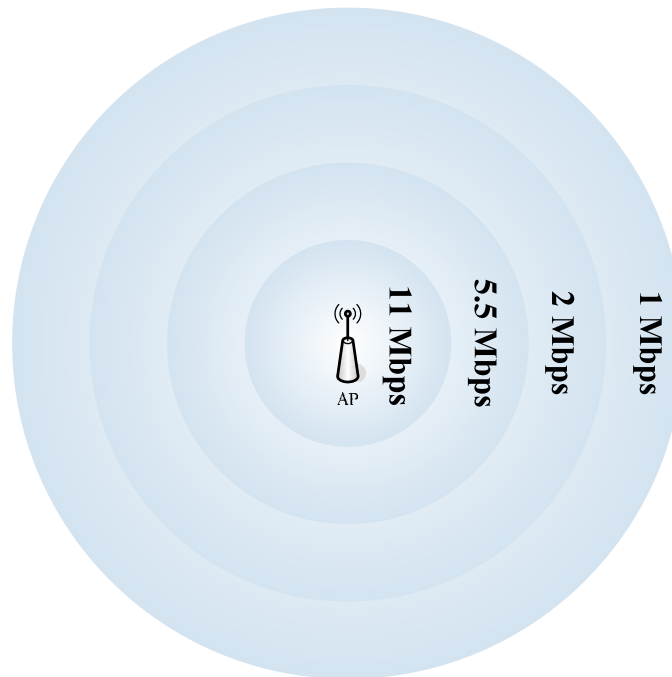


Figure 3.4: 802.11b Decreasing Signal Strength & Bandwidth

very quickly. It is the most widely accepted of the WLAN standards to date. 802.11b is usually used in an infrastructure configuration, with all STAs communicating through an AP. The range is approximately 150 feet indoors and 300 to 500 feet outdoors. 802.11b can operate at 11Mbit/s but as the signal becomes weaker it drops this rate first to 5.5 Mbit/s, then 2 Mbit/s and finally to 1 Mbit/s, as shown in Figure 3.4.

3.4.2 IEEE 802.11a

802.11a [22] was ratified in 1999. Using the same base protocol as the original standard, it operates in the 5GHz band and has a maximum data rate of 54Mbit/s (in reality, the achievable throughput is approximately 20Mbit/s). 802.11a is not interoperable with 802.11b unless the equipment being used is specifically designed for this purpose.

The main advantage of 802.11a is that the 5GHz band has less interference. However, the use of 802.11a is reduced to almost line of sight and because the range is not as good as 802.11b, more access points are required. The equipment for 802.11a was released after the equipment for 802.11b due mainly to the difficulty in acquiring the 5GHz components. By the time 802.11a was released, 802.11b was already widely used and as it was less expensive and had a larger range, people were unwilling to change. In response to this, 802.11a was improved so that the range almost equalled that of 802.11b and technology allowed more than one 802.11 standard to be used. This means that a lot of equipment now available supports 802.11a but it is still not as popular as 802.11b.

3.4.3 IEEE 802.11g

802.11g [23] was introduced in 2003. Like 802.11b, it works in the 2.4GHz range but it achieves a maximum data rate of 54Mbit/s, which is the same as 802.11a. It is backwards compatible with 802.11b and both standards work well together. The range of 802.11g is similar to that of 802.11b but to achieve the maximum advertised rate, it is necessary to be quite close to the AP. When 802.11g was ratified, the devices that had been dual band to allow for 802.11a and 802.11b became tri-band to support 802.11g as well. Although it has a higher data rate, 802.11g is subject to the same interference problems as 802.11b because of it being in the more popular 2.4GHz band.

3.4.4 IEEE 802.11n

The latest addition to the 802.11 family is 802.11n [25]. Due to be ratified in approximately early 2008, it is reported to have a theoretical output of 540 Mbit/s making it up to fifty times faster than 802.11b. 802.11n adds Multiple-Input Multiple-Output (MIMO) which uses multiple transmitter and receiver antennas to allow for increased data throughput. It will operate

in the 2.4 and 5GHz spectrum and will have a typical range of 150 feet.

With the introduction of 802.11n, there will be three modes in which an 802.11 access point can operate:

- Legacy - 802.11a, 802.11b and 802.11g
- Mixed - 802.11a, 802.11b, 802.11g and 802.11n
- Greenfield - 802.11n only (this mode gives maximum performance)

3.4.5 IEEE 802.11e

802.11e [24] is an amendment to the standard that is specifically intended for QoS enhancements. This addition to the standard is extremely important for delay-sensitive applications, particularly for the streaming of multimedia content. The enhancements are made to the Media Access Control (MAC) layer of the Open Systems Interconnection (OSI) model. 802.11e was approved in late 2005.

As described above, the legacy 802.11 uses DCF. However, some limitations are presented when DCF is used:

- If multiple stations communicate simultaneously then collisions will occur, which will reduce bandwidth
- If a station that has a low bit rate receives access to the medium and can therefore transmit, all other stations must wait until it is finished, which will take a long time when bit rate is low
- There is no prioritisation of traffic or QoS guarantees

As mentioned previously, PCF is optional and is not present on all equipment. However, when it is used, it defines two periods between beacon frames: the CFP and the Contention Period (CP). When DCF is used, only

the CP is present. In CFP, the AP sends a message to each STA individually to tell them the medium is free and they can send. These messages are known as Contention-Free Polls (CF-Polls). However, as mentioned, PCF is rarely used.

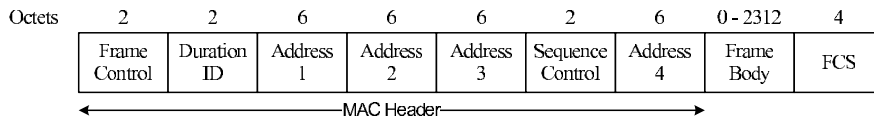
In 802.11e, a new coordination function is used which is called the Hybrid Coordination Function (HCF). In HCF, two methods of channel access are used: the HCF Controlled Channel Access (HCCA) and the Enhanced DCF Channel Access (EDCA). Traffic classes are defined within both. This allows for prioritisation of traffic. Each priority level is assigned a Transmit Opportunity (TXOP) which is an interval of time that the STA can use to send as many frames as possible. Frames that are too large to be sent in a TXOP are fragmented. While EDCA and TXOP are mandatory for APs, all other 802.11e enhancements are optional.

3.5 Power Saving in IEEE 802.11 Networks

The reception stage is the most significant power drainer in the streaming process due to the fact that the WNIC 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 PSM [5]. An AP will maintain a Power Management Status for each currently associated STA. STAs using Power Management mode inform the AP by using the Power Management bits within the Frame Control Field of transmitted frames. These frames are shown in Figure 3.5 The AP will not automatically transmit data to STAs in PSM but rather will buffer that data and transmit later. A beacon is an 802.11 frame that contains control information. All STAs, including those in PSM, will listen to beacons to check if there is traffic for them. How often a STA listens to the medium for beacons is determined by the ListenInterval. This is usually set to the same as the beacon interval, at

MAC Frame Format



Frame Control Field

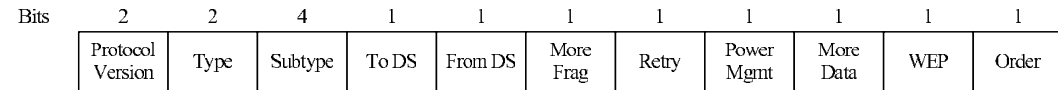


Figure 3.5: 802.11 Frame Formats

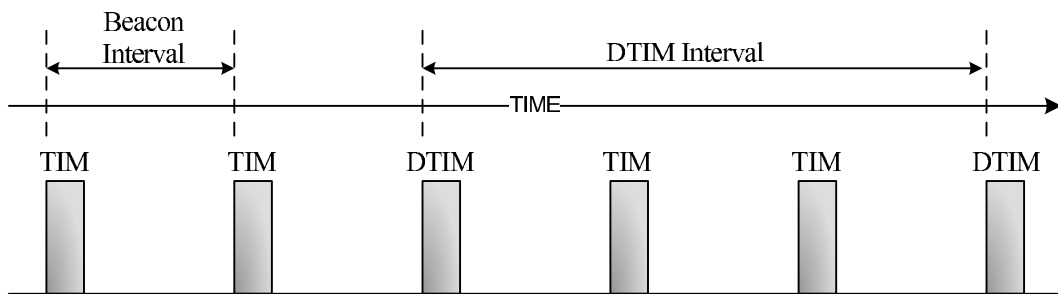


Figure 3.6: 802.11 TIMS and DTIMS - Example with DTIM sent at every three TIM intervals

the default value of 100ms, which means that STAs listen to every beacon.

The STAs that currently have buffered data within the AP are identified in a Traffic Indication Map (TIM), which is included in all beacons generated by the AP. There are two types of TIM: TIM and Delivery TIM (DTIM), as shown in Figure 3.6. A DTIM is sent less frequently, perhaps every third TIM will be a DTIM. After a DTIM, the AP shall transmit any broadcast or multicast traffic before transmitting any unicast traffic. STAs operating in Power Save (PS) mode shall periodically listen for beacons. This period is determined by the STA's ListenInterval parameter.

A STA's Power Management mode determines the manner in which the STA transitions between the possible power states. There are two power states that a STA may be in:

- Awake: the station is fully powered
- Doze: the station is asleep so it can not transmit or receive and consumes very low power

There are also two Power Management modes:

- Active Mode (AM): The station can receive frames at any time and will be in the Awake state.
- Power Save (PS): The station listens to selected beacons (based on the ListenInterval parameter) and sends PS-Poll frames to the AP if the TIM element in the most recent beacon indicates that there is buffered data at the AP for that station. The AP will transmit buffered data directed to a PS station only in response to a PS-Poll from that station. In PS mode, a station is in the Doze state and shall enter the Awake state to receive selected beacons, to receive broadcast and multicast packets and to await responses to transmitted PS-Poll frames.

When a STA that is operating in PS mode determines that there is buffered data for it at the AP, the STA will transmit a short PS-Poll frame to the AP, which shall either respond with the corresponding buffered data immediately or acknowledge the PS-Poll and respond with the corresponding data at a later time.

To change Power Management modes, the STA informs the AP through a successful frame exchange initiated by the STA. The Power Management bit in the Frame Control Field of the frame sent by the STA in this exchange indicates the Power Management mode that the STA shall adopt upon successful completion of the entire frame exchange.

Any buffered data for a STA is indicated by the TIM, which is contained within the beacon. The TIM also indicates whether there is broadcast or multicast traffic waiting. This legacy PSM is shown graphically in Figure 3.7. The first step of the process is the STA informing the AP that it is

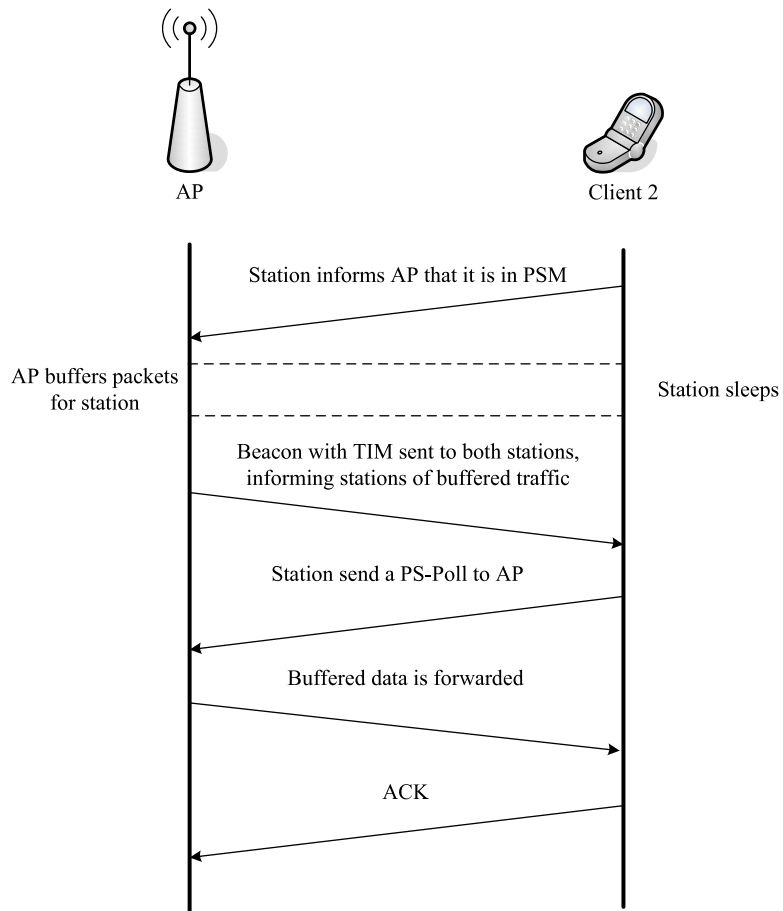


Figure 3.7: IEEE 802.11 PSM

using PS mode. The AP then buffers any packets it receives for that STA. In the next beacon, the included TIM will inform the STA that there is buffered traffic buffered at the AP for it. The STA sends a PS-Poll to the AP to say that it is awake and will receive the buffered traffic. Once received, the STA sends an ACK to the AP. The process can then start again.

Although this PS mode does save power in mobile STAs, the savings are minimal. This is because the ListenInterval is not used in the most power effective way. The ListenInterval determines whether or not the STA wakes for every beacon or whether it sleeps through some of them. This is rarely

changed from the default value of setting the STA to wake for every beacon. A significant disadvantage of using the ListenInterval to save power is that it must be set on association with the AP and therefore cannot be changed adaptively.

In [30], Kwon et al carry out experiments to investigate the performance effects and energy saving of the IEEE 802.11 PSM. They measure the throughput and performance time of a STA working in PSM with various applications and compared them to those working in active mode. Their results show that this PSM successfully saves energy when in a heavy and continuous traffic situation. They also illustrate that an MPEG streaming application suffers little decrease in performance when in PSM.

3.6 Power Saving in IEEE 802.11e

Within the 802.11e amendment to the 802.11 standard, an improvement to the power save mechanism is proposed. This optional extension of the legacy PSM is called Automatic Power Save Delivery (APSD). If it is supported by the AP, then a STA that has packets buffered for it at the AP can choose to have them delivered by the standard PSM or by APSD. 802.11e introduces the concept of a Service Period (SP). An SP is a time in which one or more unicast frames are transmitted to an 802.11e STA and/or one or more TXOPs are granted to the same STA. SPs can be scheduled or unscheduled and for a non AP STA, there can only be one SP active at any time.

There are two types of APSD, Scheduled APSD (S-APSD) and Unscheduled APSD (U-APSD), which relate directly to the two types of SP. However, as U-APSD is defined only for STAs accessing the channel using EDCA and S-APSD are defined for both EDCA and HCCA, it is more likely for EDCA and hence U-APSD to be used [31] so it is U-APSD that will be described here.

U-APSD is based on the idea that data frames sent by STAs as indicators,

known as triggers, that a STA in power save is awake. The AP then takes advantage of this by delivering any buffered data to the STA. The method is specifically designed for bi-directional streams of traffic. The main difference between APSD and the legacy PSM is that in the legacy PSM the STA must stay awake from the time it wakes to the time it has finished receiving all of its buffered data whereas in APSD, the STA is only awake during the SP.

The receipt of a trigger frame by the AP begins an unscheduled SP, which ends when the STA sends a QoS null frame which indicated what is known as the End of Service Period (EOSP). An EOSP is similar to the PS-Poll frames used in the legacy standard. Each STA can be configured as either trigger-enabled or delivery-enabled. When delivery-enabled, the AP is allowed to use EDCA to deliver traffic to a STA during an unscheduled SP triggered by a STA. When a STA is trigger-enabled, the QoS data and QoS Null frames trigger an unscheduled SP. The maximum SP length is a field within the QoS info field filled by the STA at association.

Although the APSD defined in the IEEE 802.11e standard offers significant improvement over the legacy PSM, there are some disadvantages that must be highlighted. The first and most obvious one is that as 802.11e is optional, APSD can only be used on an 802.11e enabled AP and STAs (i.e. network). The second disadvantage is that even if 802.11e is present, APSD is optional within it. This means that even if an AP is 802.11e enabled, it may not use APSD. Thirdly, APSD is optimised for traffic which has similar uplink and downlink characteristics, for example Voice over IP (VoIP). This is not ideal for multimedia streaming. The scheme proposed in this thesis is better suited to multimedia streaming tasks.

3.7 Summary of Chapter

This chapter described the IEEE 802.11 standard and its major amendments. It also discussed their characteristics. The power save mechanisms within

them were then described. Table 3.1 summarises the main features of the different 802.11 standards. Table 3.2 lists the other amendments which are in different stages of adoption.

Table 3.1: 802.11 Family

Protocol	Release Date	Frequency	Typical Rate	Max Rate
Legacy	1997	2.4 GHz	1 Mb/s	2 Mb/s
802.11a	1999	5 GHz	25 Mb/s	54 Mb/s
802.11b	1999	2.4 GHz	5.5 Mb/s	11 Mb/s
802.11g	2003	2.4 GHz	25 Mb/s	54 Mb/s
802.11n	2008	2.4 or 5 GHz	200 Mb/s	540 Mb/s

Table 3.2: 802.11 Amendments

Name	Year	Description
802.11c	2001	Bridge operation procedures
802.11d	2001	International roaming extensions
802.11e	2005	Quality of service enhancements
802.11F	2003	Inter-Access point protocol (withdrawn)
802.11h	2004	European compatible 802.11a
802.11i	2004	Security enhancements
802.11j	2004	Enhancements for Japan
802.11k	2007	Radio resource measurement enhancement (proposed)
802.11l	****	Reserved, will not be used
802.11m	****	Ongoing maintenance of standard
802.11o	****	Reserved, will not be used
802.11p	2009	WAVE - Wireless access for vehicular environment (proposed)
802.11q	****	Reserved, will not be used
802.11r	2007	Fast roaming (proposed)
802.11s	2008	ESS Mesh networking (proposed)
802.11T	2008	WPP - Wireless Performance Prediction (proposed)
802.11u	????	Internetworking with non 802.11 networks (proposed)
802.11v	????	Wireless network management (proposed)
802.11w	2008	Protected management frames (proposed)
802.11x	****	Reserved, will not be used
802.11y	****	3650 - 3700 Operation in the US (proposed)

Chapter 4

Literature Review

This chapter will present work related to this thesis, which can be divided into a number of sections. First the solutions proposed in the general area of power saving in multimedia streaming are discussed. Then the schemes proposed for power saving in each of the three stages of the multimedia streaming process are described.

4.1 Power Saving in Multimedia Streaming

The area of power saving during multimedia streaming is one that has been thoroughly researched. There have been many different approaches to it, which can make classification difficult. However, broadly speaking, the majority of approaches fall into the following categories: Dynamic Voltage Scaling (DVS), controlling the burstiness of network traffic, memory optimisation and compiler and/or application efficiency.

In [32], *Acquaviva, Benini and Ricco* focus specifically on embedded devices that are designed with multimedia streaming in mind. The authors propose to save energy by allowing an application to dynamically reconfigure the system so that the performance levels and required services are provided without wasting power. They highlight the possibility to reduce power con-

sumption by reducing the speed and supply voltage to the minimum level necessary in order to match real-time constraints. While other researchers [33] observe that frequency scaling decreases power but not energy, unless it is coupled with supply voltage scaling, Acquaviva et al demonstrate that energy can be saved by frequency scaling, even with a constant voltage supply. They achieve this by obtaining a set of three normalised curves: the overall best frame rate, the overall worst frame rate and the overall average frame rate. These curves relate multimedia streaming application performance with processor clock frequency. The authors develop an algorithm that controls the processor speed in order to save power, while ensuring that any real-time constraints are satisfied. As the application provides the information required to set the processor speed, this is an application-driven method of frequency scaling. Another application driven frequency scaling method can be found in [34], where they concentrate on user experience when the energy level is low and then when the energy level drops, the focus is towards energy conservation. Their results show up to 30% energy savings. Their solution is based on the Linux operating system and has not been tested on other platforms. An additional application driven frequency scaling method is presented in [35], where the authors present an energy model for software which separates the voltage switching and voltage leaking components and predicts the total energy consumption. Some operating-system level schemes can be seen in [36], [37] and [38]. Acquaviva et al argue that in comparison to other frequency scaling schemes, theirs is robust and does not suffer from poor stability and poor real-time performance problems. Their results show that they can achieve up to 40% saving with respect to fixed-frequency operation while satisfying real-time playback constraints.

Korhonen and Wang study the effect of burst length and peak transmission rate for observed packet loss and delay characteristics and also potential energy savings in the WLAN [39]. They argue that power related research in the area of wireless multimedia can be divided into two princi-

pal categories: power-aware multimedia decoding, where the power save is implemented when the power level drops, such as in [40] and [41] and power-efficient radio communications where power-efficient methods are used from the outset, as in [42], [43] and [44]. The authors state that streaming applications typically transmit a constant stream of packets with periodic arrival times. For this reason, the legacy power save mode of 802.11 is not suitable as the gaps between the packets are so short that the station would never actually enter sleep mode. To counter this problem, Korhonen and Wang propose a solution that employs a layered multimedia coding scheme with a customised packet scheduler which provides bursty traffic with a decreasing priority order of packets in each burst. The scheme allows the receiver to sacrifice some of the enhancement layer data in order to maintain stable power efficiency. They then implement a test system incorporating this adaptive burst length. This achieves an improved trade-off between power efficiency and congestion tolerance. One of the major disadvantages of this scheme is that the role of CSMA/CA, as used in 802.11, is omitted. It is therefore very difficult to implement this solution on current 802.11 devices.

In [45], *Anastasi et al* look specifically at streaming of audio content to a mobile device that is connected to the Internet by means of a wireless connection. They provide a solution to the power consumption problem of mobile devices dealing with multimedia streaming. Their solution allows the WNIC to reduce to between 9-24%, depending on various conditions, of the energy consumption achieved by current systems. They do this by including periodic transmission interruptions in the schedule of the frames (in their case all frames are audio) at the server so that the WNIC can be set to sleep. However, they do not integrate their solution with the legacy 802.11 PSM, instead opting to disable the legacy power save. Their reasons behind this are that the legacy PSM only performs well when used with regular traffic [46]. Although this is true, it would perhaps make more sense to improve the legacy PSM to work with burst traffic rather than omitting it

completely. They conclude their paper with an interesting result: the higher the burstiness of the traffic, the more power savings their scheme achieves.

Mohapatra et al [47] present the interesting argument that there is a distinct lack of cross-level integration between the different power saving techniques that have been proposed in recent years. They state that the majority of the proposed power saving techniques can be attributed to one of the following categories: system cache, external memory access optimisation and DVS [48] [49], dynamic power management of disks and network interfaces, efficient compilers and application/middleware based applications [50] [51]. Mohapatra et al argue that there is a disconnect between all of these approaches and that the power saving capabilities of each could be improved by combining the different techniques. In order to do this, it is necessary to investigate the trade-offs involved and the customisation required. In order to address this potential cross-level integration, Mohapatra et al devise an algorithm that takes an integrated approach to power management. They do this by combining low level architectural optimisations, Operating System (OS) power-saving mechanisms and adaptive middleware techniques. The low-level architectural optimisations that they address include the CPU, the memory and the register. The OS power-saving technique was dynamic voltage scaling and the adaptive middleware technique included admission control, optimal transcoding and network traffic regulation. All of these techniques were then combined to make the integrated power save solution. Some knowledge of the system and device characteristics were required in order for the middleware technique to adapt the video to the appropriate quality. Their results show significant improvements with energy gains up to 57.5% for the CPU and memory and 70% for the WNIC. User experience was improved by the device lasting longer.

Zhu and Cao [52] use a scheduling algorithm at the base station called rate-based bulk scheduling which decides which flow should be served at which time. A proxy at the mobile terminal buffers data so that the WNIC

can sleep for longer periods. The sleep time is adjusted based on the channel conditions.

Other power saving schemes for the WNIC have been designed at different levels of the OSI network model. For example, work proposed for the MAC layer includes [53], [54] and [55] as well as the legacy PSM in IEEE 802.11. Some solutions proposed for the network layer are [56], [57] and [1] proposes a transport layer solution. The work proposed in this thesis is a solution that could be classified as lying between the transport and application layers.

4.2 Power Saving in the Reception Stage

The reception stage of the multimedia streaming process is anything that is related to the sending and receiving of data. Power saving in this stage generally focuses on the WNIC.

Chandra and Vahdat [46] 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 between data transfers, saving power. Although this scheme looks promising, it is not 802.11 friendly 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* [58]. 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 re-

sults 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.

In [59], *Zhang and Chanson* propose a scheme that uses traffic shaping with the added benefit of a proxy scheduling algorithm. The proxy groups the frames together and sends them in a burst instead of sending them individually. The proxy also determines the optimal scheduling algorithm in order to reduce the wait time for all clients. On examination of the results presented by the authors, the scheme seems to be successful. However, it focuses on multiple clients and it is difficult to assess the performance of each individual client. Furthermore, Zhang and Chanson only take into account the reception stage, not considering the decoding and playing stages of the multimedia streaming process at all.

In [60], *Wei et al* describe a client side prediction scheme. To save energy during multimedia streaming, the time intervals during which to suspend communications by switching the WNIC to a sleep state are predicted. The main problem with this approach to energy saving is that it relies heavily on prediction accuracy. If the predicted time durations are too short, then energy savings may be imperceptible. If the predicted intervals are too long or incorrect durations are predicted, the client may miss out on data. The prediction process is a learning one, and is very complex, which may in turn consume more power. It does not seem acceptable for a client to lose data, which would cause extreme user dissatisfaction in relation to multimedia quality during streaming.

4.3 Power Saving in the Decoding Stage

The decoding stage is when the device receives the data and decodes it to a recognisable format. However, the encoding of the data will impact the

amount of battery required for this stage. Power saving that has been proposed for this stage generally focuses on making the decoding of the data more efficient.

To save power in the decoding stage of the multimedia streaming process, *Pakdeepaiboonpol and Kittitornkun* [61] propose two schemes. Two high level power-saving techniques are described which are based on reducing the number of memory/bus accesses by high level language optimisation. 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 [62] propose to reduce the decoding power of multimedia by using feedback control. A controller adjusts the decoder's speed to keep the occupancy of the buffer between the decoder and the display constant. This effectively matches 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 are 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 assess the actual energy savings.

Lee [63] proposes a scheme that is based on dynamic voltage scaling, which can lower the supply voltage and reduce the power consumption. In the proposed scheme, the encoder counts the number of non-coded blocks in a frame and stores this information in the bit stream. This allows the decoder to correctly calculate the supply voltage after which the dynamic voltage can be applied. This scheme stops missed deadlines occurring, which often happens with dynamic voltage scaling. The results that are presented for the scheme show energy savings of about 10% in MPEG-4 video decoding. This is a good scheme but as it is hardware based, it would not be suitable for incorporation into the any software, cost-effective, power saving adaptive algorithm.

Mesarina and Turner [49] investigate the effect of dynamic voltage scaling on the trade off between energy consumption and high picture quality in multimedia decoding. They show that the use of dynamic voltage scaling reduces the energy consumption considerably. In their proposed offline scheduling algorithm, they assume that the order within each stream is fixed. The algorithm assigns a single voltage level per task and each task decodes a single media frame. They state that unless future clients have built-in power monitoring, the applicability of their algorithm is limited. This is a lower level solution than the one proposed in this thesis and is not adaptive as the user cannot choose what to adjust and by how much.

4.4 Power Saving in the Playing Stage

The playing stage involves the displaying of the media to the user. Depending on the media type, this stage will involve the screen, the speakers or a combination of the two. The majority of power saving in this area has focused on the screen of the device, in particular the backlight. To the knowledge of the author, there has been little research on the effect of the speakers on the battery.

Pascrich et al [64] propose an adaptive middleware-based approach to optimise backlight power consumption for mobile handheld devices when streaming MPEG-1 encoded video content.

Another backlight power management scheme is proposed by *Shim, Chang and Pedram* [65]. In this case, a backlight power management framework for colour TFT LCD panels is proposed. The authors extend Dynamic Luminance Scaling (DLS) to cope with transfective LCD panels, which operate both with and without a backlight, depending on the remaining battery energy and the ambient luminance. The scheme, known as Extended Dynamic Luminance Scaling (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.

Due to the fact there is generally a built-in brightness control on mobile devices, there has not been a lot of research performed in the area of power saving in the playing stage. In this thesis, the playing stage is concerned with the speakers and the screen. Both of these can be controlled by the user of the device and there is generally no automatic settings possibility.

4.5 Summary of Chapter

This chapter reviewed the literature related to the work presented in this thesis. First, the general area of power saving in multimedia streaming was examined. Then power savings in each of the three stages of the multimedia streaming process (reception, decoding and playing) were examined.

Chapter 5

Power Investigation during Multimedia Streaming

This chapter will present one of the main contributions of this thesis, an examination of how multimedia streaming affects the battery of a mobile device. The design of a Total Power Save Algorithm (TPSA), which involves all stages of the multimedia streaming process, is also described. The work presented in this chapter was published in [6] and [7].

5.1 Preliminary Testing

As described previously, the multimedia streaming process can be divided into three stages: reception, decoding and playing. Tests were performed to investigate the effect of each of the three stages on the battery life. For these tests, a 3GHz Pentium 4 desktop computer with 1GB of RAM using the Microsoft Windows XP operating system, 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 aim of these tests was to determine the

areas of the streaming process that influence the battery of the device most significantly. Each stage of the multimedia streaming process was examined individually. This meant that it was possible to see the effect that each of the stages had on the battery, without interference from other stages. Table 5.1 highlights which stages were affected by which tests and shows that some tests influenced more than one stage.

Some measures were taken when doing these tests to compensate for the degradation of the battery. It is well known that batteries degrade over their lifetime. In order to ensure that this did not affect the results too much, each group of tests, i.e. each set of tests that were compared to each other, were performed consecutively, with as little time as possible in between (usually only the time it took to recharge the battery). This means that in some cases, where the same test was used in two different groups, it was performed

Table 5.1: Stages Affected by Each Test: Reception (R), Decoding (D) or Playing (P)

Test	Encoding	Test	R	D	P
Test 1	MP3	Playing		✓	✓
Figure 5.1	MP3	Streaming	✓	✓	✓
Test 2	MPEG-4	Playing		✓	✓
Figure 5.2	MPEG 4	Streaming	✓	✓	✓
Test 3	MP3 128kbps	Playing		✓	✓
Figure 5.3	MP3 192kbps	Streaming		✓	✓
Test 4	MPEG-4 256kbps	Playing		✓	✓
Figure 5.4	MPEG-4 512kbps	Streaming		✓	✓
Test 5	MP3 Sound On	Playing		✓	✓
Figure 5.5	MP3 Sound On	Playing		✓	✓
Test 6 Figure 5.3	MPEG-4 100% Brightness	Playing		✓	✓
	MPEG-4 75% Brightness	Playing		✓	✓
	MPEG-4 50% Brightness	Playing		✓	✓
	MPEG-4 25% Brightness	Playing		✓	✓

twice. Sometimes the results had changed which means that there are some tests that look like they should have identical results but they were different. By doing this, the results presented are as accurate as possible. To further strengthen the results, each group of tests were performed twice to ensure that the results remained relatively the same.

5.1.1 Reception Stage

To investigate the effect of the reception stage on the battery it was necessary to compare situations when media clips were streamed with those in which multimedia clips were played (i.e. clips that were already stored on the device). This way, the only difference between the two situations was that in the first, the reception stage of the multimedia streaming process was used and in the second, it was not. For the tests to be accurate, it was necessary to use the exact same media clip and device settings each time.

In order to test the effect of different content, two clips were used, an MP3 (which, as described previously is MPEG-1 Layer 3) audio clip and an MPEG-4 encoded video clip. For each of the tests, the clips were played in a continuous loop from the time the battery was full until the time the battery had fully depleted.

The first test carried out involved the MP3 audio content. The results, illustrated in Figure 5.1 show that there is a significant difference in battery life between when the audio data is played and when it is streamed. There is a difference of about 5000 seconds, or an increase in battery life of 50% when the media is played instead of streamed.

The second test also compared playing and streaming; however this time an MPEG-4 video clip was used. The difference between the two scenarios can be attributed to the reception stage. The results are presented in Figure 5.2. It can be seen that there is an increase in battery life of approximately 66% when the media is played instead of streamed.

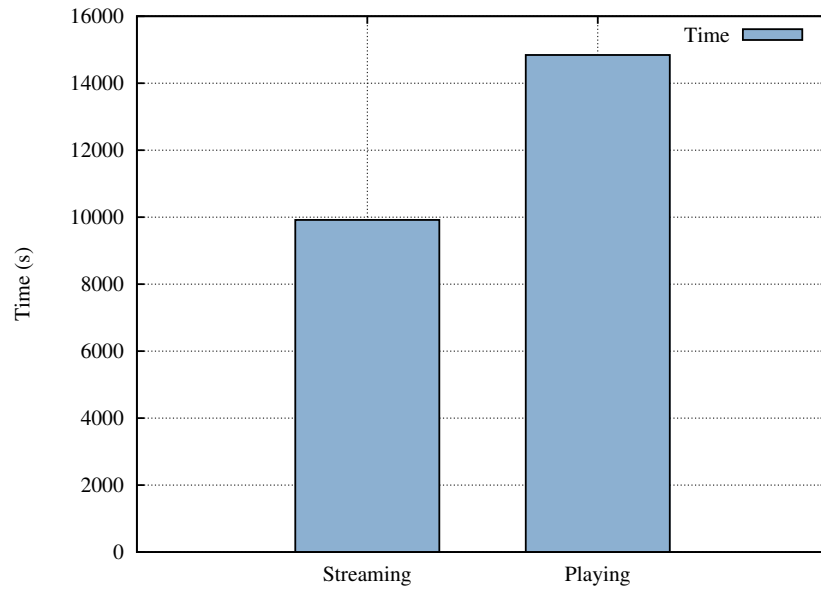


Figure 5.1: MP3: Comparing Playing and Streaming

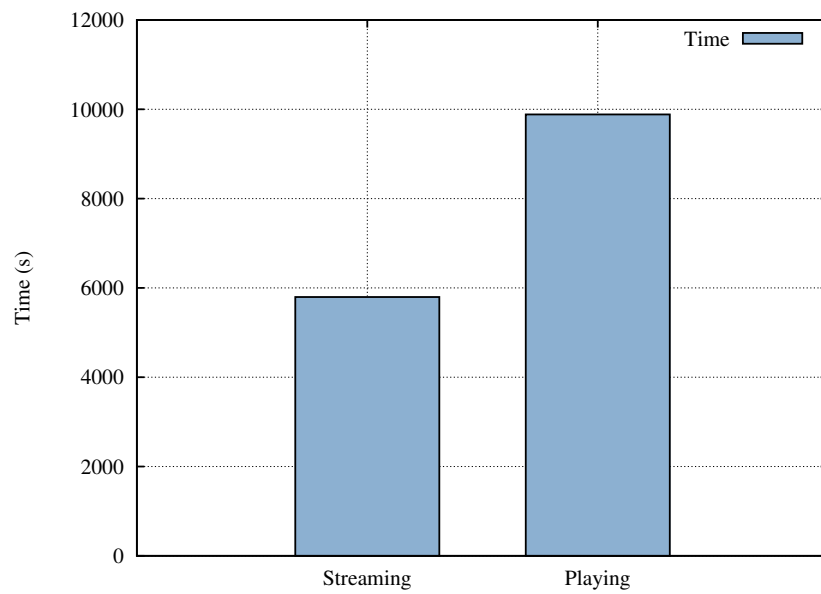


Figure 5.2: MPEG-4: Comparing Playing and Streaming

5.1.2 Decoding Stage

To test the effect that the decoding stage of the streaming process has on the battery of the mobile device, two clips were encoded at different bitrates and then played in continuous loops. Again two sets of tests were performed, one for MP3 audio and one for MPEG-4 video.

In the case of the MP3 audio, the average encoding bitrates were 128kbps and 192kbps. There is about 109% increase in battery life when the media clip with the lower bitrate is used in comparison with the higher bitrate. The results can be seen in Figure 5.3.

For the case of the MPEG-4 encoded video, the situation compared involved clips with average bit rates of 256kbps and 512kbps. The results, shown in Figure 5.4, show an increase of approximately 1000 seconds or about 10% in battery life. These results are not as significant as the results obtained in the audio tests but still show a definite saving in battery life when the multimedia clips average bitrate is reduced.

5.1.3 Playing Stage

The final stage of the multimedia streaming process is the playing stage. To test the effect of this stage on the battery of the mobile device, two aspects of playing were looked at, the speakers and the screen. To test the effect of volume, the same MP3 audio clip was played once with the sound off and once with the sound at an audible level. As can be seen from Figure 5.5, the increase in battery when the sound was muted was dramatic, at approximately 112%, going from 14,483 seconds to 31,595 seconds. A number of other tests were performed with the volume set at different levels in between mute and audible. The results showed that there was no real benefit unless the volume was muted.

To examine the effect of the screen, particularly the luminosity, the same MPEG-4 video clip was played with the client device set to four different

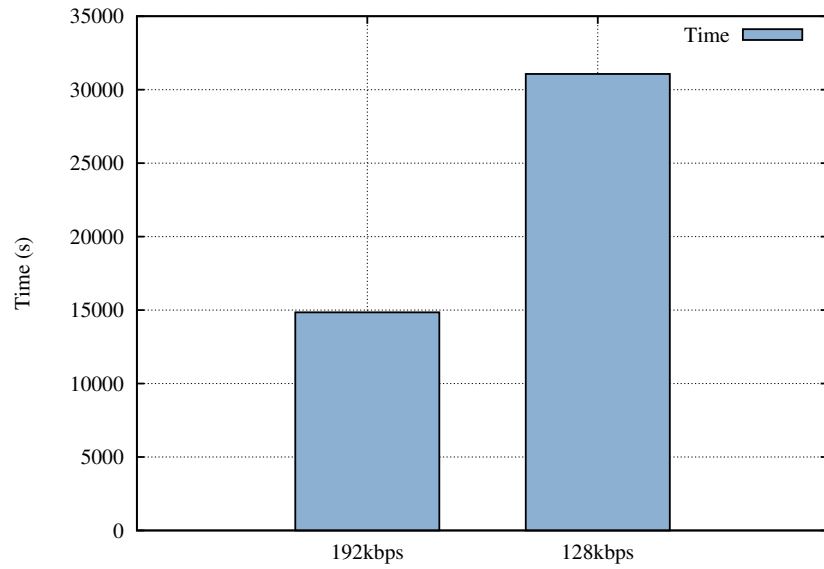


Figure 5.3: MP3: Comparing Multimedia Encoded at Different Bitrates

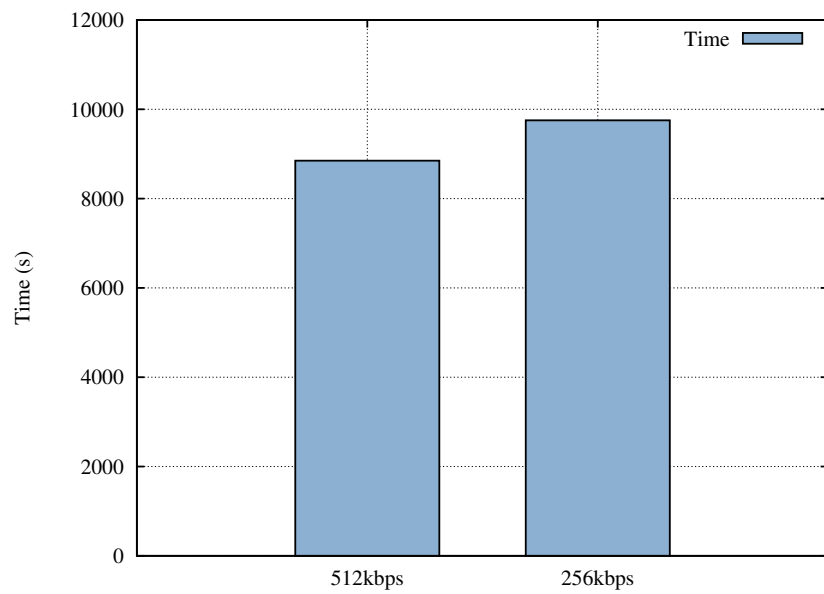


Figure 5.4: MPEG-4: Comparing Multimedia Encoded at Different Bitrates

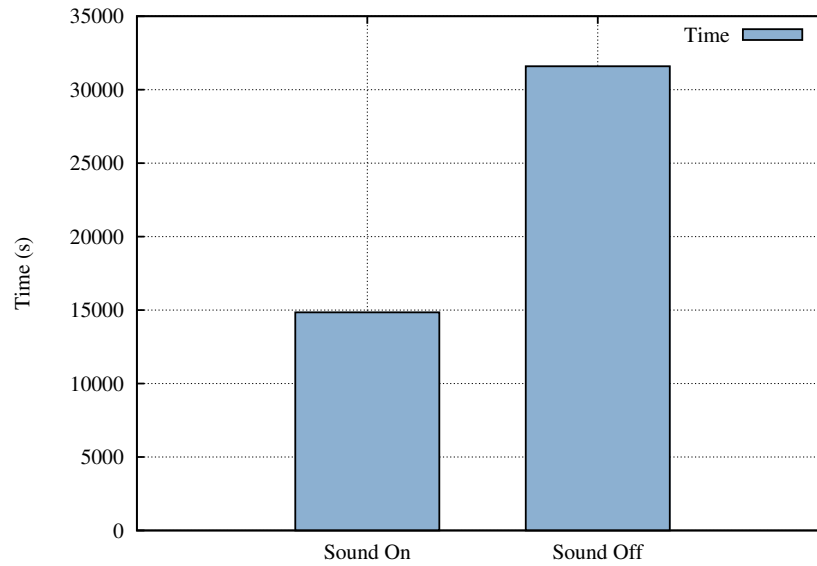


Figure 5.5: MP3: Comparing Sound On and Sound Off

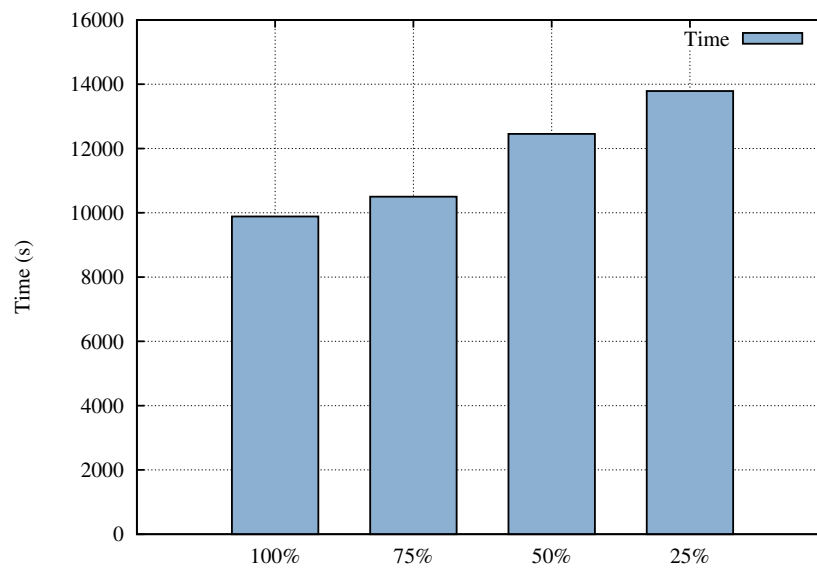


Figure 5.6: MPEG 4: Comparing Different Brightness Levels

screen brightness levels. The results, which are shown in Figure 5.6, demonstrate that lowering the screen brightness does increase the battery life. However, in order to obtain a significant increase in battery it is necessary to decrease the brightness to approximately 25%. Going from 100% brightness to 75%, 50% and 25%, gives an increase in battery of 6%, 26% and 40% respectively. It should be noted that at 25%, the media clip can still be seen and enjoyed at a good quality level.

5.2 Analysis of Results

In discussing these results, we will look at the stages in reverse order, starting with the playing and ending with the reception.

Preliminary testing has shown that savings can be made in the playing stage by reducing the brightness of the screen and the volume of the speakers. Both of these are perfectly feasible but they do present a possible problem. The issue is whether or not the user will want to do either. Although neither makes the watching and/or listening of the media impossible, they will affect the user quality of perception and will be discernible by the user. Therefore, there is a trade-off between the suggested changes and the user quality.

With regards to the decoding stage, the results show that by reducing the bitrate at which the media is encoded, we can make significant savings in battery during the decoding process. This too presents a possible problem. If four different users, with four different devices want to access the same media clip, either four copies of the same clip encoded at four different bitrates are required or a transcoding mechanism is required, which would require extra power and would therefore be counter productive. Although this is a problem that is possible to solve, it may cause other problems when trying to extend the battery lifetime.

In the reception stage, the results of multimedia streaming show that the reception stage has a significant effect on the battery. This means that any

improvement will be beneficial and will save power. The obvious advantage of making any changes in this stage is that the user will not even realise that adjustments are being made as the media clip, and therefore the end user perceived quality, will not be affected.

Therefore, it can be concluded that the reception stage will have the most benefit in terms of increased battery life while also having the least effect on the end user perceived quality.

Therefore, this thesis will propose a scheme to save battery in that stage. To save power in the reception stage, this thesis proposes the **Adaptive Buffer Power Save Mechanism (AB-PSM)**. This improves the power save mechanism proposed in the IEEE 802.11 standard, offering significant savings in battery life without affecting the user perceived quality. It is designed primarily for mobile multimedia streaming using an 802.11 WLAN. Although in this thesis, 802.11b is used, AB-PSM is easily extendable to other members of the 802.11 family such as 802.11g, 802.11n and 802.11e.

5.3 Total Power Save Algorithm

It should be noted that savings can also be achieved in the other two stages and therefore they should not be completely disregarded. It would be possible to incorporate savings for all three stages into a Total Power Save Algorithm (TPSA), the design of which will now be briefly described.

5.3.1 Overview

In order to design a TPSA, solutions need to be proposed for all three stages. For the reception stage, AB-PSM would be used. This is described in greater detail in the next chapter.

The decoding stage is related to the server encoding the media to be sent and the client decoding the received media. As the algorithm is focusing on the battery life of the mobile device, it will be more beneficial to focus on

decoding, as this is carried out by the mobile device and will directly impact the battery. However, the encoding scheme used by the server will influence the power required for the decoding. For example, if a higher bitrate is used, more battery will be required to decode it. Therefore, to save power in the decoding stage, it is necessary to use media that has been encoded at a lower bitrate, as shown in Figure 5.3 and Figure 5.4. However, a lower bitrate could result in a lower user perceived quality. Therefore, a trade off between quality and power is necessary. To make the algorithm as user friendly as possible, the decision on which of these is more important could be given to the user.

The playing stage of the multimedia streaming process involves the media being played on the client device. In Chapter 2, it was stated that the playing stage can be attributed to two principal components: the speakers and the screen. Therefore, it is obvious that in order to save power in this stage, it is necessary to focus on them. It was shown in Figure 5.5 that muting the volume gives significant savings. To minimise the power used by the screen of the mobile device, the brightness can be reduced. Figure 5.6 shows that a significant decrease in brightness is required for a significant decrease in battery lifetime but even a slight increase in battery lifetime could be considered beneficial. Once more, to increase the user friendliness of the algorithm, the choice of brightness and/or volume levels could be given to the user.

The architecture for the TPSA would be a client server architecture, similar to that described in Chapter 2. A simple block diagram of the algorithm, which is shown in Figure 5.7 will now be described. On a very basic level, the algorithm would consist of the following parts:

- Server side scalable media storage block: this is where the multimedia content would be stored. It would also have the ability to scale the media if required, for example, by encoding it at a lower bitrate.
- Server side AB-PSM block: this is where the AB-PSM would be im-

plemented. It would control how and when the data is sent.

- Server side application buffer block: this is where the media content would be stored for the AB-PSM. This is the buffer that "hides" the content from the access point.
- Server side decoding stage adaptation block: this is where the clips may be encoded at lower bitrates and where the power save would notify if lower encoding rates are required.
- Client side receive, decode and display block: this is where the media content is received, decoded and displayed. It is in effect, the client device.
- Client side battery information block: this is where the information relating to the remaining battery power in the client device is obtained.
- Client side scaling power save block: this is where the client side adaptation would be performed, such as adjusting the screen brightness and the volume.
- Client side adaptation for playing stage block: this is where the volume and brightness levels would be reduced, if the scaling power save block says it is required

The TPSA could be adaptive by implementing the different power saving techniques in an incremental fashion. This would require a decision algorithm. Within the three stages, three levels of adjustment could be set. For example, in the reception stage, three different levels of AB-PSM could be given. In the decoding stage, three different bitrates would be offered and in the playing stage, three different levels of either volume and brightness or a combination of the two.

Testing of user preferences would decide which of the stages to implement first and to which level to implement them to. Users may prefer to

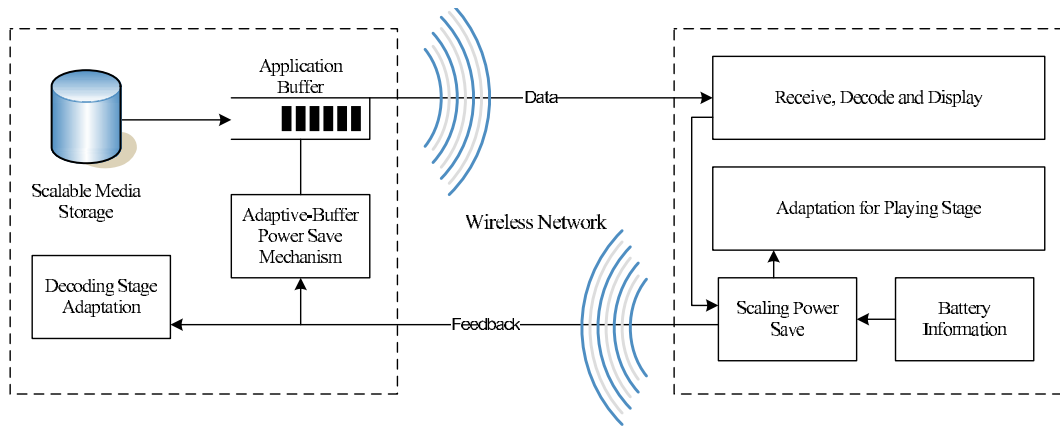


Figure 5.7: Total Power Save Algorithm

implement the first level in every stage followed by the second, etc. Or it may be more beneficial to implement all levels of the reception stage first and then alternate between the other two. Another option would be that the playing stage power save mechanisms are never implemented, or only when the battery reaches a certain level. Users may also be given the option to override the default algorithm and choose for themselves.

A sample decision algorithm is shown in Figure 5.8. The algorithm assumes that, throughout the battery lifetime, the system will always be in some state. Each of the stages has three different power save states associated with it, as shown in Table 5.2. Different indexes are used for each stage as follows: the reception stage uses i , the decoding stage uses j and the playing stage uses k . Therefore, the current power save state of the device can always be represented by the triplet $\{i, j, k\}$, where i, j and k will have a value of 0, 1 or 2 depending on the current state. The different states associated with each stage are shown in Table 5.2. The state diagram of the sample decision algorithm is shown in Figure 5.8. The user could also be given the option to override the decision algorithm and implement their own preferences.

In the decision algorithm, a comparison is made between the server and

Table 5.2: Scaling of Power Save Mechanisms

Stage	Power Save Mechanism	State
Reception	Legacy Power Save	0
	AB-PSM Level 1	1
	AB-PSM Level 2	2
Decoding	Max Bit Rate	0
	Average Bitrate	1
	Low Bitrate	2
Playing	Max Brightness/Volume	0
	Average Brightness/Volume	1
	Low Brightness/Volume	2

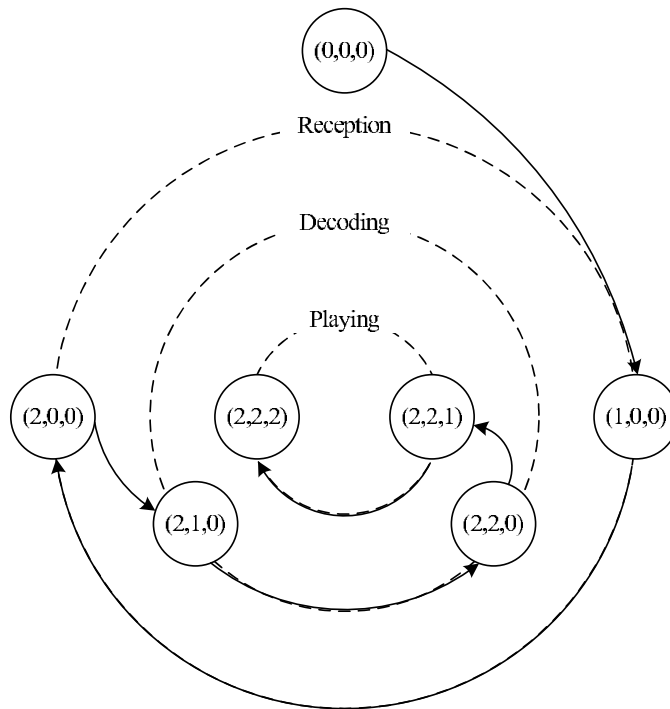


Figure 5.8: TPSA: State Diagram of Decision Algorithm

the client of battery remaining and battery required to complete the task. This comparison could be carried out as follows. The server would know the time required in order to complete any multimedia clip it has stored. The client would be able to read its remaining battery life using the client side battery information block. The client could periodically send the remaining battery life to the server, using the feedback loop. The server would take this time remaining and compare it to the time required to complete the task. If time remaining is greater than time required, then the server can tell the client that, for the moment, no further power saving mechanisms are required, again using the feedback loop. If the time remaining is less than the time required, the server will decide which power save mechanism will be implemented. This comparison would be carried out continuously throughout the streaming task. The reason for this is that circumstances may change, for example, if the user device was a mobile phone and a call was received, then the battery would be further drained.

This section has only proposed the idea of the total power save algorithm. The AB-PSM scheme proposed in this thesis provides a strong base for it, while the preliminary testing described earlier in this chapter indicates where the next logical steps would be. However, significant further testing is required. It is essential to investigate whether the power saving mechanisms proposed for each stage have any effect on the other power saving mechanisms. If they do, it may be only viable to implement some of them.

A default decision algorithm would also need to be defined. This would be used if the user chooses not to set their options. This default decision algorithm would be based on the power saving mechanism which has the least impact on the user being implemented first. To find the effect of each of the power save mechanisms on the user, it would be necessary to perform extensive subjective testing, involving a large number of users and devices and a variety of situations and multimedia tasks.

Finally the algorithm would need to be implemented and tested. This

presents some problems, particularly in terms of the device. Many operating systems do not allow automatic setting of parameters such as brightness and volume and in a lot of cases, they can only be adjusted manually on the device. This would not be satisfactory for the algorithm described above as these parameters would be required to be adjusted in the background, without the users knowledge. It may be possible to simulate the algorithm but it could be argued that simulations have limited accuracy. This is particularly relevant when referring to battery life, as sometimes batteries can be unpredictable and it is therefore, difficult to simulate an accurate representation of the life of a battery.

If these difficulties are surpassed, this algorithm could be very successful. In the next two chapters, it will be shown that the effect of AB-PSM on extending battery life is significant and if this was combined with the other power saving mechanisms, the consequent battery savings could be even more considerable.

5.4 Chapter Summary

This chapter presented a detailed study of how multimedia streaming affects the battery of a mobile device. By dividing the multimedia streaming process into three distinct stages, it is possible to examine in detail the effect that each stage has on the battery. The results from the tests were used to conclude that the most savings could be achieved in the reception stage and it was because of this that the power save mechanism, AB-PSM, proposed in this thesis was created. An alternative Total Power Save algorithm was also briefly described, which incorporates power savings for all three stages.

Chapter 6

Adaptive Buffer Power Save Mechanism

This chapter presents the novel Adaptive-Buffer Power Save Mechanism, which is the power save mechanism proposed in this thesis to extend battery life in a mobile device performing multimedia streaming. This is a reception stage power save mechanism which while saving the battery, does not affect user perceived quality. The chapter describes the algorithm and its benefits over other schemes. The work presented in this chapter was published in [8].

6.1 Proposed Solution

This thesis proposes a novel power save scheme, known as Adaptive-Buffer Power Save Mechanism (AB-PSM), which improves battery life while maintaining user quality. This scheme compliments the legacy PSM that is described in the IEEE 802.11 standard, working with it as opposed to replacing it. An important advantage of AB-PSM is that it can be implemented into current networked applications without requiring adjustment to the standard.

The algorithm is implemented on a desktop computer, acting as the

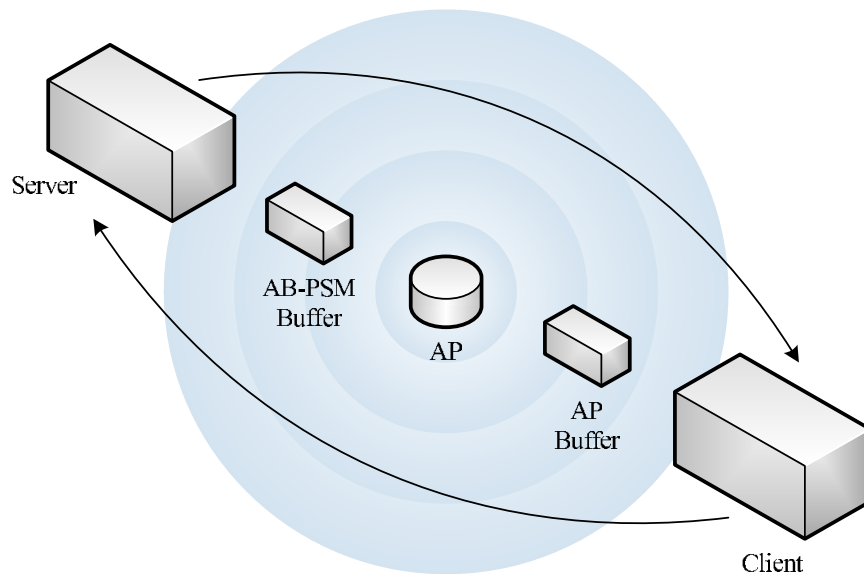


Figure 6.1: AB-PSM: Block Diagram

server, which sends the data to an 802.11b AP where it is forwarded on to the mobile client device.

A block level diagram is shown in Figure 6.1. The Application Buffer, which is hidden from the AP buffer, is completely invisible to the user. The device can request data from the AP or receive it. When it is received, it is generally from the AP buffer. In between the data going from the server to the AP buffer, it will be stored in the Application Buffer for a specified period. This is when the NIC can sleep and power can be saved.

6.2 Adaptive-Buffer Power Save Mechanism

The proposed AB-PSM works in conjunction with the legacy power save mechanism, as described in the IEEE 802.11 standard. As previously stated, AB-PSM makes no change to the IEEE 802.11 standard. Therefore, the beacons are sent as normal by the AP and the STAs listen to the beacons to check if the AP has any buffered traffic for it. The AB-PSM principle

relies on the fact that, in order to save power, the device will be allowed to enter the low-power sleep state when it is idle. However, the device will only enter this mode once it has received all waiting traffic and there is none buffered for it at the AP. In order for the device to sleep more than it usually would, it is necessary to hide the traffic from the AP, which in turn informs the STA that there is no traffic for it. To achieve significant power savings, this must be done at least every second beacon interval. In normal 802.11 operation, the traffic would be sent so frequently the device would not enter the low power sleep mode often enough to achieve significant power savings. Therefore, the way that the traffic is sent must be changed to allow the AP to inform the device that there is no traffic and therefore, allow it to sleep for longer periods. In this way, significant power savings can be achieved.

AB-PSM does this by introducing a second buffer called the Application Buffer. This is in addition to the Access Point Buffer and it is implemented on the server. The Application Buffer effectively hides packets from the default Access Point Buffer. This means that 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. Therefore, if this happens often enough, the device will sleep for longer periods.

Assuming that the ListenInterval is set to one, as is generally the case, the station will wake up at every beacon interval to listen to the beacon. If the TIM within the beacon indicates that there is buffered traffic at the AP, the station will enter the fully awake state in order to receive it, otherwise it will return to the low power sleep mode. The Application Buffer stores 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

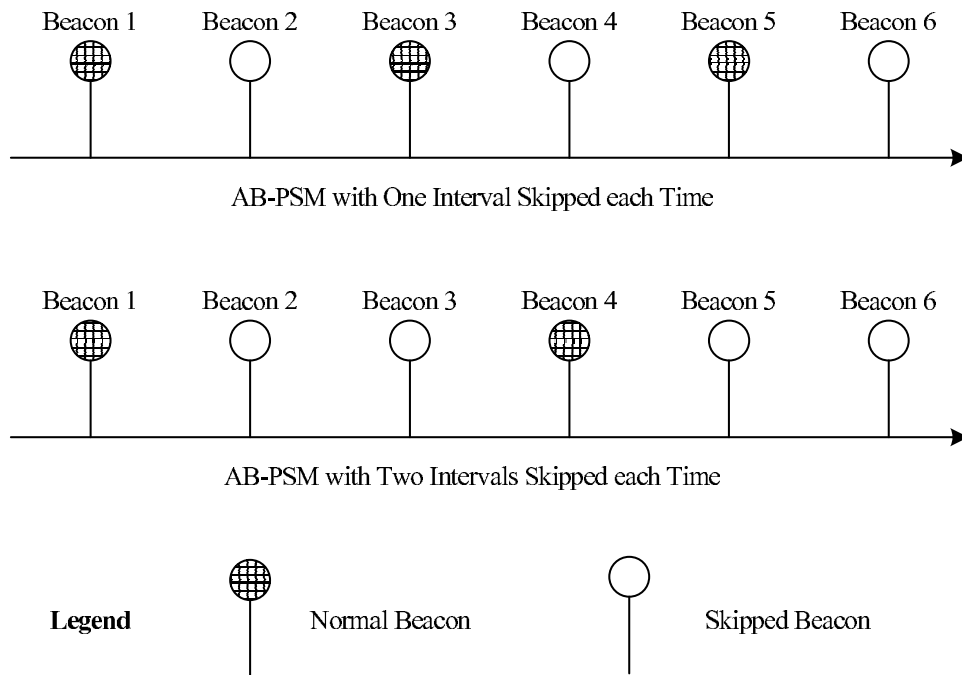


Figure 6.2: AB-PSM Examples with One and Two Skipped Beacon Intervals

can be varied. The packets could be held so as to skip one, two or more beacon intervals, saving power and enabling the battery to last longer. However, the longer the packets are held in the Application Buffer, the higher the probability of delay. For non-real time data applications, this may be acceptable but for real-time streaming, it will not be tolerated. 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. For example, two, three or more beacon intervals can be skipped and this number will directly relate to the threshold time. Figure 6.2 graphically presents two examples with one and two skipped beacon intervals.

One of the main benefits of the newly proposed 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 saves power by allowing the

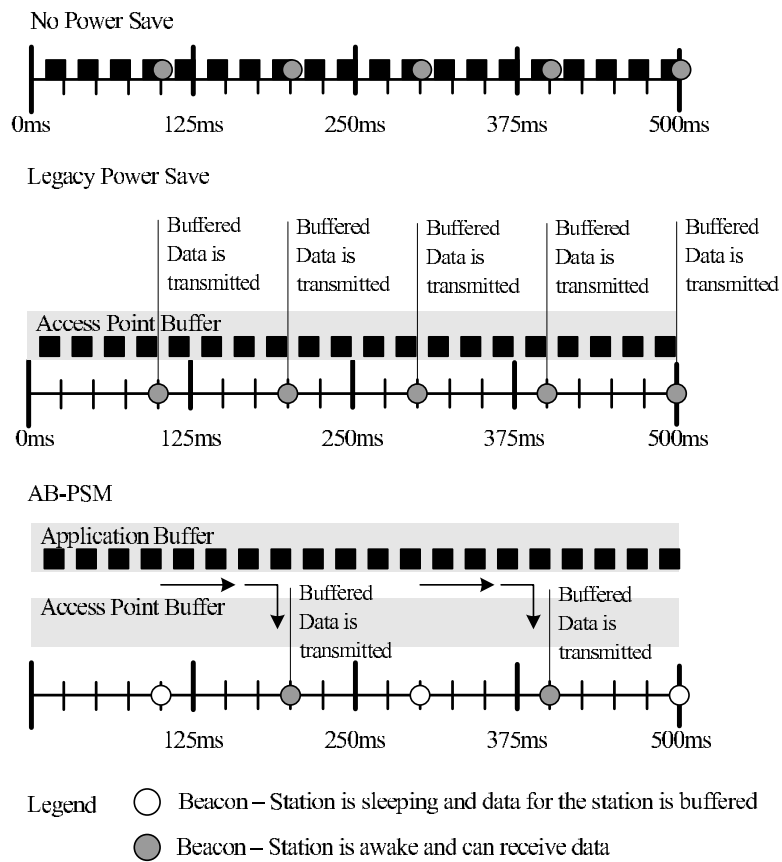


Figure 6.3: Graphical Comparison IEEE 802.11 Transmission with: No Power Save, Legacy Power Save and AB-PSM

station to sleep for a longer period 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.

The scheme is described graphically in Figure 6.3. 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 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.

6.3 Scalability Issues and Solutions

Some scalability issues occur with AB-PSM when more than one client is included. For example, assuming that two clients synchronise, i.e. they sleep through the same beacon intervals and wake to receive traffic at the same time, this will cause inefficient and unfair usage of AB-PSM. Due to the fact that both clients will not be able to receive traffic synchronously (they cannot use the medium at the same time, as specified in the 802.11 standard [5]), one client must stay awake for twice as long. This is because both clients wake at the beacon interval and see that there is buffered traffic waiting for them. Therefore both clients will stay awake to receive this data. However, only one client will be able to receive traffic at a time, which means that the second client will have to remain awake while the first client receives its data and then while it receives its own data. This is counter-productive because the client that must wait is using battery power when it is not doing anything and could be asleep, saving power.

This is shown in detail in Figure 6.4. It can be seen that the clients synchronise at the beginning and because of this, Client 2 has to remain awake until Client 1 finishes receiving its data. This is completely inefficient.

To avoid this, it is necessary to ensure that the clients do not synchronise. To do this, instead of just specifying the interval (e.g. telling the client to wake at every second interval), the starting point will also be specified. This means that clients will sleep and wake at different points and in this way, issues of synchronisation can be avoided.

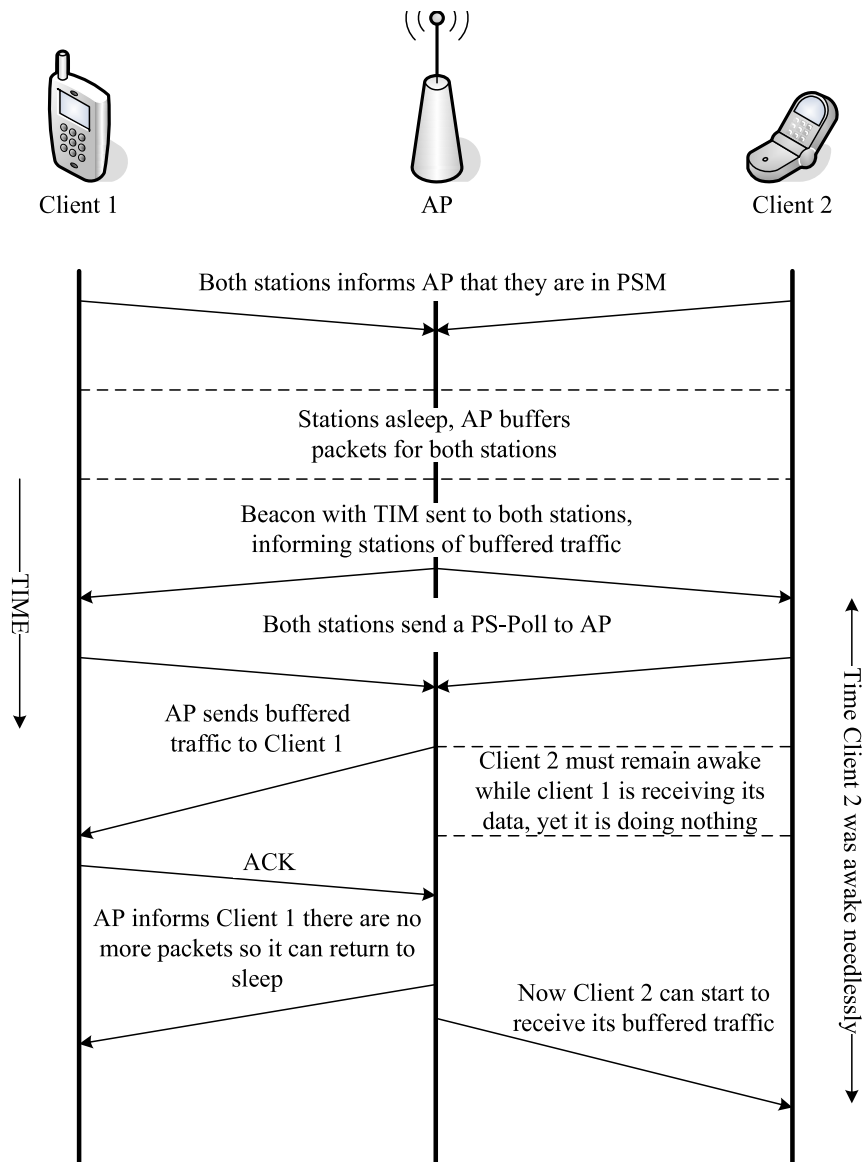


Figure 6.4: Scalability Issues of AB-PSM

6.4 Advantages of AB-PSM

AB-PSM has many benefits over other schemes. In terms of the PSM specified in the standard, AB-PSM offers better savings in a more efficient manner.

It does this without changing the specifications of the standard. The beacons are taken advantage of in AB-PSM. This is in contrast to other proposed schemes, as discussed in Chapter 3, which choose to ignore the beacons even though they are a fundamental part of the 802.11 standard. This means that any of these proposed schemes that do not work in conjunction with the 802.11 standard may either not work with 802.11 equipment or will not take full advantage of the 802.11 features, making them difficult to implement. AB-PSM will work without any changes to hardware required, making it extremely user friendly and implementable.

Another advantage of AB-PSM is that the user will not know that anything is happening. AB-PSM works in the background and ensures that there is no significant delay. This means that user quality is not affected and the only obvious difference from the user's point of view is that the battery of their device will last longer. It is known that the quality is unaffected because the data rate remains the same and no delays are experienced. The data is always in the buffer when it needs to be.

Another advantage of AB-PSM is its adaptability. Based on either the remaining battery life or the user's preference, the level of AB-PSM being used can be adjusted. This could be performed adaptively by implementing a feedback loop from the device to the server. The feedback could contain the remaining battery life as well as user preference. An example of user preference is that the user may choose not to implement the scheme because they do not need to save battery. AB-PSM will automatically be disabled if the device is plugged in. This information would be sent in the feedback loop. If the battery becomes critically low and the multimedia task is almost complete, AB-PSM could temporarily bypass the threshold in order to complete the task. As this would likely introduce a user detectable delay, this would be a decision that would have to be approved by the user.

6.5 Summary of Chapter

This chapter presented the novel power scheme proposed in this thesis. The scheme, called Adaptive-Buffer Power Save Mechanism, works in conjunction with and improves upon the legacy power save mechanism that is defined in the IEEE 802.11 standard. The operation of the scheme is described in detail and its benefits over other schemes are highlighted. Some limitations are also identified and discussed.

Chapter 7

Experimental Tests

This chapter presents the experimental tests that were performed in relation to AB-PSM. A detailed description of the test setup is given as well as an in-depth analysis of the results that were obtained. The presented results will prove the effectiveness of the proposed AB-PSM.

7.1 Test Setup and Scenarios

In order to assess the effectiveness of the newly proposed AB-PSM, tests were performed to compare it to the case when streaming is performed over IEEE 802.11 with no PSM employed and with the legacy PSM, respectively. For these tests, a 3GHz Pentium 4 desktop computer with 1GB of RAM using the Microsoft Windows XP operating system, 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 testbed is shown in Figure 7.1.

The tests involved continuously sending multimedia packets to the client. Two parameters were adjusted for each test. The first was the interval between packets being sent. The second parameter was the packet size. The

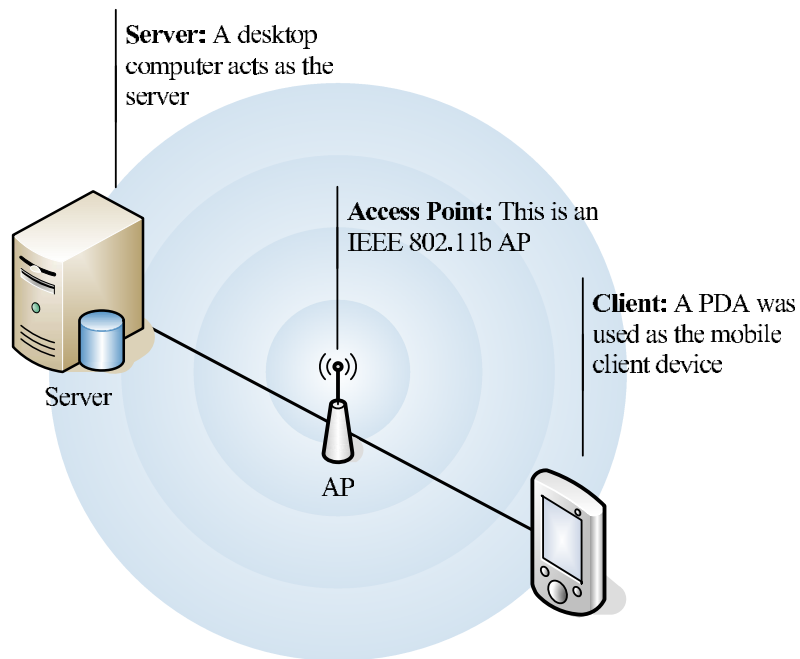


Figure 7.1: Test Setup

interval and packet size were increased by the same factor, based on the scheme being used.

With no PSM and the legacy PSM, the data is just sent. It makes no difference what the packet size or interval are. With no PSM, the device never sleeps. With the legacy PSM, if the device is idle, it will enter the low power sleep mode. However, as the main idea of AB-PSM is to send a larger amount of data less frequently, the tests involved 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.

7.2 Preliminary Tests

The goal of the preliminary tests was to see if AB-PSM had any effect on applications with a smaller bandwidth than that required for multimedia

streaming. The testbed as shown in Figure 7.1, was used with all the same settings described in Section 7.1. The first set of tests used smaller packets than would be used for video streaming. These were performed in order to show the effects of AB-PSM on general network tasks, as opposed to multimedia streaming. 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(Bytes) to 4096B. The first test was 802.11 with the PSM disabled, with a 25ms interval and a packet size of 512B. The second test was the same settings as the first but this time, the legacy PSM was enabled. The third test uses AB-PSM with an inter-packet sending interval of 100ms and a packet size of 2048B and the final test, also using AB-PSM, had an interval of 200ms and a packet size of 4096B. As all of the intervals were increased by the same factor as the packet size, it meant that the same amount of data was sent in every case.

7.2.1 Analysis of Results

The preliminary results are presented in Figure 7.2. The graph shows four bars, each representing one of the tests described above. The first bar shows the results of the first test, which was 802.11 with the PSM switched off, the next bar shows the legacy 802.11 PSM with the same settings. The two AB-PSM tests are represented by the third and fourth bars. The y-axis in the graphs represents the time in minutes that the battery of the device lasted.

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 are employed respectively. There is an increase of 140% in the battery life when the 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.

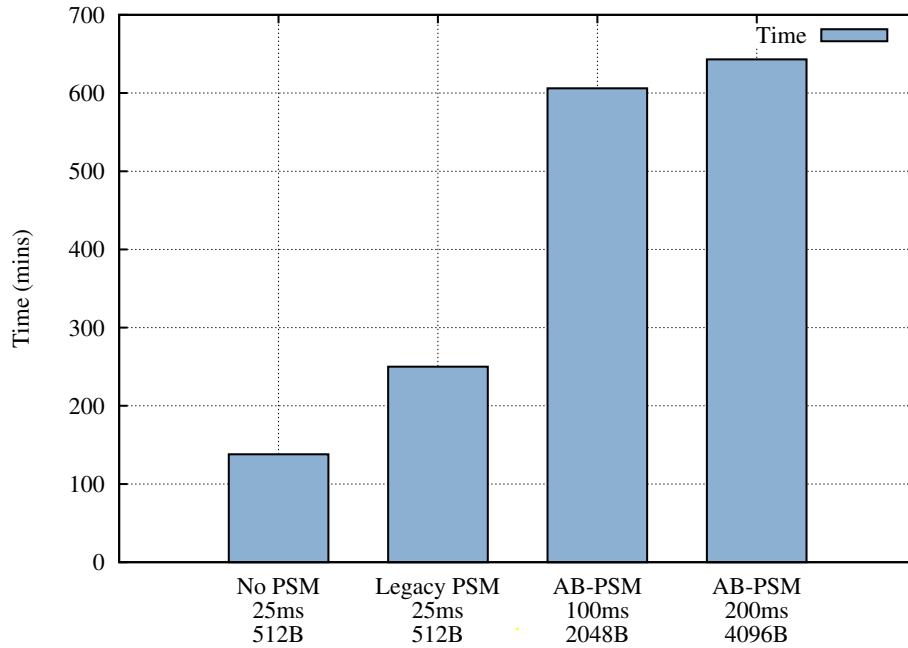


Figure 7.2: Results of AB-PSM Tests with Non-Video Packets

7.2.2 Investigation into the Effect of Inter Packet Sending Interval

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 previous tests, with the results shown in Figure 7.2, in all cases the average sending data rate was the same. However, in the following tests, the amount of data being sent varies. These mainly aim to investigate the effect of the packet size and inter-packet sending interval have on battery power level and do not represent the correct use of AB-PSM (because the packet size and interval are not being increased at the same rate).

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 7.3, 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.

7.2.3 Investigation into the Effect of Packet Size

This section investigates the effect that the packet size has on the battery life. For these tests, AB-PSM was used. The base values were an interval of 100ms and a packet size of 2048B.

7.3 Investigation into Effectiveness of AB-PSM in Multimedia Streaming

This section aims at examining how effective AB-PSM is in multimedia streaming. Multimedia streaming applications require a higher bandwidth and the packets are much larger than in other applications.

To test the effectiveness of AB-PSM, it was necessary to increase the packet size and inter-packet sending interval, while ensuring the data rate was kept constant. To determine the packet size, it was necessary to take into account that Ethernet limits the packet size to 1500 Bytes [66] and when headers are taken into account, this reduces the payload to a maximum of 1460 Bytes. Anything that exceeds the maximum packet size of 1460 Bytes will be fragmented. In the AB-PSM tests, this fragmentation was controlled by specifying the number of 1460B packets that would be sent.

If for example, a test involved streaming a video with a bitrate of 512 kbps, the following calculations were performed in order to calculate the in-

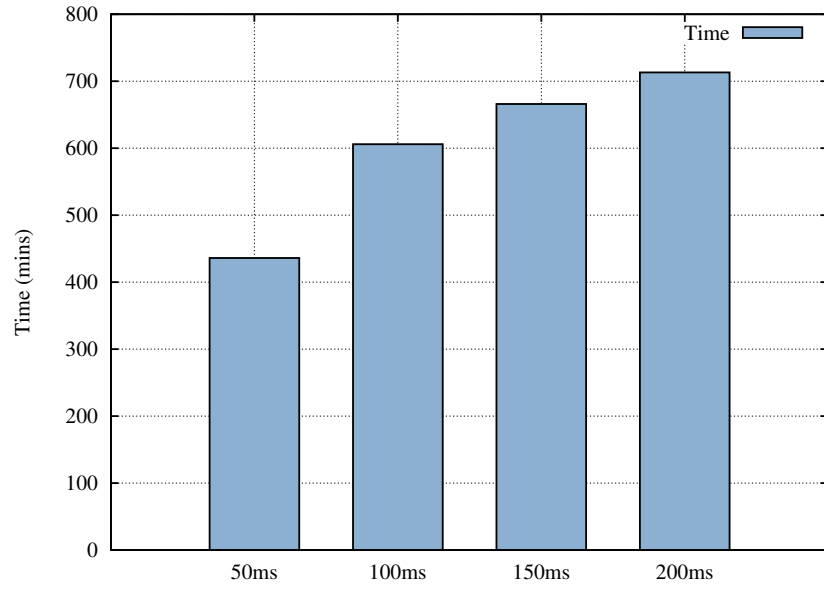


Figure 7.3: Constant Packet Size, Varied Interval

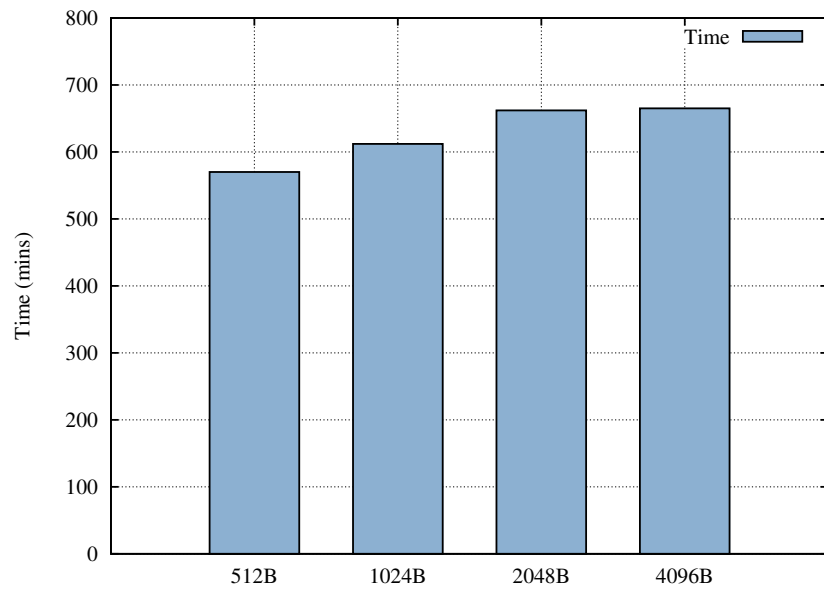


Figure 7.4: Varied Packet Size, Constant Interval

terval between packets.

$$\text{Bitrate} = 512\text{kbps} \approx 512000\text{bps}$$

$$\text{PacketSize} = 1460 \text{ Bytes} = (1460 \times 8) = 11680 \text{ bits}$$

$$\text{PacketRate} = \frac{512000}{11680} \text{ packets per second} = 43.83 \approx 44 \text{ packets per second}$$

$$\text{Interval} = \frac{1}{44}\text{ms} = 22.72 \times 10^{-3} = 23\text{ms}$$

Therefore, based on the above equation, in order to stream a video without using AB-PSM, it is necessary to send one 1460B packet every 23ms. When AB-PSM is used, the interval and packet size are increased by the same factor, the value of which depends on the level of AB-PSM being used.

In order to see exactly what happened during the transmission of the data, the WildPackets AiroPeek ¹ network sniffer was used. This allowed a detailed examination of the packets being sent and showed clearly exactly how the scheme worked.

The first test that was performed was without any power save mechanisms implemented. Consequently, the NIC never entered the low power sleep mode. The details of the packets that were sent are shown in Figure 7.5. In this case, it can be seen that there is activity between all beacons. This setup would not have allowed the Network Interface Card (NIC) to enter sleep mode at any time. In this case, the only packets sent are the data, the acknowledgements and the beacons.

The next test involved no AB-PSM, however this time the legacy PSM was enabled, the packets printout can be seen in Figure 7.6. As can be seen, there is activity between every beacon, with UDP datagrams, acknowledgements and other 802.11 packets being sent. Consequently, the WNIC is not sleeping very much and even though the legacy power save is enabled, due to the frequency of packets, there are no significant power savings. In this scenario, the addition of the PS-Poll packets, which were described Chapter 3, can

¹www.wildpackets.com

Packet	Data Rate	Size	Relative Time	Protocol
9	1.0	71	00.093760	802.11 Beacon
10	11.0	1524	00.121650	UDP
11	11.0	14	00.121798	802.11 Ack
12	11.0	1524	00.153602	UDP
13	11.0	14	00.153746	802.11 Ack
14	1.0	88	00.157470	802.11 Beacon
15	11.0	1524	00.184467	UDP
16	11.0	14	00.184609	802.11 Ack
17	11.0	1524	00.215794	UDP
18	11.0	14	00.215940	802.11 Ack
19	11.0	1524	00.246988	UDP
20	11.0	14	00.247134	802.11 Ack
21	1.0	88	00.259810	802.11 Beacon
22	11.0	1524	00.278346	UDP
23	11.0	14	00.278487	802.11 Ack
24	1.0	71	00.298853	802.11 Beacon
25	11.0	1524	00.309498	UDP
26	11.0	14	00.309639	802.11 Ack
27	11.0	1524	00.340752	UDP
28	11.0	14	00.340893	802.11 Ack
29	1.0	88	00.362206	802.11 Beacon
30	11.0	1524	00.372004	UDP
31	11.0	14	00.372143	802.11 Ack
32	1.0	71	00.401024	802.11 Beacon
33	11.0	1524	00.404473	UDP
34	11.0	14	00.404612	802.11 Ack
35	11.0	1524	00.434519	UDP
36	11.0	14	00.434658	802.11 Ack
37	1.0	88	00.464618	802.11 Beacon
38	11.0	1524	00.467276	UDP
39	11.0	14	00.467416	802.11 Ack
40	11.0	1524	00.497115	UDP

Figure 7.5: Analysis of Packets, No AB-PSM, Legacy 802.11 PSM disabled

Packet	Data Rate	Size	Relative Time	Protocol
14	1.0	88	00.102604	802.11 Beacon
15	2.0	20	00.102746	802.11 PS-Poll
16	2.0	14	00.102992	802.11 Ack
17	11.0	1524	00.105820	UDP
18	11.0	14	00.105959	802.11 Ack
19	2.0	20	00.106097	802.11 PS-Poll
20	2.0	14	00.106241	802.11 Ack
21	11.0	1524	00.108289	UDP
22	11.0	14	00.108454	802.11 Ack
23	2.0	20	00.108592	802.11 PS-Poll
24	2.0	14	00.108729	802.11 Ack
25	11.0	1524	00.110375	UDP
26	11.0	14	00.110514	802.11 Ack
27	2.0	20	00.110650	802.11 PS-Poll
28	2.0	14	00.110788	802.11 Ack
29	11.0	1524	00.112426	UDP
30	11.0	14	00.112564	802.11 Ack
31	1.0	88	00.204999	802.11 Beacon
32	2.0	20	00.205139	802.11 PS-Poll
33	2.0	14	00.205517	802.11 Ack
34	11.0	1524	00.208253	UDP
35	11.0	14	00.208392	802.11 Ack
36	2.0	20	00.208531	802.11 PS-Poll
37	2.0	14	00.208668	802.11 Ack
38	11.0	1524	00.210312	UDP
39	11.0	14	00.210451	802.11 Ack
40	2.0	20	00.210588	802.11 PS-Poll
41	2.0	14	00.210733	802.11 Ack
42	11.0	1524	00.212418	UDP
43	11.0	14	00.212582	802.11 Ack
44	1.0	88	00.307424	802.11 Beacon
45	2.0	20	00.307564	802.11 PS-Poll
46	2.0	14	00.307710	802.11 Ack
47	11.0	1524	00.310653	UDP
48	11.0	14	00.310792	802.11 Ack
49	2.0	20	00.310929	802.11 PS-Poll
50	2.0	14	00.311067	802.11 Ack

Figure 7.6: Analysis of Packets, No AB-PSM, Legacy 802.11 PSM enabled

be seen. There is also a lot more activity between the beacons in this case. This is because each time the client receives data, it will check with the AP whether there is more data to receive and if there is not, it will enter the low power sleep state. With the PSM enabled, the device receives the data more efficiently due to the PS-Poll asking the AP is there more waiting and if there is, send it.

If AB-PSM is enabled on the same media clip with the data rate kept the same, then the following parameters are necessary. In order to allow the device to sleep, it is necessary to "hide" the waiting packets from the device for one beacon interval. This means that the WNIC can sleep through that beacon interval and power will be saved. As the beacon interval is set to the default value of 100ms, in order to skip every second one, it is necessary to set the interval between packets to 200ms. This increases the original interval (of 23ms) by approximately eight times which means that it is necessary to send a packet that is eight times as big. By sending eight 1460B packets as opposed to one 11,680B (eight times as big) packet, the fragmentation is controlled and the data is sent in the most efficient manner possible.

In figure 7.7, an analysis of the packets sent during four beacon intervals with the first level of AB-PSM enabled, is shown. Between the first and the second beacon, nothing is sent. This is because the waiting packets are hidden from the AP and the NIC of the mobile device can sleep. The device does wake and listens to the beacon to see if there is buffered traffic at the AP for it, however once it sees there is none, it returns to the low power sleep state. Once the first beacon has been sent and the WNIC has returned to its low power sleep state, then the packets are transferred to the AP buffer. When the second beacon is received, the device sees that there is buffered traffic at the AP for it. It sends a PS-Poll to the AP to let it know that it is awake and can receive the traffic. The AP acknowledges this and sends the UDP data. This will be sent in eight parts, shown as Internet Protocol (IP) fragments in the diagram, with each part being acknowledged by the client.

Packet	Data Rate	Size	Relative Time	Protocol
34	1.0	88	00.102459	802.11 Beacon
35	1.0	88	00.204886	802.11 Beacon
36	2.0	20	00.205148	802.11 PS-Poll
37	2.0	14	00.205392	802.11 Ack
38	11.0	1536	00.208538	UDP
39	11.0	14	00.208677	802.11 Ack
40	2.0	20	00.208815	802.11 PS-Poll
41	2.0	14	00.208952	802.11 Ack
42	11.0	14	00.209108	802.11 Ack
43	2.0	20	00.209628	802.11 PS-Poll
44	2.0	14	00.209766	802.11 Ack
45	11.0	1536	00.212873	IP Fragment
46	11.0	14	00.213011	802.11 Ack
47	2.0	20	00.213149	802.11 PS-Poll
48	2.0	14	00.213287	802.11 Ack
49	11.0	1536	00.214932	IP Fragment
50	11.0	14	00.215070	802.11 Ack
51	2.0	20	00.215208	802.11 PS-Poll
52	2.0	14	00.215353	802.11 Ack
53	11.0	1536	00.217001	IP Fragment
54	11.0	14	00.217140	802.11 Ack
55	2.0	20	00.217276	802.11 PS-Poll
56	2.0	14	00.217415	802.11 Ack
57	11.0	1536	00.219072	IP Fragment
58	11.0	14	00.219219	802.11 Ack
59	2.0	20	00.219357	802.11 PS-Poll
60	2.0	14	00.219495	802.11 Ack
61	11.0	14	00.220007	802.11 Ack
62	2.0	20	00.220417	802.11 PS-Poll
63	2.0	14	00.220556	802.11 Ack
64	11.0	1384	00.223332	IP Fragment
65	11.0	14	00.223471	802.11 Ack
66	1.0	88	00.307411	802.11 Beacon
67	1.0	88	00.409808	802.11 Beacon
68	2.0	20	00.409947	802.11 PS-Poll
69	2.0	14	00.410328	802.11 Ack
70	11.0	1536	00.412950	UDP

Figure 7.7: Analysis of Packets, AB-PSM with One Skipped Beacon, Legacy 802.11 PSM enabled

Chapter 7: Experimental Tests

Packet	Data Rate	Size	Relative Time	Protocol
50	1.0	88	00.204922	802.11 Beacon
51	1.0	88	00.307363	802.11 Beacon
52	1.0	88	00.409721	802.11 Beacon
53	2.0	20	00.409862	802.11 PS-Poll
54	2.0	14	00.410037	802.11 Ack
55	11.0	1536	00.413150	UDP
56	11.0	14	00.413289	802.11 Ack
57	2.0	20	00.413428	802.11 PS-Poll
58	2.0	14	00.413595	802.11 Ack
59	11.0	1536	00.415260	IP Fragment
60	11.0	14	00.415399	802.11 Ack
61	2.0	20	00.415537	802.11 PS-Poll
62	2.0	14	00.415676	802.11 Ack
63	11.0	1536	00.417371	IP Fragment
64	11.0	14	00.417510	802.11 Ack
65	2.0	20	00.417647	802.11 PS-Poll
66	2.0	14	00.417785	802.11 Ack
67	11.0	1536	00.419496	IP Fragment
68	11.0	14	00.419635	802.11 Ack
69	2.0	20	00.419772	802.11 PS-Poll
70	2.0	14	00.419919	802.11 Ack
71	11.0	1536	00.421873	IP Fragment
72	11.0	14	00.422012	802.11 Ack
73	2.0	20	00.422151	802.11 PS-Poll
74	2.0	14	00.422290	802.11 Ack
75	11.0	1536	00.423957	IP Fragment
76	11.0	14	00.424096	802.11 Ack
77	2.0	20	00.424234	802.11 PS-Poll
78	2.0	14	00.424374	802.11 Ack
79	11.0	1536	00.426040	IP Fragment
80	11.0	14	00.426179	802.11 Ack
81	2.0	20	00.426319	802.11 PS-Poll
82	2.0	14	00.426457	802.11 Ack
83	11.0	1536	00.428225	IP Fragment
84	11.0	14	00.428410	802.11 Ack
85	2.0	20	00.428548	802.11 PS-Poll
86	2.0	14	00.428686	802.11 Ack
87	11.0	1536	00.430360	IP Fragment
88	11.0	14	00.430499	802.11 Ack
89	2.0	20	00.430637	802.11 PS-Poll
90	2.0	14	00.430776	802.11 Ack
91	11.0	1536	00.432520	IP Fragment
92	11.0	14	00.432658	802.11 Ack
93	2.0	20	00.432797	802.11 PS-Poll
94	2.0	14	00.432934	802.11 Ack
95	11.0	1536	00.435421	IP Fragment
96	11.0	14	00.435559	802.11 Ack
97	2.0	20	00.435697	802.11 PS-Poll
98	2.0	14	00.435838	802.11 Ack
99	11.0	1304	00.437295	IP Fragment
100	11.0	14	00.437440	802.11 Ack
101	1.0	88	00.511999	802.11 Beacon
102	1.0	88	00.614397	802.11 Beacon
103	1.0	88	00.716935	802.11 Beacon

Figure 7.8: Analysis of Packets, AB-PSM Level with Two Skipped Beacond, Legacy 802.11 PSM enabled

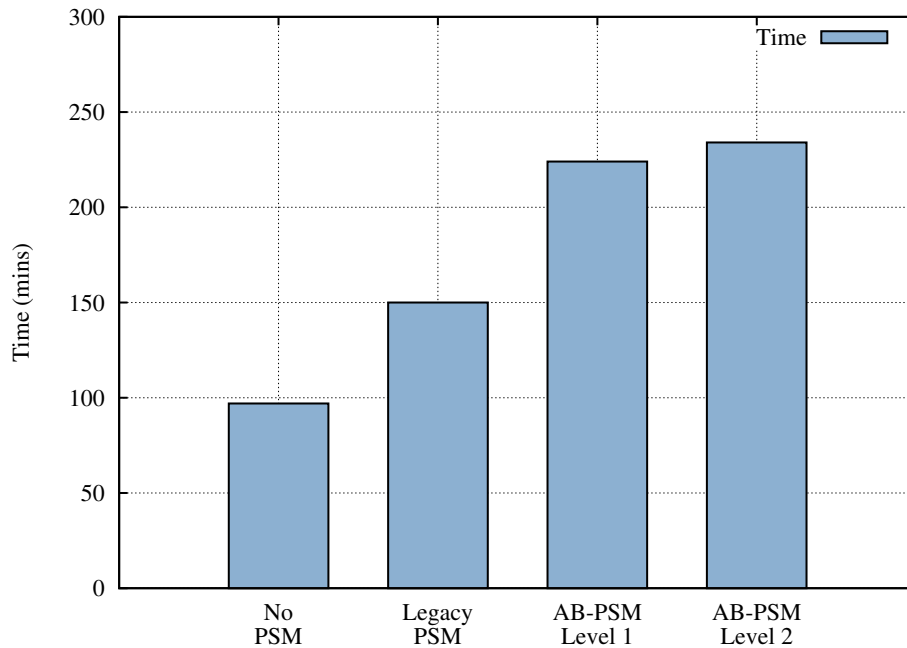


Figure 7.9: Battery Lifetime for Different Power Save Mechanisms

The client continues to send PS-Polls until all the data has been received. Then it can return to its low power sleep state. The next time it wakes is for the third beacon, where it sees no buffered traffic and goes back to sleep. Then at the fourth, the same process as the second will occur. Between the alternating beacons, there is no activity at the client and this is where the power savings are achieved.

The final test performed was with level 2 of AB-PSM enabled. This meant that the interval was increased by 12, to 300ms, and that packets of size 17,520B, divided into 12 packets of size 1460B, were used. The printout for this test is shown in Figure 7.8. The results are similar to the previous set except this time, two beacon intervals are skipped before any transmission takes place.

The results of the above tests are shown in Figure 7.9. This figure shows the effect on the battery life that each of the PSMs have.

7.4 Summary of Results

The results in these tests show a shorter overall battery life than in Figure 7.2. This is due to the fact that in these tests, the packets are much larger to represent video. The difference when AB-PSM is used is obvious. For the first level of AB-PSM, there is almost a 50% increase in battery life. When the next level of AB-PSM is implemented, an extra ten minutes is achieved. This extra battery life is significant for multimedia streaming because it could mean the difference between being able to complete a task and the battery depleting before the task is complete.

7.5 Summary of Chapter

This chapter presented the tests and results relating to AB-PSM, particularly in comparison to the legacy 802.11 PSM, which AB-PSM is an improvement of. A number of different test scenarios were described and their results given. The results show the significant increase in battery life that can be achieved when AB-PSM is used. The results were discussed and further tests were performed to find the effect of different parameters related to AB-PSM.

Chapter 8

Conclusions & Future Work

This chapter summarises the work in this thesis and presents possible future work.

8.1 Conclusion

There has been significant development in the area of multimedia streaming, particularly in relation to the devices and the applications; however the development of batteries tends to be slower. Consequently, in many cases the battery is unable to support the applications, depleting before the tasks complete. This thesis focused on multimedia streaming to mobile devices, particularly on ways of making it more battery friendly and less power intensive.

After describing the motivation behind the work in this thesis, the second chapter describes the details of some of the protocols used. These included the wireless solution chosen, the protocol used to send the data and the multimedia encoding standards.

Chapter 3 examined IEEE 802.11 wireless networks and described the power save techniques associated with them. Although there is a power save mechanism defined in the standard, it is not sufficient when used for multi-

media streaming. Chapter 4 looked at what other researchers have proposed to improve upon this legacy power save, and also examined some of the work that has been carried out in the more general areas of multimedia streaming and power saving in wireless networks. Each of the proposed solutions were described and their advantages and disadvantages were highlighted.

Chapter 5 presented the results of a detailed examination on how multimedia streaming affected the battery. By dividing the multimedia streaming process into three stages - decoding, reception and playing - it was possible to see exactly where the battery was used most. As the reception stage was the most power intensive, it was concluded that it was this stage on which the work would focus on. A possible Total Power Save algorithm is also described.

In chapter 6, the proposed solution for saving power in the reception stage was described. The Adaptive-Buffer Power Save Mechanism (AB-PSM) works in conjunction with the legacy power save to enhance power savings. No changes are required to be made to the standard, which is one of its main advantages. Tests show that significant power savings are achieved in comparison with the legacy PSM when AB-PSM is used, with little or no other effects to the user. A detailed description of what happens when AB-PSM is implemented is given and Chapter 7 presented the results of the tests that show the effectiveness of AB-PSM.

8.2 Contributions

The principal contribution of this thesis is the Adaptive-Buffer Power Save Mechanism (AB-PSM). There are many advantages to this proposed solution. It is easy to implement and requires no changes to be made to the standard. It makes use of and improves the legacy power save mechanism defined in IEEE 802.11, rather than trying to override it. The user is unaffected by the implementation of it and can choose different levels depending

on their preferences. It can extend the battery life of a mobile device to allow a task to be completed that otherwise could not have been.

A second contribution of this thesis is the investigation into the effect that multimedia streaming has on the battery of a mobile device. The idea of the multimedia streaming process being comprised of three stages is also proposed and based on this, the effect on the battery of the device are examined.

The results of the tests performed show that AB-PSM achieves significant power savings. When video packets are used, as would be typical of wireless multimedia streaming, the battery life is increased by over 50%. This extra battery lifetime would be very beneficial to a user wishing to perform a multimedia streaming task. When non-video packets are used, the increase in battery life increases by over 100%. Although this is a less realistic situation and does not relate to multimedia streaming to wireless devices, which is the principle focus of this thesis, these results are still important as they show that it may be possible to use AB-PSM for non multimedia streaming applications and to see even better results.

The results obtained in the investigation into how multimedia streaming affects the battery of a mobile device give a novel insight into this area. By breaking the multimedia streaming process into the three stages: reception, decoding and playing, it was possible to see exactly how the device and its battery were influenced by multimedia streaming. This could be very beneficial for future research work in the area.

8.3 Future Work

It would be beneficial to improve the adaptability of AB-PSM. By giving the user further control over the level of AB-PSM used and when to implement it, the power savings may increase. Therefore, when a streaming task begins, the user can decide whether or not to implement AB-PSM. There could also be an automatic switch off if the device is connected to a power supply and automatic switch on when the power supply is disconnected.

AB-PSM can be implemented on any device with IEEE 802.11 capability. It would be interesting to adapt so that it worked with other wireless solutions. This may increase its usability factor because in a lot of mobile phones, there is the capability to receive multimedia content without the use of 802.11. By making it portable among wireless protocols it would improve the utilisation of AB-PSM.

It would also be interesting to test AB-PSM on a number of different device types. Throughout the thesis, a PDA was used but to see it implemented on a laptop or mobile phone would be interesting, particularly when there is the possibility of non-user controlled activity such as an incoming phone call or text message. It may be beneficial to have time limits on the power saving. So for example, someone may need stringent power save on their mobile phone throughout their working day so that they can always receive a call. However, after work, say after 5.30 in the evening, the phone is used for personal reasons so the user is more interested in optimising the multimedia applications and is less worried about the battery.

It could also be beneficial to use a different method to measure battery life. For example, an examination into whether a measurement of current or voltage during the battery lifetime gives similar results to what were obtained in this work. For this to be achieved, suitable measurement equipment would be required.

The use of mobile devices for multimedia applications is becoming very popular and batteries are extremely important. There are countless ap-

proaches to take in order to extend battery life and infinite scope for extending the work described in this thesis.

AB-PSM provides significant power savings and is successful in allowing streaming tasks to complete. To gain further savings it may be beneficial to incorporate power savings for the decoding and playing stages as well. This could be achieved by developing the Total Power Save Algorithm (TPSA) described in this thesis.

Bibliography

- [1] M. Stemm and R. Katz, “Measuring and reducing energy consumption of network interfaces in hand-held devices,” *IEICE Transactions on Communications*, vol. E80-B, pp. 1125 – 31, August 1997.
- [2] G. Anastasi, M. Conti, and W. Lapenna, “Power saving policies for wireless access to TCP/IP networks,” *Proc. of the 8th IFIP Workshop on Performance Modelling and Evaluation of ATM and IP Networks (IFIP ATM & IP 2000)*, 2000.
- [3] R. Kravets and P. Krishnan, “Power management techniques for mobile communication,” *MobiCom’98. Proceedings of Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 157 – 68, 1998.
- [4] L. Zhong and N. Jha, “Energy efficiency of handheld computer interfaces: limits, characterization and practice,” *Proceedings of the Third International Conference on Mobile Systems, Applications, and Services (MobiSys 2005)*, pp. 247 – 60, 2005.
- [5] “IEEE 802.11: Wireless lan medium access control (MAC) and physical layer (PHY) specification,” standard, IEEE, 2000.
- [6] 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.

BIBLIOGRAPHY

- [7] J. Adams and G.-M. Muntean, "Power save adaptation algorithm for multimedia streaming to mobile devices," *IEEE International Conference on Portable Information Devices, Orlando, Florida, USA*, 2007.
- [8] J. Adams and G.-M. Muntean, "Adaptive-buffer power save mechanism for mobile multimedia streaming," *IEEE International Conference on Communications (ICC07), Glasgow, Scotland, UK*, 2007.
- [9] "IEEE 802.11b: Higher-speed physical layer (PHY) extension in the 2.4ghz band," standard, IEEE, 2001.
- [10] "Irish communications market - quarterly key data report," tech. rep., Commission for Communications Regulation, 2007.
- [11] J. Postel, "User datagram protocol," RFC 768, IETF, August 1980.
- [12] L.-A. Larzon, M. Degermark, and S. Pink, "Udp lite for real time multimedia applications," *HP Laboratories Technical Report*, vol. IRI, no. 1, pp. 6 –, 1999.
- [13] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications," tech. rep., IETF, 2003.
- [14] H. Schulzrinne, A. Rao, and R. Lanphier, "Real time streaming protocol (RTSP)," tech. rep., IETF, 1998.
- [15] J. Postel, "Transmission control protocol," RFC 793, IETF, August 1981.
- [16] D. Roberts, P. Sharkey, and P. Sandoz, "A real-time, predictive architecture for distributed virtual reality," *First Workshop on Simulation and Interaction in Virtual Environments*, pp. 279 – 88, 1995.
- [17] "ISO/IEC JTC1/SC29/WG1 - MPEG-1: Coding of moving pictures and associated audio for digital storage media at up to about 1,5 mbit/s," tech. rep., Moving Pictures Experts Group, 1996.

BIBLIOGRAPHY

- [18] “ISO/IEC JTC1/SC29/WG11 - MPEG-2: Generic coding of moving pictures and associated audio information,” tech. rep., Moving Pictures Expert Group, 2000.
- [19] “ISO/IEC JTC1/SC29/WG11 N4668 - overview of the MPEG-4 standard,” tech. rep., Moving Pictures Expert Group, 2002.
- [20] “H.263 : Video coding for low bit rate communication,” tech. rep., International Telecommunication Union, 2005.
- [21] “H.264 : Advanced video coding for generic audiovisual services,” tech. rep., International Telecommunication Union, 2005.
- [22] “IEEE 802.11a: High-speed physical layer in the 5GHz band,” standard, IEEE, 1999.
- [23] “IEEE 802.11g: Further higher-speed physical layer extension in the 2.4GHz band,” standard, IEEE, 2003.
- [24] “IEEE 802.11e: Medium access control (MAC) quality of service enhancements,” standard, IEEE, 2005.
- [25] “IEEE 802.11n: Standard for enhancements for higher throughput,” standard, IEEE, 2006.
- [26] “IEEE 802.16: Air interface for fixed broadband wireless access systems,” tech. rep., IEEE, 2004.
- [27] “IEEE 802.16e: Physical and medium access control layers for combined fixed and mobile operation in licensed bands,” tech. rep., IEEE, 2004.
- [28] “IEEE 802.15.1: Wireless medium access control (MAC) and physical layer (PHY) specifications for wireless personal area networks (WPANs),” tech. rep., IEEE, 2005.

BIBLIOGRAPHY

- [29] “Ultra-wideband (UWB) technology - enabling high-speed wireless personal area networks,” tech. rep., Intel Communications, 2004.
- [30] D. Kwon, S. S. Kim, C. Y. Park, and C. I. Jung, “Experiments on the energy saving and performance effects of IEEE 802.11 power saving mode (PSM),” *Lecture Notes in Computer Science*, vol. 3391, pp. 41 – 51, 2005.
- [31] X. Perez-Costa and D. Camps-Mur, “AU-APSD: Adaptive IEEE 802.11e unscheduled automatic power save delivery,” *ICC '06. IEEE International Conference on Communications, 2006*, vol. 5, pp. 2020 – 2027, 2006.
- [32] 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.
- [33] A. Chandrakasan, S. Sheng, and R. Brodersen, “Low-power CMOS digital design,” *IEEE Journal of Solid-State Circuits*, vol. 27, no. 4, pp. 473 – 84, 1992.
- [34] J. Flinn and M. Satyanarayanan, “Energy-aware adaptation for mobile applications,” *Operating Systems Review*, vol. 33, no. 5, pp. 48 – 63, 1999.
- [35] A. Sinha and A. Chandrakasan, “Energy aware software,” *VLSI Design 2000. Wireless and Digital Imaging in the Millennium. Proceedings of 13th International Conference on VLSI Design*, pp. 50 – 5, 2000.
- [36] C. Krishna and Y.-H. Lee, “Voltage-clock-scaling adaptive scheduling techniques for low power in hard real-time systems,” *Proceedings Sixth IEEE Real-Time Technology and Applications Symposium. RTAS 2000*, pp. 156 – 65, 2000.

BIBLIOGRAPHY

- [37] T. Ishihara and H. Yasuura, “Voltage scheduling problem for dynamically variable voltage processors,” *Proceedings. 1998 International Symposium on Low Power Electronics and Design (IEEE Cat. No.98TH8379)*, pp. 197 – 202, 1998.
- [38] T. Martin, *Balancing batteries, power and performance: System issues in CPU speed-setting for mobile computing*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, August 1999.
- [39] 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.
- [40] J. Pouwelse, K. Langendoen, and H. Sips, “Application-directed voltage scaling,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 5, pp. 812 – 26, 2003.
- [41] K. Choi, K. Dantu, W.-C. Cheng, and M. Pedram, “Frame-based dynamic voltage and frequency scaling for a MPEG decoder,” *IEEE/ACM International Conference on Computer Aided Design. IEEE/ACM Digest of Technical Papers (Cat. No.02CH37391)*, pp. 732 – 7, 2002.
- [42] H. Zhu and G. Cao, “A power-aware and QoS-aware service model on wireless networks,” *IEEE INFOCOM 2004*, vol. vol.2, pp. 1393 – 403, 2004.
- [43] P. Havinga and G. Smit, “Energy-efficient wireless networking for multimedia applications,” *Wireless Communications and Mobile Computing*, vol. 1, no. 2, pp. 165 – 84, 2001.
- [44] A. Acquaviva, E. Lattanzi, and A. Bogliolo, “Design and simulation of power-aware scheduling strategies of streaming data in wireless LANs,” *ACM MSWiM 2004 - Proceedings of the Seventh ACM Symposium on*

BIBLIOGRAPHY

- Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 39 – 46, 2004.
- [45] 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.
- [46] 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.
- [47] S. Mohapatra, R. Cornea, N. Dutt, A. Nicolau, and N. Venkatasubramanian, “Integrated power management for video streaming to mobile handheld devices,” *Proceedings of the eleventh ACM international conference on Multimedia*, pp. 582–591, 2003.
- [48] P. Pillai and K. G. Shin, “Real-time dynamic voltage scaling for low-power embedded operating systems,” *Operating Systems Review (ACM)*, vol. 35, no. 5, pp. 89 – 102, 2002.
- [49] M. Mesarina and Y. Turner, “Reduced energy decoding of MPEG streams,” *Multimedia Systems*, vol. 9, no. 2, pp. 202 – 13, 2003.
- [50] S. Mohapatra and N. Venkatasubramanian, “PARM: Power aware reconfigurable middleware,” *Proceedings - International Conference on Distributed Computing Systems*, pp. 312 – 319, 2003.
- [51] B. Noble, M. Satyanarayanan, D. Narayanan, J. Tilton, J. Flinn, and K. Walker, “Agile application-aware adaptation for mobility,” *Operating Systems Review*, vol. 31, no. 5, pp. 276 – 87, 1997.

BIBLIOGRAPHY

- [52] 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.
- [53] R. Krashinsky and H. Balakrishnan, "Minimizing energy for wireless web access with bounded slowdown," *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, pp. 119 – 130, 2002.
- [54] B. Prabhakar, E. Biyikoglu, and A. El Gamal, "Energy-efficient transmission over a wireless link via lazy packet scheduling," *Proceedings - IEEE INFOCOM*, vol. 1, pp. 386 – 394, 2001.
- [55] L. Y. Zhang, Y. Ge, and J. Hou, "Energy-efficient real-time scheduling in IEEE 802.11 wireless lans," *Proceedings - International Conference on Distributed Computing Systems*, pp. 658 – 667, 2003.
- [56] Q. Li, J. Aslam, and D. Rus, "Online power-aware routing in wireless ad-hoc networks," *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, pp. 97 – 107, 2001.
- [57] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, pp. 70 – 84, 2001.
- [58] 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.
- [59] F. Zhang and S. Chanson, "Proxy-assisted scheduling for energy-efficient multimedia streaming over wireless LAN," *Networking 2005. Networking Technologies, Services and Protocols; Performance of Computer and*

BIBLIOGRAPHY

- Communication Networks; Mobile and Wireless Communication Systems. 4th International IFIP-TC6 Networking Conference. Proceedings (Lecture Notes in Computer Science Vol. 3462)*, pp. 980 – 91, 2005.
- [60] Y. Wei, S. M. Bhandarkar, and S. Chandra, “A client-side statistical prediction scheme for energy aware multimedia data streaming,” *IEEE Transactions on Multimedia*, vol. 8, no. 4, pp. 866 – 874, 2006.
- [61] P. Pakdeepai boonpol and S. Kittitornkun, “Energy optimization for mobile MPEG-4 video decoder,” *Mobile Technology, Applications and Systems, 2005 2nd International Conference on*, pp. 1–6, November 2005.
- [62] 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.
- [63] S. Lee, “Low-power video decoding on a variable voltage processor for mobile multimedia applications,” *ETRI Journal*, vol. 27, no. 5, pp. 504 – 10, 2005.
- [64] S. Pasricha, S. Mohapatra, M. Luthra, N. Dutt, and N. Venkatasubramanian, “Reducing backlight power consumption for streaming video applications on mobile handheld devices,” *ACM/IEEE/IFIP Workshop on Embedded Systems for Real-Time Multimedia*, pp. 11–17, 2003.
- [65] 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.
- [66] S. Zanero and S. M. Savaresi, “Unsupervised learning techniques for an intrusion detection system,” *Proceedings of the ACM Symposium on Applied Computing*, vol. 1, pp. 412 – 419, 2004.

Publications Arising from this Research

- [1] J. Adams and G.-M. Muntean, "Power-Dependent Adaptation Algorithm for Mobile Multimedia Networking", *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, Las Vegas, Nevada, USA*, April 2006
- [2] J. Adams and G.-M. Muntean, "Power Save Adaptation Algorithm for Multimedia Streaming to Mobile Devices", *IEEE International Conference on Portable Information Devices, Orlando, Florida, USA*, 2007
- [3] J. Adams and G.-M. Muntean, "Adaptive-Buffer Power Save Mechanism for Mobile Multimedia Streaming", *IEEE International Conference on Communications (ICC07), Glasgow, Scotland, UK*, 2007