A Cross-layer Quality-oriented Energy-efficient Scheme for Multimedia Delivery in Wireless Local Area Networks

Yang Song

A Dissertation submitted in fulfilment of the

requirements for the award of

Doctor of Philosophy (Ph.D.)

to the

Dublin City University

Faculty of Engineering and Computing

School of Electronic Engineering

Supervisor: Dr. Gabriel-Miro Muntean

January, 2014

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work, and that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, 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:

ID No.:

Date:

Acknowledgment

First and foremost, I would love to thank my dear parents. You are the most valuable treasures in my life and I feel so lucky to be your daughter. I will never be who I am without you. You are always the most supportive friends and helped me through all difficulties. Thank you dad for encouraging and supporting me at every stage of my life. Mum thank you for your endless love and care, I would not have made it this far without you.

A special acknowledgement is given to my boyfriend Ruiqi, who has shared laughter and tears with me for five years. He has been a true and great advisor for both life and academic research and has unconditionally supported me during my good and bad times. Thank you for having faith in me and instilling confidence in me. I've gained so much drive and an ability to tackle challenges through your help.

I would like to express my deepest appreciation to my principal supervisor, Dr. Gabriel-Miro Muntean, for his consistent efforts and help. Thank you for your guidance which helped me in all the time of research during the past years. I've benefited so much from your wisdom and scholarly input. You are always very encouraging and supportive, otherwise I will hardly make the academic achievements that I've made today. You are always the best supervisor and a good friend. I could not have imagined having a better advisor and mentor. Besides, I am extremely grateful to my co-supervisor Dr. Bogdan Ciubotaru who shared his extremely valuable experience with me. Bogdan thank you for pushing me through the path towards a successful PhD. Thank you for inspiring me and helping me with my papers and difficulties encountered during the research.

Many thanks to dozens of people who have helped and taught me immensely at the Performance Engineering Lab. Ramona thank you for always telling me everything is going to be fine during the depressing times, Irina you are such a nice girl and I am always cheered up after talking to you and being affected by your optimism. My sincere thanks also goes to Zhenhui and his lovely fiancee Xin, who never hesitated to share their experience and valuable advice with me. My heartful thanks are given to my dear colleagues for their

consistent support: Martin, Lejla, Shengyang, Ting, Longhao, Ronan and Quang.

Also I want to thank the technical staff, especially Robert Clare and Conor Maguire from the School of Electronic Engineering. You are always a great helper and offer many professional advice.

Last but not least, I gratefully acknowledge Prof. Amaia Arrinda, Prof. Noel E O'Connor and Dr. Conor Brennan for making this happen.

Dublin, November 2013

Yang Song

List of Publications

- Y. Song, B. Ciubotaru, and G.-M. Muntean, “A Slow-start Exponential and Linear Algorithm for Energy Saving in Wireless Networks,” *Broadband Multimedia Systems and Broadcasting (BMSB), 2011 IEEE International Symposium on* , pp.1–5, June 2011.

- Y. Song, B. Ciubotaru, and G.-M. Muntean, “Application-aware Adaptive Duty Cycle-based Medium Access Control for Energy Efficient Wireless Data Transmissions,” *Local Computer Networks (LCN), 2012 IEEE 37th Conference on* , pp.172–175, Oct. 2012.

- Y. Song, B. Ciubotaru, and G.-M. Muntean, “Q-PASTE: A Cross-Layer Power Saving Solution for Wireless Data Transmission,” *IEEE International Conference on Communications (ICC), IEEE International Workshop on Energy Efficiency in Wireless Networks & Wireless Networks for Energy Efficiency (E2Nets)*, Jun. 2013.

- Y. Song, B. Ciubotaru, and G.-M. Muntean, “STELA: A Transceiver Duty Cycle Management Strategy for Energy Efficiency in Wireless Communications”, *Local Computer Networks (LCN), 2012 IEEE 38th Conference on* , Oct. 2013.

Abstract

Wireless communication technologies, although emerged only a few decades ago, have grown fast in both popularity and technical maturity. As a result, mobile devices such as Personal Digital Assistants (PDA) or smart phones equipped with embedded wireless cards have seen remarkable growth in popularity and are quickly becoming one of the most widely used communication tools. This is mainly determined by the flexibility, convenience and relatively low costs associated with these devices and wireless communications. Multimedia applications have become by far one of the most popular applications among mobile users. However this type of application has very high bandwidth requirements, seriously restricting the usage of portable devices. Moreover, the wireless technology involves increased energy consumption and consequently puts huge pressure on the limited battery capacity which presents many design challenges in the context of battery powered devices. As a consequence, power management has raised awareness in both research and industrial communities and huge efforts have been invested into energy conservation techniques and strategies deployed within different components of the mobile devices.

Our research presented in this thesis focuses on energy efficient data transmission in wireless local networks, and mainly contributes in the following aspects:

1. **Static STELA**, which is a Medium Access Control (MAC) layer solution that adapts the sleep/wakeup state schedule of the radio transceiver according to the bursty nature of data traffic and real time observation of data packets in terms of arrival time. The algorithm involves three phases– slow start phase, exponential increase phase, and linear increase phase. The initiation and termination of each phase is self-adapted to real time traffic and user configuration. It is designed to provide either maximum energy efficiency or best Quality of Service (QoS) according to user preference.

2. **Dynamic STELA**, which is a MAC layer solution deployed on the mobile devices and provides balanced performance between energy efficiency and QoS. Dynamic STELA consists of the three phase algorithm used in static STELA, and additionally employs a

traffic modeling algorithm to analyze historical traffic data and estimate the arrival time of the next burst. Dynamic STELA achieves energy saving through intelligent and adaptive increase of Wireless Network Interface Card (WNIC) sleeping interval in the second and the third phase and at the same time guarantees delivery performance through optimal WNIC waking timing before the estimated arrival of new data burst.

3. **Q-PASTE**, which is a quality-oriented cross-layer solution with two components employed at different network layers, designed for multimedia content delivery. First component, the Packet/ApplicaTion manager (PAT) is deployed at the application layer of both service gateway and client host. The gateway level PAT utilizes fast start, as a widely supported technique for multimedia content delivery, to achieve high QoS and shapes traffic into bursts to reduce the wireless transceiver's duty cycle. Additionally, gateway-side PAT informs client host the starting and ending time of fast start to assist parameter tuning. The client-side PAT monitors each active session and informs the MAC layer about their traffic-related behavior. The second component, dynamic STELA, deployed at MAC layer, adaptively adjusts the sleep/wake-up behavior of mobile device wireless interfaces in order to reduce energy consumption while also maintaining high Quality of Service (QoS) levels.

4. A comprehensive **survey** on energy efficient standards and some of the most important state-of-the-art energy saving technologies is also provided as part of the work.

Abbreviations and Acronyms

1G	First Generation of Wireless Telephone Technology
2G	Second Generation of Wireless Telephone Technology
2.5G	Second and Half Generation of Wireless Telephone Technology
3G	Third Generation of Mobile Telecommunications Technology
3GPP	3rd Generation Partnership Project
4G	Fourth Generation of Mobile Telecommunications Technology
ACK	Acknowledgement
AID	Association ID
AMPS	Advanced Mobile Phone System
AP	Access Point
ARQ	Automatic Repeat Request
ASN	Access Service Network
AUC	Authentication Centre
BE	Best Effort
BSC	Base Station Controller
BSS	Base Station Subsystem
BSS	Base Service Set
BTS	Base Transceiver Stations
CBR	Constant Bit Rate
CDMA	Code Division Multiple Access

CDN	Content Delivery Network
CN	Core Network
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS	Clear-To-Send
DCCP	Datagram Congestion Control Protocol
DCF	Distributed Coordination Function
DCT	Discreet Cosine Transform
DIFS	DCF Interframe Space
DSSS	Direct-Sequence Spread Spectrum
EDGE	Enhanced Data rates for Global Evolution
EIR	Equipment Identification Register
ertPS	extended real-time Polling Service
ETSI	European Telecommunications Standards Institute
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FR	Full Reference
FTP	File Transfer Protocol
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HLR	Home Location Register
HTTP	Hypertext Transfer Protocol

IBSS	Independent BSS
ICMP	Internet Control Message Protocol
ICT	Information and Communications Technologies
IEEE	Institute of Electrical and Electronics Engineers
IMAP	Internet Message Access Protocol
IMT-2000	International Mobile Telecommunications-2000
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ITU	International Telecommunications Union
JPEG	Joint Photographic Experts Group
MAC	Medium Access Control
MIMO	Multiple Input Multiple Output
MOS	Mean Opinion Score
MPEG	Moving Pictures Experts Group
MSC	Mobile Service Switching Centre
MSS	Maximum Segment Size
NAV	Network Allocation Vector
NMT	Nordic Mobile Telephony
NR	No Reference
nrtPS	non-real-time Polling Service

NSP	Network Service Provider
NSS	Network Switching Subsystem
OFDM	Orthogonal Frequency-Division Multiplexing
OMC	Operation Maintenance Centre
OSS	Operation and Support System
PAT	Packet/ApplicaTion manager
PCF	Point Coordination Function
PDA	Personal Digital Assistants
PDV	Packet Delay Variation
PESQ	Perceptual Evaluation of Speech Quality
PEVQ	Perceptual Evaluation of Video Quality
POP	Post Office Protocol
PSM	Power Saving Mode
PSQM	Perceptual Speech Quality Measurement
QoE	Quality of Experience
QoS	Quality of Service
RR	Reduced Reference
RTCP	RTP Control Protocol
RTP	Real-Time Transport Protocol
RTT	Round Trip Time
rtPS	real-time Polling Service

RTS	Request-To-Send
RTSP	Real Time Streaming Protocol
Rx	Receive
SCTP	Stream Control Transmission Protocol
SDP	Session Description Protocol
SIFS	Short Inter-Frame Space
SIM	Subscriber Identity Module
SMTP	Simple Mail Transfer Protocol
STELA	Slow sTart Exponential and Linear Algorithm
TCP	Transmission Control Protocol
TDMA	Frequency Division Multiple Access
TIM	Traffic Indication Map
Tx	Transmit
UDP	User Datagram Protocol
UGS	Unsolicited Grant Service
UMTS	Universal Mobile Telecommunications System
UE	User Equipment
UTRAN	UMTS Radio Access Network
VBR	Variable Bit Rate
VLR	Visitor Location Register
VoIP	Voice over IP

WCDMA	Wideband Code Division Multiple Access
Wi-Fi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Networks
WMAN	Wireless Metropolitan Area Network
WNIC	Wireless Network Interface Card
WPAN	Wireless Personal Area Network
WWAN	Wireless Wide Area Network

Symbols

Table 1 List of Symbols

Symbol	Description
W_s, W_{thre}	Size, and threshold of sleeping window
I_{ob}	Interval between two consecutive bursts observed during parameter tuning
T_{fs}, r_{fs}	Total length, and data rate of fast start
t_{bf}	Buffering period
T_{bf}	Maximum buffering period without compromising QoS
r_{ec}	Encoding rate
s_{po}, t_{po}	size and duration of playout
s_{ts}	The size of scheduled data after fast start period
s_{bf}	The size of data that is going to be released
t_{ts}	The time elapsed after fast start
$T_{bf_{min}}$	Lower bound of T_{bf}
t_{sl_N}	The sleeping interval of the Nth sleep/wakeup cycle
t_{ds}	Actual burst scheduling time
D_t, D_{ss}, D_{bf}, D_{sl}, D_{pp}, D_{pr},	Total delay, server-side delay, buffering delay, sleeping delay, propagation delay and processing delay
I_{ac},	Delay without PSM
D_{ex}	Extra delay caused by STELA
I_{bc}	Beacon duration.
I_{ac}	Actual arrival time of the burst

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter discusses the exceptional growth in wireless communications, their user base and multimedia applications which put high pressure on device battery life. Some energy saving solutions implemented by both battery manufactures and software researchers are briefly introduced. The novel cross-layer solution for balancing energy conservation and user quality of experience levels is then introduced, and the contributions of this thesis are then listed.

1.1 Research Motivation

With the fast pace of development in network, wireless communications are realized and becoming more and more significant in terms of both daily life and professional activities, such as military and medical applications. The trend is to switch from traditional devices such as desktop PCs to wireless-enabled portable ones such as Personal Digital Assistants (PDA) and smartphones. The wide use of this type of devices can be attributed to the simplicity of deployment of wireless networks.

Wireless communication technologies have served people for a long time in delivering multimedia content via broadcasting (e.g. terrestrial television, radio broadcasting, and mobile TV [1] [2] [3]). Moreover, wireless communication technologies enable user support from different points of view: people are able to connect to the Internet or communicate with each other even when on the move. The number of people accessing mobile Internet has reached half a million in 2009 [1] and is expected to exceed 7.1 billion by 2015 according to Cisco [2]. Statistics have shown that in Q1 2013 tablets exceeded traditional desktop devices for conversion rates for the first time [3]. Figure 1.1 illustrates a wireless heterogeneous communication environment. One of the most significant benefits brought by wireless technologies is the ability to keep in touch with long distance contacts via applications like Voice over IP (VoIP) and video conferencing such as Skype. Apart from audio and video exchange, users can also search for certain information using search engines, check emails or use wireless communications for entertaining purposes. For example, people can entertain themselves using video applications such as YouTube or social networking applications such as Facebook and Twitter, which are becoming highly popular tools to connect with other people. Statistics [4] have shown that as of 2011, there are over 500 million active Facebook users, which means this service is used by 1 in every 13 people on Earth, and the number is still growing. Among this, over 100 million users access their accounts through mobile devices and these mobile users are twice as active as those who do not use mobile devices. Besides entertaining purposes, people are allowed to do business, purchase items, seek health and medical advice, etc.

Wireless access to the Internet is enabled by technologies such as cellular and broadband networks, including Global System for Mobile Communication (GSM), Wireless Local Area Networks (WLAN) and Wireless Personal Area Network (WPAN) etc. These technologies differ in terms of specifications such as bandwidth, minimum data rate guaranteed, range of area covered, and they are applied for different purposes providing specific

[1] Global Mobile Statistics 2012-http://mobithinking.com/mobile-marketing-tools/latest-mobile-stats-mobile-internet-access

[2] Cisco visual networking index: Forecast and methodology, 2011-2016.http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf

[3] http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/

[4] http://www.facebook.com/notes/facebook/500-million-stories/409753352130

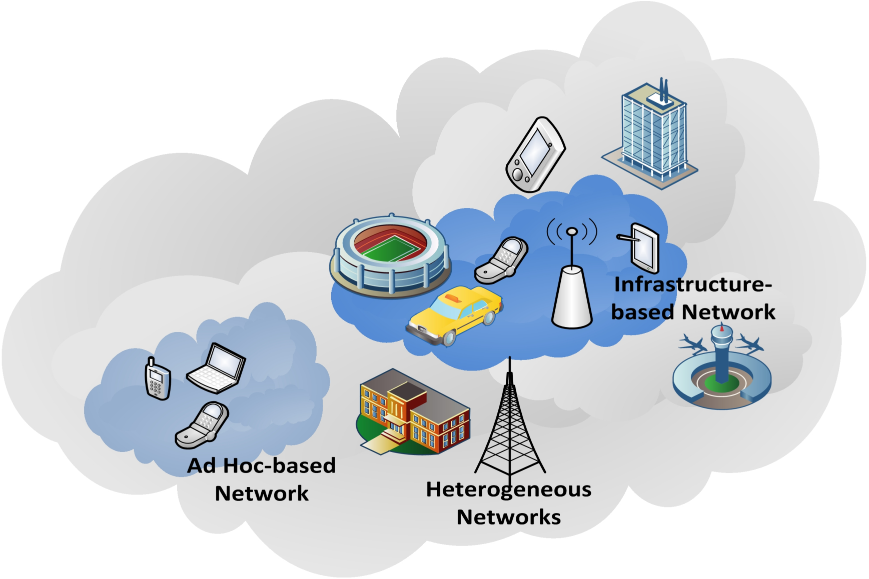


Figure 1.1 Wireless communication in heterogeneous network environment

wireless services.

Mobility and wireless connectivity imply that devices should have network access anywhere and anytime. However, mobility and wireless communication require most devices use battery power, which has very limited capacity and is a scarce resource. This will definitely restrict the usage of these devices at least in terms of duration between recharges with a significant negative impact on user satisfaction. Major battery manufacturers and research institutes such as Argonne National Laboratory [5], Zic Matrix Power [6], and Panasonic [7] have put huge efforts into the development of rechargeable batteries with large capacity.

On the other hand, multimedia applications have become by far the most popular among mobile users [4]. It has been pointed out that online video usage measured as time spent on viewing videos online increased by 45% from 2010 to 2011 in the U.S. and the total number of video streams grew by 31.5% to 14.5 billion streams [8]. According to Allot [9], video streaming as the fastest growing application drives the growth of mobile data band-

[5] Argonne National Lab-http://www.anl.gov/

[6] ZPower- http://www.zpowerbattery.com/

[7] Panasonic-http://www.panasonic.com/

[8] http://blog.nielsen.com/nielsenwire/online_mobile/january-2011-online-video-usage-up-45/

[9] Mobile trends report h1,2011. http://www.allot.com/MobileTrends_Report_H1_2011.html, 2011.

width usage and accounts for 37% of mobile bandwidth usage in 2011. The multimedia-based traffic will account for 49% and 53 percent% of the total data consumption over smartphones and tablets in 2017, respectively [5]. However this type of applications has very high bandwidth requirements and consequently puts high pressure on the limited battery capacity, seriously restricting the usage of portable devices [6]. Therefore the ability of saving energy and maximizing battery life has become a critical issue for both companies and academic research centers. The importance of energy conservation not only lies in the requirements of mobile users, but also attracts interests from the environmental point of view. Studies have pointed out the key role Information and Communications Technologies (ICT) plays in addressing the problem of global climate change and facilitating low carbon footprint and also the influence of ICT technologies on the way our society works and people behave [10].

1.2 Problem Definition

Substantial effort has been made to improve energy efficiency through hardware improvements [7]. One main focus of research for battery companies is to expand the battery capacity for battery companies. Argonne National Laboratory developed composite battery materials to increase the battery capacity by 30% [11]. Zinc Matrix Power introduces silver-zinc batteries as an alternative to lithium-ion batteries as silver-zinc batteries could add several hours to the time that laptops can run between charges [12]. Panasonic developed new Li-ion rechargeable batteries that improve the battery capacity by 30% compared with the largest capacity of any cell of the same size [13].

Another trend in prolonging battery life is to explore energy efficient solutions for wireless communications. Those research efforts are built on top of the existing protocol stack, and achieve energy efficiency mainly by modifying protocols at a single or multiple

[10] http://www.smart2020.org/_assets/files/02_Smart2020Report.pdf, 2008.

[11] http://www.technologyreview.com/news/409860/longer-lasting-batteries-for-laptops/?mod=related, 2008.

[12] http://www.technologyreview.com/news/406659/safer-higher-capacity-batteries/, 2006.

[13] http://techon.nikkeibp.co.jp/article/HONSHI/20100223/ 180545/, 2010.

layers. Some solutions extend battery life by adapting the behaviour of applications, for example the streaming pattern of multimedia content at the application layer, such as [8] and [9]. Some studies, for example [10] and [11] sacrifice an acceptable degree of traffic reliability to increase energy efficiency at transport layer. There is another body of studies that chooses energy efficient routes that can save energy for those battery critical nodes instead of traditional routes of the least hops or shortest distance as part of network layer protocols [12] [13]. Some other MAC layer based works enable periodical turn-off of radio transceiver to prevent energy waste [14] [15].

However, energy saving is achieved in return for compromising other performance aspects. Adaptive streaming at application layer may involve packet loss due to deliberate data drop and deliberate reduced quality at the server side. Energy efficiency at transport layer may compromise user experience due to longer packet delays. The delivery time of packets may be much longer if energy efficient paths are chosen instead of shortest paths at network layer. Serious jitter may be experienced if radio transceivers are switched off for an unacceptable duration and packets could not arrive at their destination in time.

Currently, there are a few solutions that employ cooperation among multiple layers to achieve energy efficiency maximization for multimedia streaming. Those solutions aim at maximizing energy saving through message flows within the protocol stack and the wise combination of various energy efficient solutions at multiple layers. Moreover, it explores and balances the trade-off between energy saving and quality of service delivered so that energy performance is increased without compromising quality of experience too much.

1.3 Solution Overview

This research focuses on finding a solution to balance energy efficiency and quality of service levels when delivering video content over wireless local area networks. Generally speaking, in order to improve battery lifetime, the wireless interface of mobile devices are switched off to low power saving mode when there is no data arriving. At the same time, quality of service levels are not compromised as the interfaces are switched on before the arrival of data through various mechanisms.

As mentioned, this research provides a set of solutions for energy efficient data delivery over wireless local networks. This set is composed of three solutions.

The first solution proposed, i.e. **static Slow sTart Exponential and Linear Algorithm (STELA)**, increases battery lifetime through wise scheduling of the wireless interface card. STELA is a MAC-layer based mechanism which runs on the client side. The novel proposed MAC layer solution intelligently adapts the sleeping schedule of the mobile terminals radio interface for data transfer over wireless networks in order to reduce the energy consumption while still maintaining high delivery performance. The solution divides the whole process into three phases: slow start, exponential, and linear increase, and adjusts sleeping window of wireless interfaces dynamically according to real time traffic.

Dynamic STELA balances the energy efficiency and delivery quality and employs similar approaches as static STELA. However, the solution adopts an extra traffic modelling algorithm following the data flow which analyzes historical traffic data and estimates the arrival time of the next burst, to additionally improves quality of experience while saving energy for wireless devices.

Besides MAC layer energy saving, energy efficiency can also be achieved in upper layers, for example the choice of transport layer protocol would influence the number of packets transmitted, the retransmission scheme, and the congestion control mechanism, which significantly impact the overall energy consumption. Moreover, adaptive streaming at application layer can also be wisely chosen according to user requirements. Therefore, the solutions chosen in real time at each layer form a set of parameters which can be adjusted either in favor of energy saving or quality of service. **Q-PASTE** is proposed as cross-layer solution that achieves energy efficiency for wireless multimedia content delivery without degrading significantly quality of service levels. As part of the cross-layer mechanism, dynamic STELA is employed at the client side to save energy for the wireless interface, while Packet/ApplicaTion manager (PAT) at application layer helps prolong the interval between bursts through deliberate traffic shaping at the service gateway. In the protocol stack, message flow is utilized to pass information among different layers. The cross-layer solution adopts the message exchange scheme in order to make the optimal decision on the sleeping schedule of the wireless interface. The cross-layer solution will not only maintain energy savings achieved by MAC layer modifications, but also improve user quality of experience levels.

1.4 Contributions

The research mentioned in this thesis makes the following four major contributions.

- **Static STELA**. The first contribution of this research is the static Slow-sTart Exponential and Linear Algorithm for energy saving in wireless networks (STELA), a novel Medium Access Control (MAC) layer power saving scheme for wireless data transmission proposed to achieve energy consumption or high delivery performance. Studies [16] [17] have proved that traffic flow shows a significant amount of regularity and exhibits burstiness. This means that packets are sent in bursts with relatively short inter-arrival duration while long inter-arrival duration is observed between bursts. STELA, to the best of the author's knowledge, is the first algorithm to consider not only the bursty nature but also the real time traffic pattern at the MAC layer. For example, the power saving mechanism adopted by [18] employs the exponential increase phase to increase sleeping interval between two consecutive bursts, but it does not consider the impact of real time traffic on the threshold value adopted. On the other hand, static STELA employs three different adaptation strategies and adds to the existing slow start and exponential increase phases a novel linear increase stage. The adaptive algorithm renders a significant decrease in energy consumption, and the perceived quality of service is also maintained at high levels. The solution can be configured to either achieve maximum energy efficiency or provide high quality of service, depending on which factor is of greater importance to the users.

- **Dynamic STELA**. Another contribution made in this thesis is dynamic STELA which adds an additional traffic modelling-based threshold adjusting phase which is employed at the beginning of the establishment of each data flow. It is based on the regularity of traffic as observed in some studies [16] [19], and historical data is used to adjust a proper point which determines the termination of the second phase and initiation of the third phase in STELA. Based on the observation of traffic, STELA is able to predict the arrival pattern of following data burst and switch on the network interface in time before traffic arrives. Therefore the interface is awake when data arrives and the degradation in QoS in terms of delay, jitter, PSNR which are introduced by switching off the WNIC is minimized. The threshold adjusting phase guarantees better quality of service provided to the users and at the same time improves energy efficiency. Different from static STELA, this solution provides balanced performance between energy efficiency and QoS, without the need of user configuration.

- **Q-PASTE**. The third contribution of this research is Q-PASTE, which is a quality-oriented cross-layer solution for energy efficient multimedia delivery in wireless LAN. Q-PASTE balances the need for: 1) increased energy efficiency/battery lifetime and 2) high user quality of experience required in multimedia streaming applications. The cross-layer solution mainly consists of two components. First, the Packet/ApplicaTion manager (PAT) is employed as a novel traffic shaping scheme at the application layer of both the gateway and the client host. According to experimental results shown in [20], traffic bursts not only prolong the sleeping interval of wireless interface, but are also beneficial to efficient energy use and battery life. Therefore, at the service gateway, PAT utilizes first a fast start approach to support smooth multimedia playback, then shapes traffic from the server into bursts to reduce the wireless transceivers duty cycle for energy efficiency. At client side, PAT performs traffic monitoring and provides the Medium Access Control (MAC) layer with session-specific traffic-related information. Second, dynamic STELA is employed at the MAC layer of the client side with the tuning procedure assisted by additional information provided by PAT in order to ensure data arrives at the client without serious delays. The solution can be applied to various multimedia streaming applications such as well known online audio-visual services (e.g. YouTube).

- **A comprehensive survey**. Finally, a comprehensive survey on energy efficient standards and some of the most important state-of-the-art energy saving technologies is also provided as part of this work.

1.5 Report Structure

The remainder of the document is organized as follows. Chapter 2 explores the existing standards related to wireless communications and quality evaluation. The wireless communication standards are categorized according to the evolution of technologies. The standards at different network protocol stacks are also presented. Quality evaluation tools used for both quality of service and quality of experience measurement are introduced. Chapter 3 investigates some of the most important state-of-the-art energy saving solutions which are classified following the layer structure of the protocol stack. Some solutions which take a cross-layer approach towards energy conservation are also presented. In Chapter 4, the architecture and details of the proposed algorithms are presented, while Chapter 5 explains the testing environment and testing scenarios for each solution. Chapter 6 details the simulation results and result analysis through comparison with other solutions and standards. In Chapter 7, the prototype-based testing of Q-PASTE is conducted with results and analysis presented. Finally, Chapter 8 summarizes the work and points out the possible future work and their potential benefits.

Chapter 2

Background Technologies

This chapter introduces the background technologies that lead the evolution of wireless communications. It starts with a brief introduction to cellular wireless networks which are categorized according to generations of development witnessed in the last few decades. WLAN, WMAN and WPAN as three classes of broadband wireless technologies that are widely deployed are then presented. Following that is the discussion on protocol stack which is widely used as an implementation of a computer networking protocol suite. Moreover, standards that are widely used at each layer of the protocol stack are presented and discussed in relation to energy conservation. Finally, the most widely used tools for measuring quality of service and quality of experience levels are introduced.

2.1 Wireless Networks Overview

The past few decades have witnessed an explosive growth of both the Internet and mobile telephony services which together lead to the incredibly fast development of wireless networks. Data transmission and reception over wireless networks are built on several primary technologies that are widely supported by mobile devices. These technologies are categorized based on their range into Wireless Wide Area Network (WWAN), Wireless

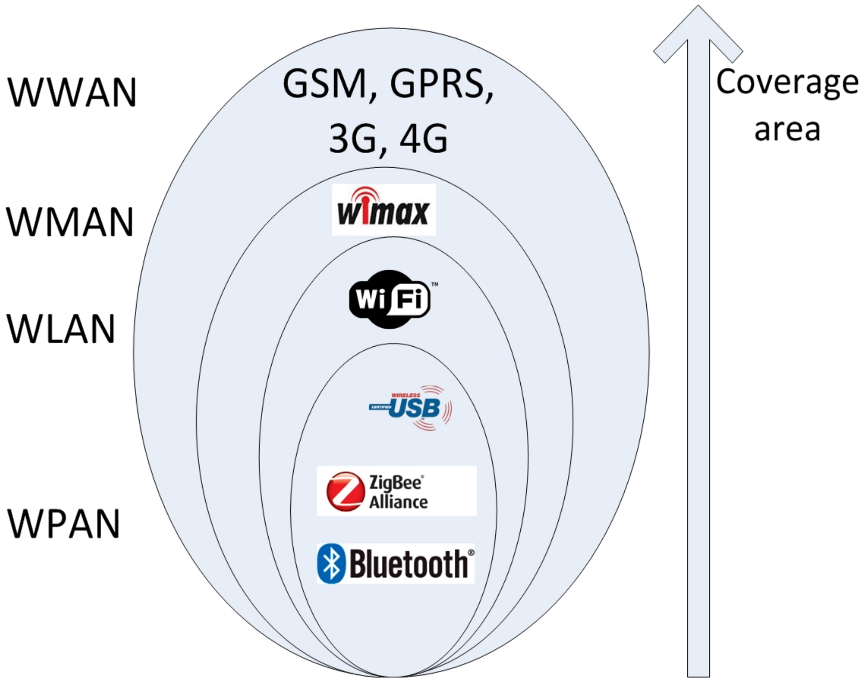


Figure 2.1 Wireless technologies range-based classification

Local Area Network (WLAN), Wireless Personal Area Network (WPAN) and Wireless Metropolitan Area Network (WMAN).

WWAN is a form of wireless network that provides wide range of network access to mobile devices compared to local area networks. It is then categorized in this thesis into several groups of technologies, including 2G, 2.5G, 3G and 4G, based on the timing and generation of development in wireless communications of each standard. Unlike WWAN which supports tens of kilometres of wireless service, WMAN is designed to provide network access of up to several kilometres for mobile devices. WLAN serves shorter distance but with higher bitrate, while WPAN is designed for personal area and has the smallest coverage area. A comparison based illustration of each category is depicted in Figure 2.1. These network types will be discussed with more details in the following sections.

2.1.1 Wireless Wide Area Network (WWAN)

2.1.1.1 Second Generation (2G)

The first generation (1G) telephone technologies were designed for analogue voice delivery until they were replaced by the second generation (2G) technologies. There were a number of standards designed across the world independently for 1G which means the equipments were not supported across boundaries of countries or regions. For example, **Nordic Mobile Telephony** (NMT) is the first fully automatic cellular 1G phone system, and it was deployed in eastern Europe, Russia etc, while Advanced Mobile Phone System (AMPS) developed by Bell Labs was widely used in the North America and Australia. Replacing 1G, the second generation wireless telephone technologies move from analogue to digital signals and were mainly designed for voice services and slow data transmissions.

Global System for Mobile Communications (GSM) [21] developed by the European Telecommunications Standards Institute (ETSI) is the dominant standard among 2G technologies. It was designed as a replacement for first generation cellular networks and is a standard still in use. Compared with 1G, it provides better call quality and a low cost call alternative, simple messaging service. Moreover it is widely supported, enabling roaming and solving the boundary problem occurred in 1G. However, the maximum cell site range of GSM is only 120 km, although it is expanded from the old limit of 35 km[1].

A typical GSM system consists of three parts: base station subsystem, switching system and operation and maintenance centre. The Base Station Subsystem (BSS) consists of multiple Base Transceiver Stations (BTS) and a Base Station Controller (BSC) which provides all control functions and physical link between Mobile Service Switching Centre (MSC) and BTS. BTS provides the radio interface with mobile subscribers which are normally mobile devices registered with a Subscriber Identity Module (SIM). The Network Switching Subsystem (NSS) is responsible for connecting calls between a GSM user and another party such as another GSM user. As the core element of the switching subsystem,

[1] http://www.allbusiness.com/electronics/ computer-electronics-manufacturing/6838169-1.html

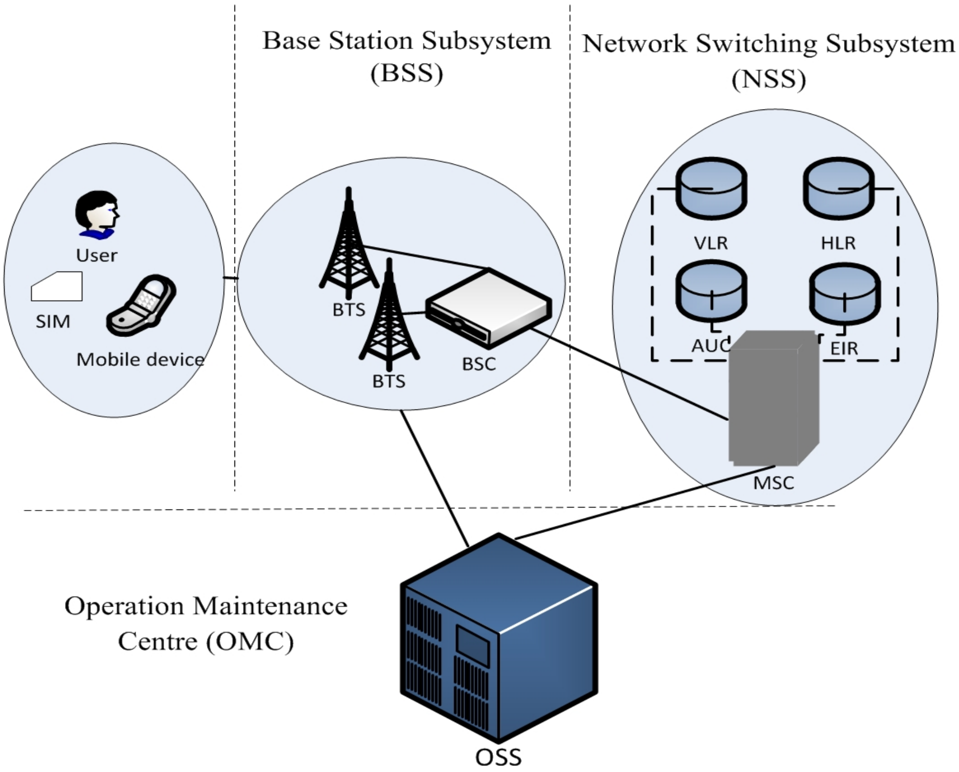


Figure 2.2 GSM architecture

MSC is responsible for call setup and maintenance, resource management, call handover and data encryption. Call processing and subscriber information is contained in several databases that are used by the MSC. The databases are divided into Home Location Register (HLR) which stores permanent data about subscribers, Visitor Location Register (VLR) which stores temporary information about subscribers, Authentication Centre (AUC) used for user identification and Equipment Identification Register (EIR) which stores data about equipment identity. The Operation Maintenance Centre (OMC) supervises the operation of switching system blocks connected to it. It basically monitors and reports traffic, taking care of call failures. The implementation of OMC is called the Operation and Support System (OSS) connected to all equipment in the switching system and to the BSC. An illustration of the components in a GSM network is shown in Figure. 2.2.

2.1.1.2 Two Point Five Generation (2.5G)

The two point five generation (2.5G) is an extension of the 2G services. The main novelty is the introduction of packet-switched networks into the circuit-switched domain, used in traditional 2G networks. The two widely supported technologies **General Packet Radio Service** (GPRS) [22] and **Enhanced Data rates for Global Evolution** (EDGE) [23] are standards evolved from 2G providing Internet services to existing 2G networks.

GPRS is a technology for GSM networks, which adds packet switching protocols. It was first standardized by the European Telecommunications Standards Institute (ETSI) and is now maintained by the 3rd Generation Partnership Project (3GPP)[2]. The standard improves the efficiency of network resources as information is broken up into packets and resources are allocated during the handling of individual packets only. The packet switching feature implies that GPRS provides best effort service instead of the guarantee of quality of service (QoS), provided by circuit switched networks.

3GPP EDGE is sometimes considered as a third generation standard but it is generally deployed on existing 2G networks. It is a technology designed for mobile phones to improve data transmission rates. EDGE is compatible with existing GSM networks and offers a maximum speed of 384 kbps to IP-based networks.

2.1.1.3 Third Generation (3G)

The third generation mobile telecommunications (3G) standardized by the International Mobile Telecommunications-2000 (IMT-2000) is a generation of standards designed for wireless communications. It is required that 3G provides a data rate of at least 200 kbps while most 3G services offer higher speed than the requirements.

Universal Mobile Telecommunications System (UMTS) [24], also known as WCDMA (Wideband Code Division Multiple Access) developed by the 3GPP is a widely deployed 3G technology based on GSM networks. It has both circuit switched and packet switched

[2] 3GPP-http://www.3gpp.org/.

elements. UMTS achieves higher spectral and bandwidth efficiency through utilization of wide-band code division multiple access. The data transfer rate has been increased to 45 Mbps which is a significant improvement. UMTS specifications deal with three main components of the network: Core Network (CN), The UMTS Radio Access Network (UTRAN) and User Equipment (UE). The core network uses the same core network standard as GSM/EDGE. Although the same structure is supported which indicates easy migration for existing GSM operators, the cost of purchasing spectrum licenses and overlaying UMTS is high. UTRAN consists of one or multiple base stations which directly provide connection to mobile equipments.

CMDA2000 as a family of 3G standards uses Code Division Multiple Access (CDMA) as the multiplexing technique to provide broadband data rate up to 14.7 Mbps [3]. CDMA2000 transmits on one or several pairs of 1.25 MHz radio channels, while UMTS transmits on a pair of 5 MHz-wide radio channels. Unlike UMTS, CDMA2000 is backward compatible with its former generation known as cdmaOne, which was first standardized in 1993.

2.1.1.4 Fourth Generation (4G)

As a successor of 2G and 3G technologies, 4G is the fourth generation of wireless communication standards, the requirements of which have been specified by the International Telecommunications Union (ITU)[4]. 4G technologies are required to support peak bitrate of 100Mbps for high mobility communications and 1Gbps for low mobility communications. 4G provides a flexible channel bandwidth between 5 and 20 MHz. Another requirement of 4G standards is smooth handover within various types of networks and dynamical resource allocation to improve efficiency and user experience.

3GPP Long Term Evolution [25], normally referred as LTE was developed by the 3GPP to provide higher speed and network capacity. LTE Advanced [26] is the current 3GPP proposal for 4G standards and is an enhancement of the LTE standard. It introduces

[3] http://www.qualcomm.com/media/documents/wireless-networks-rev-b-enhanced-mobile-broadband-all,2010.

[4] ITU-http://www.itu.int

multicarrier which enables the use of wide bandwidth for high speed data transmissions. Another significant contribution of LTE Advanced is the wise use of advanced topology networks with deployment of low power nodes.

2.1.2 Wireless Metropolitan Area Network (WMAN)

IEEE 802.16 [18] is a set of standards as basic techniques for Wireless Metropolitan Area Network (WMAN) and is commercialized under the name **Worldwide Interoperability for Microwave Access** (WiMAX). WiMAX was designed to remove line of sight requirements (or passing small obstructions such as buildings etc.) and to provide broadband wireless access up to 50 km for fixed stations and 5-15 km for mobile devices. The common architecture of a WMAN is similar to WLAN which consists of access points as central coordinator and mobile devices. But the main differences lie in the protocol specifications thus the data speed and coverage area of each standard. Two standards including 802.16, formally known as 802.16.1, and 802.16.2 were released with an amendment 802.16c at first, followed by the standardization of 802.16a, 802.16-2004 and 802.16e-2005. 802.16e-2005 was approved in 2005 and was announced as being deployed around the world in 2009. 802.16e uses Scalable OFDMA to carry data, supporting channel bandwidths of between 1.25 MHz and 20 MHz. It supports adaptive modulation at physical layer which means the coding mechanism varies according to the signal condition. Multiple Input Multiple Output (MIMO) which improves communication performance by adding antennas at both transmitter and receiver sides is incorporated in the standard. 802.16k was standardized in 2007 as an amendment to IEEE 802.1D, which is an IEEE MAC Bridges standard. 802.16-2009, with amendment 802.16j and extension 802.16j, is standardized in 2009 and is used as the current IEEE 802.16 standard. 802.16m, also known as Mobile WiMAX Release 2 or WirelessMAN-Advanced aims at fulfilling the ITU-R IMT-Advanced requirements on 4G systems.

Table 2.1 802.16 Family

Standard	**Release Year**	**Description**
802.16.1	2001	Fixed Broadband Wireless Access (1066 GHz)
802.16.2	2001	Recommended practice for coexistence
802.16c	2002	System profiles for 1066 GHz
802.16a	2003	Physical layer and MAC definitions for 211 GHz
802.16.2-2004	2004	Recommended practice for coexistence (Maintenance and rollup of 802.16.2-2001 and P802.16.2a)
802.16e	2005	Mobile Broadband Wireless Access System
802.16k	2007	Bridging of 802.16 (an amendment to IEEE 802.1D)
802.16-2009	2009	Air Interface for Fixed and Mobile Broadband Wireless Access System
802.16j	2009	Multihop relay
802.16h	2010	Improved Coexistence Mechanisms for License-Exempt Operation
802.16m	2011	Advanced Air Interface with data rates of 100 Mbps mobile and 1 Gbps fixed

The basic architecture of WiMAX is shown in Figure 2.3. WiMAX consists of four major components: Internet, Network Service Provider (NSP), Network Access Provider (NAP), and subscribers. The Internet provides Internet content to a user/subscriber and connectivity to a NSP. The main function of NSP is to provide IP connectivity services. The NAP is composed of one or more base stations and Access Service Network (ASN) gateways which connect the NAP with NSP. And the subscribers include all user mobile devices, such as mobile phones, PDAs, laptops, etc.

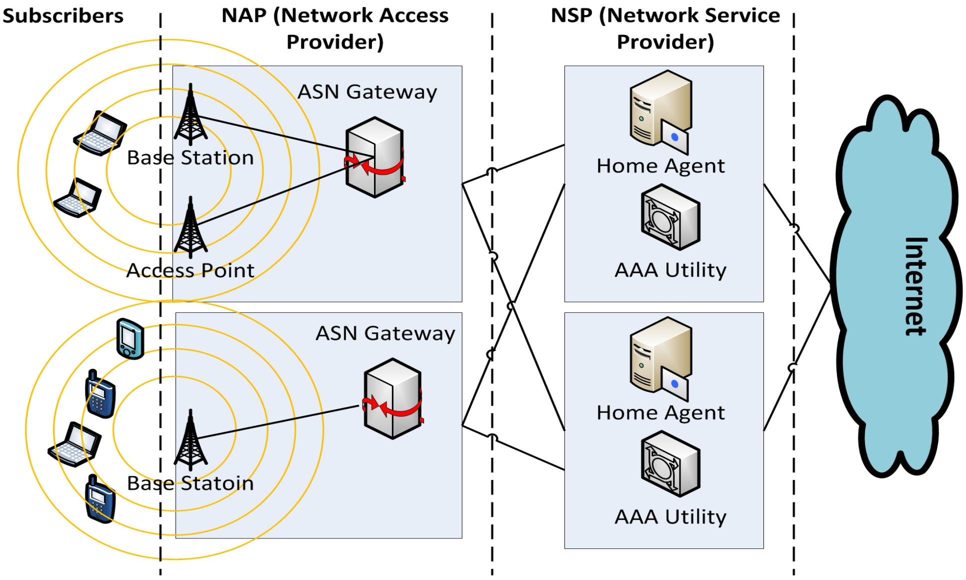


Figure 2.3 WiMAX architecture

2.1.3 Wireless Local Area Networks (WLAN)

WLAN, also known as **Wireless Fidelity** (Wi-Fi), is based on the **IEEE 802.11** [27] family of standards. WLAN is one of the main wireless communication technologies designed for wireless devices. It basically consists of multiple mobile stations including laptops, PDA etc. and an access point which carries data to and from those stations for the purpose of communication, as illustrated in Figure. 2.4.

The standard family has a series of techniques built in the basic protocols. In 1997, the Institute of Electrical and Electronics Engineers (IEEE) created the first standard **802.11-1997** [27] within this framework. The standard only supports a maximum bandwidth of 2 Mbps which is far from enough for most applications and therefore was substituted by other standards very soon. **802.11b** [28], standardized in 1999, is the first widely accepted standard. 802.11b increases network bandwidth to 11 Mbps, which is comparable to Ethernet. 802.11b operates in the unregulated 2.4 GHz ISM band which means all channels overlap and serious interference with other devices using the same frequency such as microwave oven and Bluetooth can occur. To prevent signal interference, a signalling method such as Direct-Sequence Spread Spectrum (DSSS) is employed by 802.11b. On the other hand,

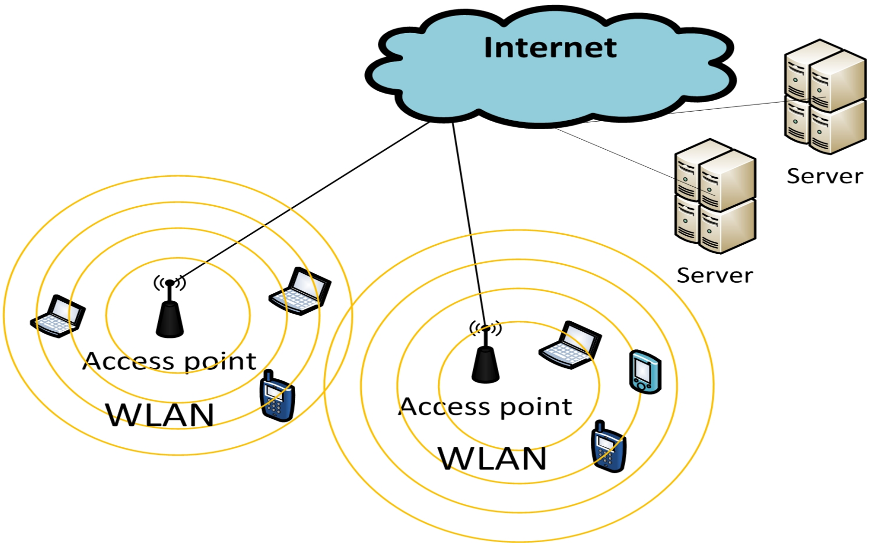


Figure 2.4 WLAN architecture

due to the frequency used, 802.11b is widely used due to low cost in manufacturing and production. At the same time when 802.11b was created, another amendment to 802.11 called **802.11a** [29] was developed by IEEE as well. 802.11a provides up to 54 Mbps of bandwidth and uses a regulated frequency spectrum around 5GHz. Higher frequency indicates shorter coverage area and low ability of penetrating obstructions and more importantly higher costs, and therefore 802.11a is more frequently used in business networks, rather than private deployments. **802.11g** [30] is a standard created in 2003 which is designed to combine the advantages of both 802.11a and 802.11b. It supports high data rate of up to 54 Mbps and operates in the 2.4 frequency to cover wider range. It applies Orthogonal Frequency-Division Multiplexing (OFDM) to solve the problem of signal interference. **802.11n** [31], standardized in 2009, is an amendment to the IEEE 802.11-1997 standard and achieves a significant improvement in data rate over the previous standards. It adds to existing standards Multiple Input Multiple Output (MIMO) which adds antennas at both sides of the communications in order to improve communication performance. Multiple wireless signals are utilized to offer higher signal intensity and greater range. 802.11 also doubles the channel width at the physical layer which enables operation in both 5 GHz and 2.4 GHz band. The range of a Wi-Fi network is normally 30-100m. However, the deployment of 802.11n involves higher costs and some difficulties due to the different technology used. The development of 802.11 standards and general specification of each standard is shown in Table 2.2.

The IEEE 802.11 Task Group ”k” recently developed an amendment to IEEE 802.11 wireless LAN standard, which is referred to as **802.11k** [32]. IEEE 802.11k provides information to the wireless devices so that they can find the best available access point to associate with. It is one of the key industry standards that enables seamless basic service set transition in WLAN. It was designed to improve the way traffic is distributed within a WLAN. In traditional WLAN, devices are always connected to the AP with the strongest signal, which might lead to overloading on one AP. To solve the problem, a wireless device is connected to one of the under-utilized AP if the AP with the strongest signal is loaded full.

IEEE 802.11s [33] is another amendment to IEEE 802.11 standardized in 2011 and was designed for mesh networking. It defines how wireless devices can be connected among themsevels to organize a mesh network, which is often used for static topologies and ad hoc networks. IEEE 802.11s inherently depends on one of the existing 802.11 standards, such as 802.11a, 802.11b, 802.11g or 802.11n, for the purpose of carrying the actual traffic and requires an additional routing protocol.

2.1.4 Wireless Personal Area Network (WPAN)

Wireless Personal Area Network (WPAN) is a wireless network used for short range communications. Multiple types of technologies support WPAN including ZigBee [34], Bluetooth [35] and so on. One of the mostly applied technologies in practice is **Bluetooth** defined by IEEE 802.15 standard. It is a wireless technology designed to provide a universal short-rage data exchange capability with high level of security. The motivation of WPAN Bluetooth is based on the fact that most people spend the majority of the day within 10m of some kind of Internet access port [5]. A typical Bluetooth network consists of a dynamic group of less than 255 devices connected within themselves without any central

[5] Towards 4g: wpan and the person-centered concept. http: //trends-in-telecoms.blogspot.com/2011/04/towards-4g-wpan-and-person-centered_04.html.

Table 2.2 802.11 Family

Standard	Release Year	Description
802.11-1997	1997	First 802.11 standard, very low data rate
802.11b	1999	Pros: low cost, modest signal range. Cons: signal interference, low data rate
802.11a	1999	Pros: fast speed, regulated frequency prevents signal interference. Cons: high cost, short signal range
802.11g	2003	Pros: fast speed, good signal range. Cons: high cost, signal interference
802.11n	2009	Pros: fast speed, best signal range, more resistant to signal. Cons: high cost
802.11k	2008	Improves seamless basic service set transition in WLAN
802.11s	2011	Supports mesh networking

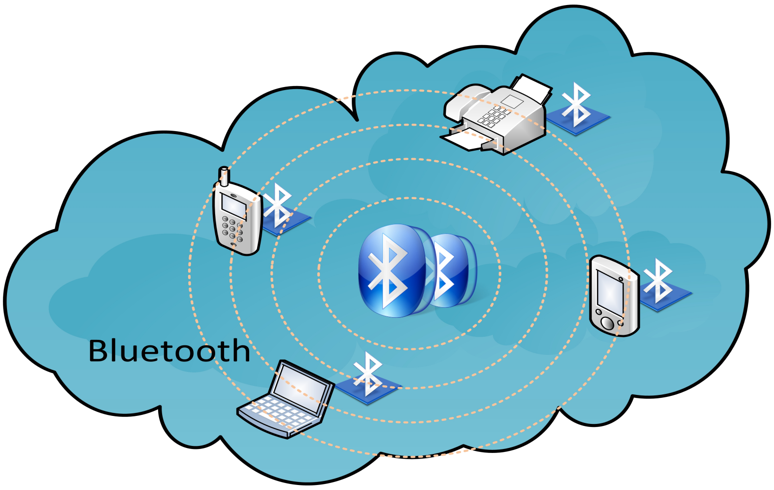


Figure 2.5 Bluetooth architecture

router, as illustrated in Figure 2.5.

Bluetooth [35] uses radio technology frequency-hopping spread spectrum, and data is divided into chunks before transmitted in the range of 2400-2483.5 MHz. The standard allows data exchange between portable and stationary devices which can share up to 720 kbps of capacity within 10 m. Bluetooth, designed as a replacement of wired communications, avails of low power consumption short range data exchange on low-cost transceiver

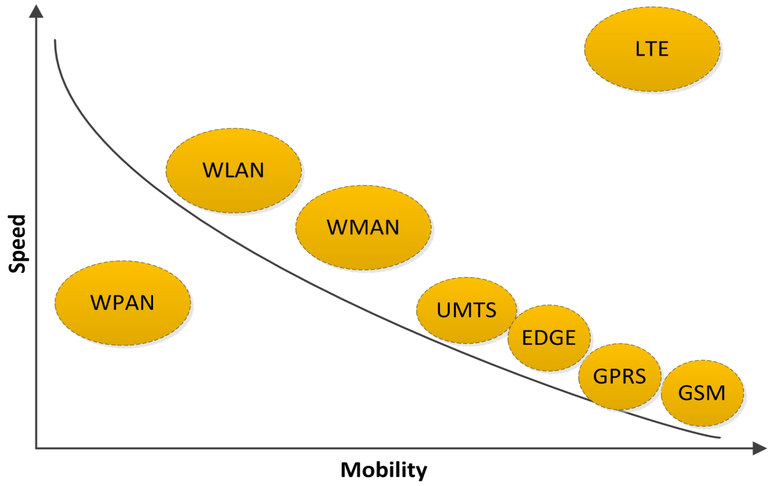


Figure 2.6 Wireless technologies– speed VS. mobility

microchips devices [6].

Zigbee [34], as another widely used technology, provides a defined rate of 250 kbit/s, and it is best suited for periodic or intermittent data or a single signal transmission. It is mostly deployed for short-range wireless data transfer with low data rate, while providing long battery life. Zigbee is a simpler and less expensive solution than other WPANs, such as Bluetooth. The technology is widely deployed in wireless light switches, electrical meters with in-home-displays, traffic management systems etc.

Figure 2.6 provides an overview of the wireless technologies in terms of mobility and speed. It can be seen that traditional wireless cellular networks such as GSM, UMTS are relatively low in speed while providing high mobility, and wireless technologies such as WMAN and WLAN provides much faster data speed but with lower mobility. WPAN provides only stationary or pedestrian mobility and relatively low speed but still achieves popularity through easy deployment. 4G techniques such as LTE, when compared with other solutions, are more advanced in terms of both mobility and data speed at compensation of high cost.

[6] How bluetooth technology works.http://www.mobileinfo.com/Bluetooth/how_works.htm.

2.2 Network Protocol Stack

Network rotocol stack provides an overall view of the implementation of computer networking protocol suite, and simplifies the design and evaluation of each individual protocols in the context of other protocols. Within the stack, protocols are divided into different layers, which only relate to their upper and lower counterparts. Protocol stack uses encapsulation to provide abstraction of protocols and services. Generally speaking, the information is passed down from higher to immediate lower layers for being further encapsulated at each level before being sent out at the sender side, and received data at the receiver side is translated at each corresponding level and forwarded up from the lowest to the highest layers. **TCP/IP** protocol suite consists of four [36] or five layers [37] depending on preferences. The four layer model consists of application layer, transport layer, Internet layer and Link layer, while the five layer model is comprised of application layer, transport layer, network layer, data link layer and physical layer. The structure and main standards at each layer of the latter model are shown in Figure 2.7. The Open Systems Interconnection (OSI) model is another widely used model and divides the internal functions of a communication system into seven layers including physical layer, data link layer, network layer, transport layer, session layer, presentation layer and application layer. Next, different protocols are discussed in details at each layer.

2.2.1 Application Layer Standards

Application layer protocols on the protocol stack reside the closest to client users, and interact directly with the users and are influenced directly by different requirements of users. Protocols on this layer provide host-to-host connections and service requirements of different applications. There are several widely used protocols at this layer which will be discussed in more details.

The Hypertext Transfer Protocol (HTTP) is the basis of data transfer in the World Wide Web. The version in common use is standardized in RFC 2616 [38], and runs on top

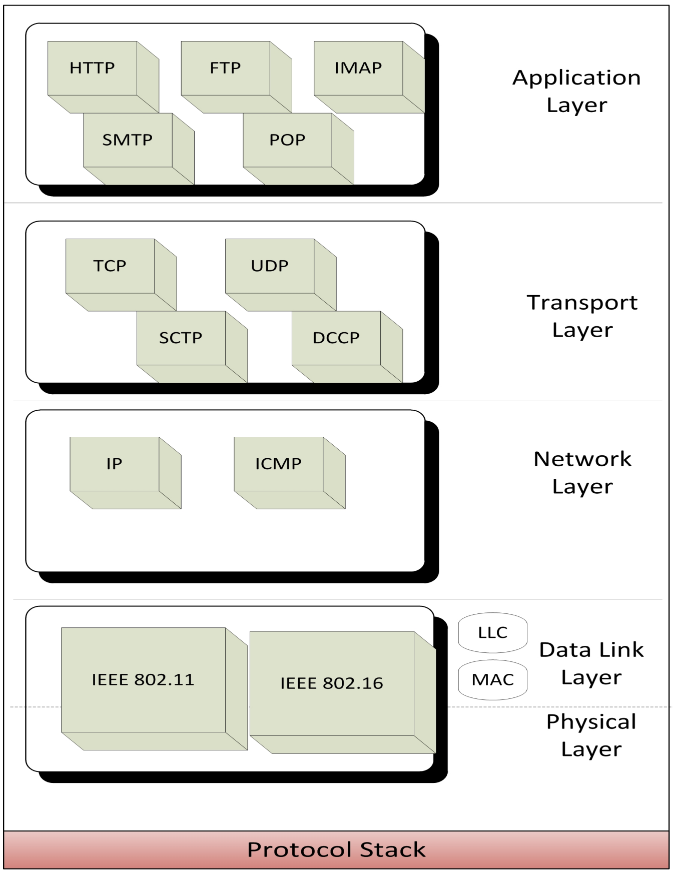


Figure 2.7 TCP/IP protocol suite

of reliable protocols at transport layer such as Transmission Control Protocol (TCP). HTTP mainly defines how the data should be formatted, transferred and the proper way of interaction between request on the client side and response on the server side. Although UDP is used by many streaming protocols, it may sometimes be blocked by the firewall. In this case, HTTP based streaming provides a solution as it works on the top of TCP and requires a simple web server. For this reason, HTTP streaming has one of the largest penetration and in the market HTTP traffic accounts for a large fraction of Internet bandwidth used for streaming.

The File Transfer Protocol (FTP) is another widely deployed protocol used for file transfer. The protocol is standardized in RFC 0959 [39]. It provides supports for separate control and data connections between client and server. The main function of control connection is to perform user authentication and command exchange and remains open when file transfer is being carried on, while the data connection is only active during data transmission.

The Simple Mail Transfer Protocol (SMTP) defined by RFC 2821 [40] is a standard designed for email transmission. The protocol is normally used for outgoing data transmission for example when emails are sending out. On the other hand, incoming data is fetched by the client using either the Post Office Protocol (POP) RFC 1939 [41] or the Internet Message Access Protocol (IMAP) defined by RFC 3501 [42].

Real-Time Transport Protocol (RTP) [43] provides support for end-to-end delivery for real time audio/video data streaming. Applications run RTP on the top of UDP to make proper utilization of its multiplexing and checksum capabilities. RTP does not provide timely or in-order packet delivery or QoS support. The sequence number contained in the RTP packet only allows the receiver to reconstruct the senders sequence number. An RTP session consists of one or more participants, where each of the clients can send or receive media data. A network address and two port numbers are used to identify the participants. One port number is for media data and the other one is for RTP Control Protocol (RTCP). The participants are enabled to choose the media types they are willing to receive. For example, a participant may just want to receive the audio part of a media streaming video only.

RTP Control Protocol (RTCP) [44] aims at maintaining high QoS levels of RTP through providing the feedback information such as packet loss, jitter condition to all the participants. It works along with RTP and does not carry any media content. The feedback information is used to adjust the media transfer rate. Moreover, it can also be used to monitor network conditions and diagnose the problem in data distribution among receivers. Although RTP runs on UDP, TCP is used for RTCP data transmission. Figure 2.8 illustrates an RTP session.

Real Time Streaming Protocol (RTSP) [45] is an application layer protocol designed to control streaming media servers used in entertainment and communications systems. It provides support for establishing and controlling media sessions between end points. The

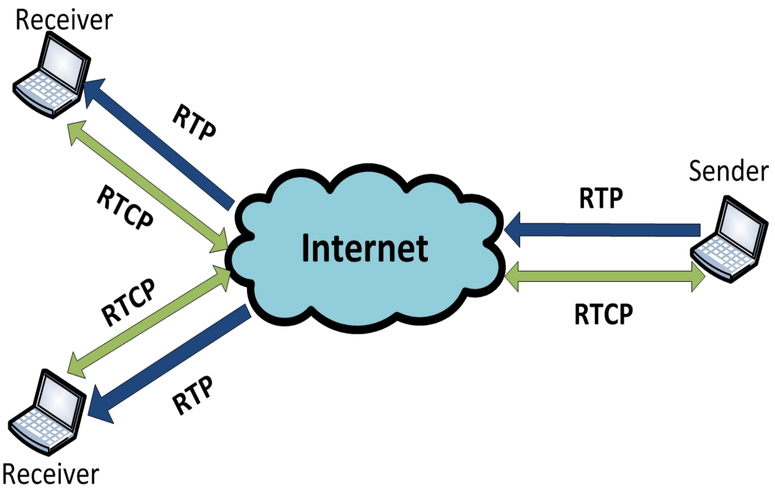


Figure 2.8 Illustration of an RTP session

protocol itself does not provide transmission of streaming data, and most RTSP servers use the RTP and RTCP for data delivery. An RTSP session is not bound to any specific underlying transport layer protocol, and the protocol provides an extended framework to choose media delivery channels such as UDP, multicast UDP and TCP, and RTP based delivery mechanisms. An RTSP session starts by requesting a presentation or media to be started at the server. Server labels each session with an identifier. This session identifier represents the shared state between the server and client and is used in all subsequent controls. If the state is lost, RTSP stops the transmission of media by not receiving RTCP messages while using RTP. Figure 2.9 depicts an RTSP session and illustrates the basic requests used in RTSP.

Session Description Protocol (SDP) [46] is a protocol used to carry media details, transport addresses and other session description metadata to other participants. The protocol is normally used by other streaming protocols such as RTSP to provide the description of a multimedia session for the purpose of session announcement, session invitation or other form of multimedia session initiation. A common SDP session description consists of the name and purpose of the session, the time duration of an active session, the type of media comprising the session, and information like address, port, format, etc. used for data receiving.

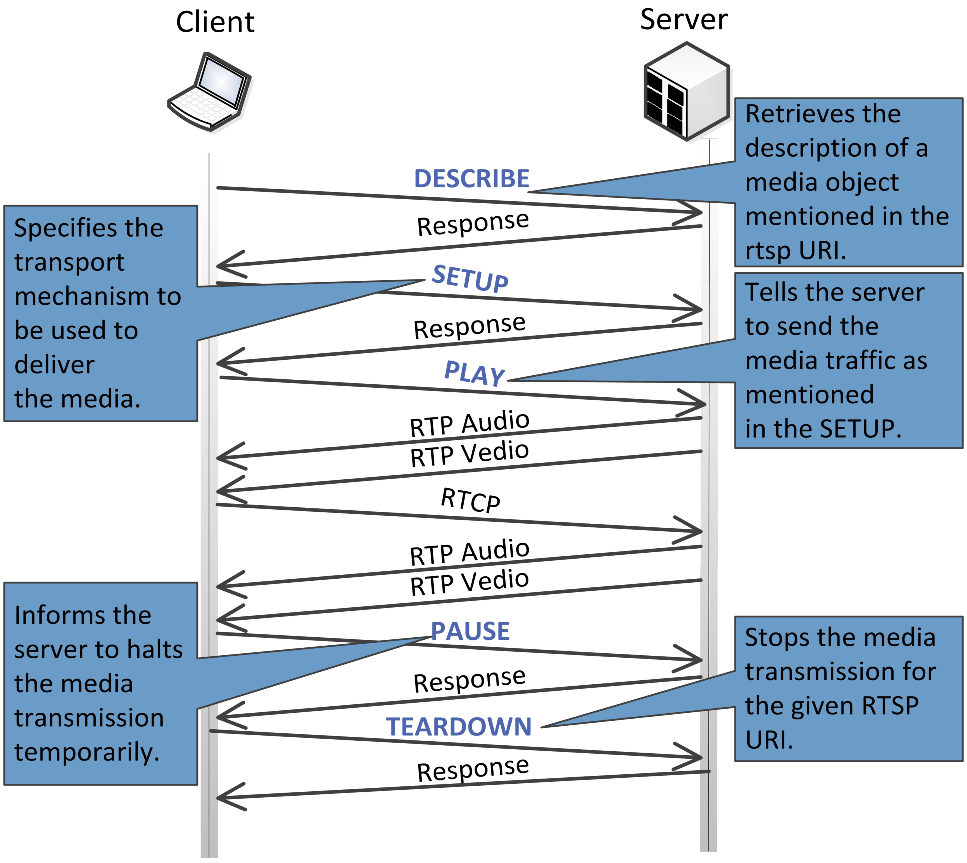

Client
Server
Retrieves the description of a media object mentioned in the rtsp URI.
DESCRIBE
Response
Specifies the transport mechanism to be used to deliver the media.
SETUP
Response
Tells the server to send the media traffic as mentioned in the SETUP.
PLAY
RTP Audio
RTP Vedio
RTCP
RTP Audio
RTP Vedio
Informs the server to halts the media transmission temporarily.
PAUSE
Response
Stops the media transmission for the given RTSP URI.
TEARDOWN
Response

Figure 2.9 Illustration of an RTSP session and basic RTSP requests

Session Initiation Protocol (SIP) [47] is an application layer protocol widely used for controlling multimedia communication sessions. It can be used for either two-party or multi-party sessions involed in audio and video communications. The design of SIP is similar to HTTP, the client of which makes a request through a particular method and receives response from the server. Some of the header fields, encoding rules and status codes of HTTP are re-used by SIP.

Dynamic Adaptive Streaming over HTTP (DASH) [48] is another protocol designed for multimedia transmission. Dash chopps the multimedia content into HTTP-based segments, each of which is made available of various bitrates. The client selects the segment with the highest bitrate possbile in real time to make sure smooth playback is provided without causing stalls or rebuffering events.

2.2.2 Transport Layer Standards

Transport layer protocols provide end-to-end data transmission and optionally provide functions such as congestion avoidance, reliability and flow control. Transport Control Protocol (TCP) [49] and User Datagram Protocol (UDP) [50] are two de-facto protocols employed at transport layer. These two protocols are designed and widely deployed in wired network communication environments. Several studies [51] [52] [53] on the performance evaluation of these two protocols in wireless network environments have shown that there are various performance issues when using them for data transport over wireless communication networks.

TCP [49] is a connection-oriented protocol that provides reliable and in-sequence data transmission. The reliability is achieved by performing necessary control data exchange which involves extra overhead and necessary retransmission if packets are predicted lost. The main problems with using TCP in wireless environments are the energy and performance related, the former is crucial for battery powered wireless portable devices. The control mechanisms adopted in TCP such as handshaking and packet acknowledgements are energy inefficient as the overhead might dominate the traffic in some cases such as in sensor networks where there is not so much data generation. Relating to performance issues, as opposed to wired communications where packets are not acknowledged by recipient within the expected deadline are supposed to be lost due to packet congestion and buffer overflow, packet loss in wireless communications may be caused by interference, noisy channel etc. which does not necessarily imply traffic congestion. The default action of TCP is reducing the window size unnecesarily inefficiently uing the available bandwidth.

UDP [50] is a message-oriented protocol that provides no delivery guarantee to upper layers, and does not provide any support mechanism for congestion detection or reliability control. UDP is not suitable for applications requiring guaranteed reliability such as e-mail or file transfer applications.

Stream Control Transmission Protocol (SCTP) [54] is another protocol deployed lately at transport layer. It provides a mix of services present in both TCP and UDP: it guarantees reliable and in-order transmission of data like TCP and is message-oriented, similar to UDP. SCTP provides a novel feature of multi-homing to increase transmission reliability. Multi-homing enables multiple IP addresses within one association connecting two SCTP hosts. An end-point of SCTP can have several IP addresses and one of them is used as primary address for current data exchange. The other addresses are used for retransmission or potential candidate addresses if the primary one is unreachable or fails. SCTP provides multi-streaming which means multiple streams of data chunks are allowed at the same time to provide better transmission performance.

The **Datagram Congestion Control Protocol** (DCCP) [55] is another message-oriented transport layer protocol which is deployed for multimedia streaming. DCCP does not provide reliable in-sequence delivery of data and is useful for applications with restrict time requirements. Besides the establishment, maintenance and tear-down of packet flow, DCCP is also used for congestion control purpose for UDP based applications.

2.2.3 Network Layer Standards

Network layer is responsible of packet delivery through host addressing and routing. It lies under transport layer and above data link layer in the protocol stack.

The **Internet Protocol** (IP) typically referred to Internet Protocol version 4 (IPv4), which is defined by RFC 791 [56], is the dominant protocol used for data relaying in both the wired and wireless networks. The main function provided is to deliver packets from the source to the destination based on the addresses associated. It mainly defines how the address of hosts works and tries to establish routes of packets through address lookup. It does not guarantee reliability which means every packet is forwarded with best effort and the successful delivery is the responsibility of upper layers. Despite the widely use of IPv4, Internet Protocol version 6 (IPv6) [57] is emerging as a replacement due to the fast growth of the Internet. It mainly provides more addresses and also some additional features such as built-in security and better support for QoS.

Internet Control Message Protocol (ICMP) defined by RFC 792 [58] is another protocol running at network layer and it is a complementary protocol to IP. ICMP is designed to diagnose IP errors and for routing purposes over IP network. In the ICMP segment, the type field combined with the code field specifies the cause of errors, and therefore the protocol is widely used for troubleshooting.

2.2.4 MAC & Physical Layer Standards

Physical layer is the lowest layer of the protocol stack and is responsible for transmitting raw bits over the hardware transmission medium. For wireless communications, physical layer receives and serialises the frame from data link layer and sends it to the corresponding receiver over electromagnetic radio waves. Physical layer protocols mainly take care of definition of hardware specifications which includes the details of operation of each device such as wireless radio transceiver and network interface card, data encoding and signalling, and data transmission and reception.

The MAC (Medium Access Control) layer is a sub-layer of the data link layer, which provides point-to-point data transmission. The main function of MAC protocol is to allocate wireless resources to multiple concurrent network devices connected to the same physical medium. The MAC protocol is responsible for achieving efficient resource usage and provides a certain level of reliability to upper layers. MAC layer interacts with physical layer and controls the wireless network interface card (WNIC) which involves a significant impact on the energy consumption involved by the sleep/wake cycles of the WNIC.

Two main functions of MAC layer protocols are: addressing mechanism and access control mechanism. In the addressing mechanism, there is a unique MAC address which is also called physical address of each network interface. The address combined with upper layer IP address enable packets to reach their destination. To be more specific, when a packet is forwarded to the destination sub-network, the IP address is resolved into the physical address of the destination host for successful delivery.

The channel access control is designed to provide access for connection of wireless devices to the shared physical medium. It is critical as packet collisions may easily occur under wireless conditions if two or multiple wireless hosts transmit data at the same time. Channel access control schemes are categorized according to the way resources are allocated to multiple hosts [59]. The most frequently used channel access methods are Frequency Division Multiple Access (FDMA) which divides and allocates frequency bands to users, Time Division Multiple Access (TDMA) which allocates different time slots to prevent collisions, and Code Division Multiple Access (CDMA) which allows simultaneous transmission by multiplexing.

IEEE 802.11 [27] is an IEEE Standard for both MAC and physical layers and it is the fundamental technology used by Wi-Fi. It supports both contention based medium access scheme and contention free scheme.

The contention free scheme is Point Coordination Function (PCF) built on infrastructure based scenarios. In this scenario an Access Point (AP) functions as a central node, which receives and forwards all packets sent within its service set. Packet collision is avoided due to the coordination of the AP. It has a built-in Power Saving Mode (PSM) and is the basis of most MAC layer power saving schemes. In PSM, time is divided into beacon intervals, on a regular base of which the access point broadcasts beacon frames. One component of the beacon announcement is the maximum duration of the contention-free period, CFPMaxDuration. All the associated stations set the Network Allocation Vector (NAV) timer to the maximum duration to lock out Distributed Coordination Function (DCF) based access to the wireless medium. On the mobile nodes side, the standard enables network card to sleep for a fixed duration, i.e. one or several round of beacon intervals, and wakes up listening to beacons. Once the node finds there are packets addressed to it, rounds of polling would take place. For each polling process, one buffered data will be transferred to and acknowledged by the receiver. The whole process ends when all buffered data is retrieved by the wireless node. On the other side, AP buffers packets for sleeping nodes, and notifies those nodes by beaconing regularly. Figure 2.10 illustrates the buffered data retrieval process.

The contention based scheme is also known as DCF and employs Carrier Sense Mul-

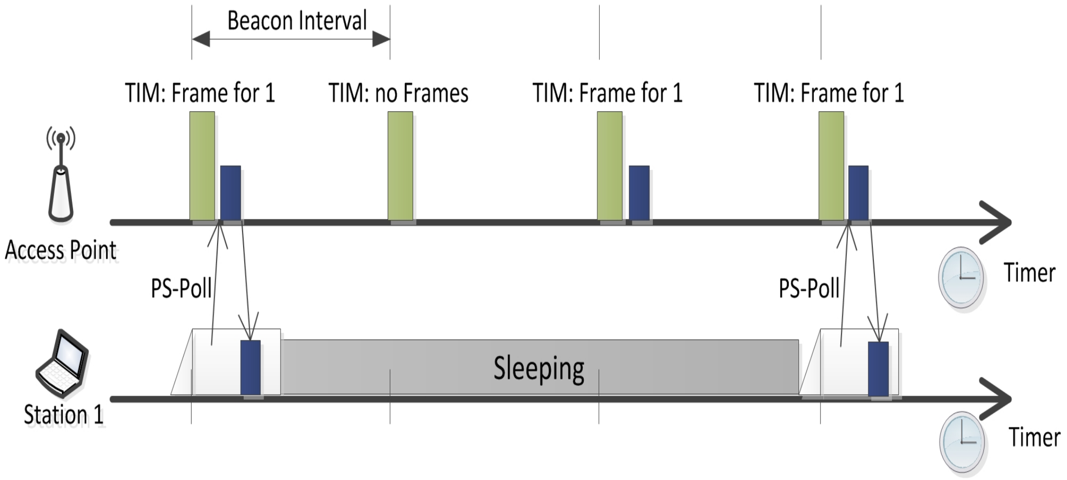


Figure 2.10 Buffered frame retrieval process in PCF

tiple Access with Collision Avoidance (CSMA/CA) to avoid packet collision. CSMA/CA is a contention avoidance scheme which can be deployed in IEEE 802.11-based protocols. Nodes wishing to send packets wake up and listen to the channel for a DCF Interframe Space (DIFS) interval. If the medium is free, transmission will start immediately; otherwise the node will back off and the station defers its transmission. In a network where a number of stations contend for the wireless medium, if multiple stations sense the channel busy and defer their access, they will also virtually simultaneously find that the channel is released and then try to seize the channel. As a result, collision occurs. In order to avoid such collisions, DCF also specifies random backoff, which forces a station to defer its access to the channel for an extra period. The length of the backoff period is determined by the following equation:

$$BackoffTime = random() * aSlotTime \tag{2.1}$$

This process repeats until packets are transmitted successfully. DCF also employs an optional virtual carrier sense mechanism that exchanges short Request-To-Send (RTS) and Clear-To-Send (CTS) frames between source and destination stations during the intervals between the data frame transmissions. The RTS packet is transmitted after the node senses an idle network for the duration of DIFS, and the receiver will reply with a CTS packet af-

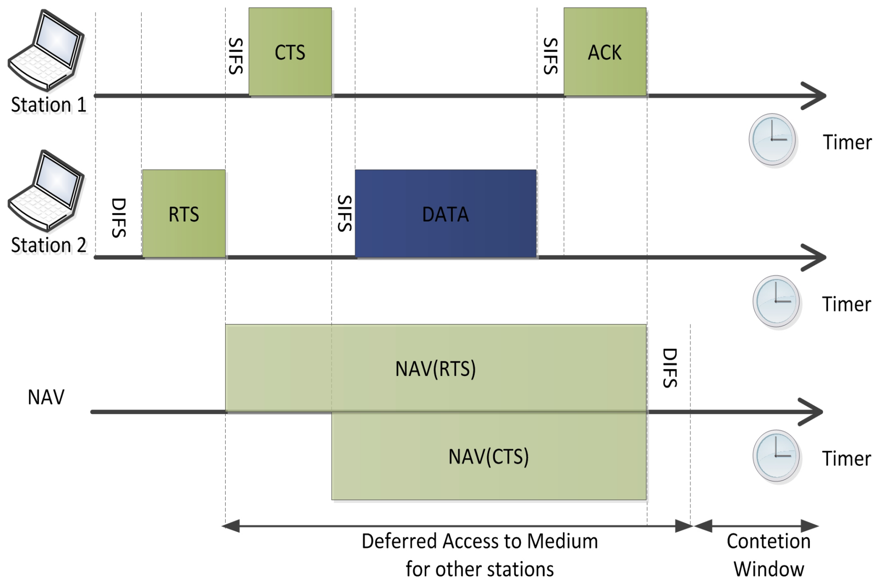


Figure 2.11 Data transmission process in DCF

ter a Short Inter-Frame Space (SIFS). Data packets are only transmitted after the exchange of the pair of packets. Virtual carrier-sensing is provided by the NAV. Most 802.11 frames carry a duration field, which can be used to reserve the medium for a fixed time period. The NAV is a timer that indicates the amount of time the medium will be reserved, in microseconds. The station which is about to transmit data sets the NAV to the time for which they expect to use the medium, so that other stations can count down from the NAV to 0. After this period, the medium should be idle. Power saving under these circumstances can be achieved through preventing packet collision, and turning off the radio while packets transmission is happening among other nodes. Figure 2.11 illustrates the event sequence of DCF.

IEEE 802.16 which is widely known as the supporting technology of WiMAX standardizes air interface and related functions for wireless broadband. The standard divides MAC layer into three sub-layers including MAC security sub-layer which provides data encryption and MAC address based authentication, MAC common part sub-layer which provides medium access, Quality of Service (QoS) etc, and MAC convergence sub-layer which provides interface to various upper layer protocols such as IP and Ethernet. MAC in IEEE 802.16 is a connection-oriented solution with a Base Station (BS) allocating both uplink and downlink bandwidth. The connection-oriented feature enables strong support of quality of service as different bandwidth allocation to individual sessions is allowed and the connections are unidirectional.

Five sets of services are supported with various specifications on QoS parameters: *Unsolicited Grant Service (UGS)*, *real-time Polling Service (rtPS)*, *non-real-time Polling Service (nrtPS)*, *Best Effort (BE)* and *extended real-time Polling Service (ertPS)*. Each of these scheduling services has a mandatory set of QoS parameters that must be included in the service flow definition when the scheduling service is enabled for a service flow, as shown in Table 2.3. UGS is designed to support real time data streams consisting of fixed-size data packets issued at periodic intervals. It is used in applications such as Voice over IP (VoIP) without silence suppression. The ertPS service is added by the 802.16e amendment. It is built on the efficiency of both UGS and rtPS. Similar to UGS, the base stations provides unicast grants in an unsolicited manner, but variable-sized data is allowed. The rtPS type is designed to support real-time data streams consisting of variable-sized data packets that are issued at periodic intervals. The service can be used in Moving Pictures Experts Group (MPEG) video transmission and real time video delivery. It introduces more request overheads than UGS, but supports variable grant sizes for optimum real-time data transport efficiency. The nrtPS scheduling service aims at supporting delay-tolerant data streams consisting of variable-size data packets for which a minimum data rate is required. NrtPS can be employed for applications such as FTP transmissions. The BE service is designed to support data streams for which no minimum service guarantees are required and therefore may be handled on a best available basis.

Power saving in IEEE 802.16 is achieved through the employment of sleep or idle modes of mobile stations (MS). In idle mode, the mobile station is not registered with any base station, but it receives downlink traffic through paging. The sleep mode consists of three classes supporting different quality of service levels. In *power save class one* the sleeping window of mobile station increases exponentially to achieve maximum energy saving, whereas in *power save class two* the sleeping window size is fixed. The third power save class is a one-time scheme which means the transceiver of mobile device sleeps for a

Table 2.3 WiMAX Services and QoS Requirements

Service	QoS Specifications	Application
UGS	• Minimum reserved rate • Maximum sustained rate • Request/transmission policy • Tolerated jitter • Maximum latency tolerance	VoIP
ertPS	• Minimum reserved rate • Maximum sustained rate • Request/transmission policy • Tolerated jitter • Traffic priority • Maximum latency tolerance	Voice over IP without silence suppression
rtPS	• Minimum reserved rate • Maximum sustained rate • Request/transmission policy • Maximum latency tolerance	Streaming audio or video; Tele medicine; E-learning
nrtPS	• Minimum reserved rate • Maximum sustained rate • Request/transmission policy • Traffic priority	FTP, document sharing
BE	• Minimum reserved rate • Request/transmission policy • Traffic priority	E-mail

predefined period and then returns to the normal mode.

2.3 Quality Measurements in Wireless Communications

2.3.1 Overview

Two important concepts are widely studied in the measurement of data transmission over wireless networks: Quality of Service (QoS) and Quality of Experience (QoE).

Quality of Service (QoS) was first defined in [60] and refers to the ability to guarantee the quality of telephony communications. The definition mainly consists of 6 components: Support, Operability, Accessibility, Retainability, Integrity and Security. Generally, QoS comprises requirements on all the aspects of a connection, such as service response time, loss, signal-to-noise ratio, cross-talk, echo, interrupts, frequency response, loudness levels, and so on. With the fast development of computer network technology, the topic has been widely studied for measuring data delivery performance. In the field of computer networking, the term in general refers to the ability to provide different priority to different applications, users, or data flows, or to guarantee a certain level of performance to a data flow.

On the other hand, Quality of Experience (QoE) study is a fast emerging multidisciplinary field based on social psychology, cognitive science, economics, and engineering science, focused on understanding overall human quality requirements. It is a subjective method of measuring the client's experience with a service (TV broadcast, call to a Call Center, web browsing, phone call etc). While QoS reflects the capability of a network to provide certain level of quality to a service, which is most of the time not related to customer, but to media, QoE provides a measurement of the user's perception of the service provided.

The main difference between QoS and QoE is depicted in Figure. 2.12.

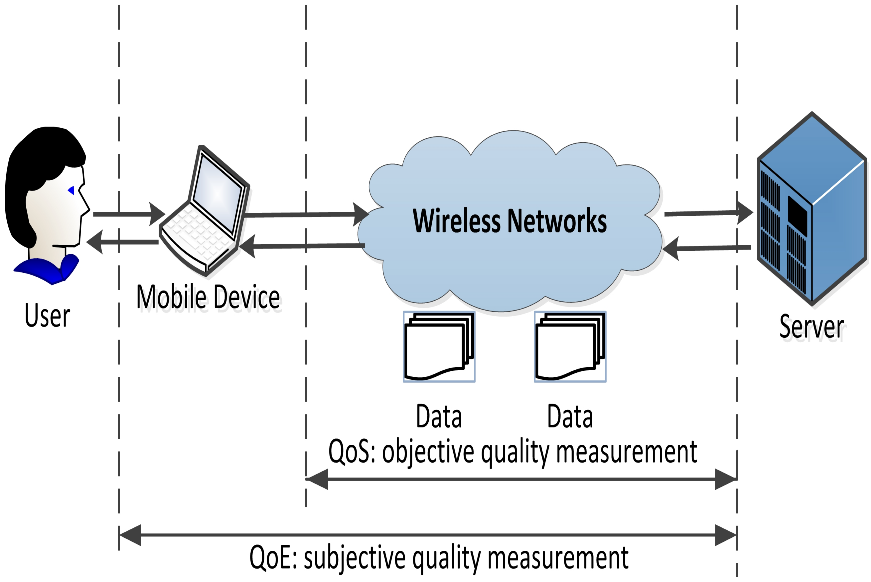

User
Mobile Device
Wireless Networks
Server
Data
Data
QoS: objective quality measurement
QoE: subjective quality measurement

Figure 2.12 QoS vs. QoE in wireless communications

2.3.2 QoS Evaluation Metrics

QoS, in general, is related to the underlying data transport network and measures network-related parameters. Several metrics have been developed for the evaluation of QoS, including packet delay, loss, jitter, etc.

2.3.2.1 End-to-end Delay

End-to-end delay, as defined in [61], refers to the time taken for a packet to be transmitted across a network from source to destination. Delay in wireless networks compromises two primary parts: end-point delay and network delay.

End-point delay refers to the time taken before a piece of data is pushed to the network at the end points. It is mainly caused by the processing of information, such as data encoding and decoding which are used in multimedia streaming applications, or sample analysis process employed in Voice over IP (VoIP) applications.

Network delay specifies how long it takes for a piece of data to travel across the network from the time the first bit is pushed into the network to the time the last bit is received by the destination. It is divided into several parts:

- **Processing delay** is the time taken to process a packet at intermediate nodes within a network. For example, routers may check for bit-level errors in the packet or look up the routing table to determine where the packet's next destination is.
- **Queuing delay** is the time a packet spends in the queues, waiting to be transmitted. It is most often used in reference to routers, which can only process one packet at a time. If packets arrive faster than the router's processing speed, they are put into the queue until being executed by the router.
- **Transmission delay** refers to the time it takes to push the all bits of a packet onto the link. D_T is calculated as the total number of bits in a packet divided by the transmission rate, as in equation (2.2), where N is the number of bits, and R is the rate of transmission.

$$D_T = N/R \tag{2.2}$$

- **Propagation delay** is the time required to deliver a packet over a medium. It is computed as the ratio between the link length L and the propagation speed, S, over the specific medium, as shown in equation (2.3). Propagation delay varies for different medium. For instance, the value reaches 10 Gbps in fibre while in copper up to 15 Mbps of speed is obtained.

$$D_P = L/S \tag{2.3}$$

2.3.2.2 Packet Loss Rate

Packet loss rate, as specified in [62], is defined as the number of dropped packets divided by total number of packets transmitted by the source. The main reasons leading to packet loss are:

- Network device failure which leads to unreachable routes.

- Fading effect which is a characteristic of wireless data transmission. Signal strength experiences degradation over the network medium which could result in packet loss.
- Interference among devices sharing the same frequency as an access point could lead to packet loss and retransmission, which seriously affects user experience.
- Unmatched transmission speed, which means the outer link has a slower transmission speed than the inner link in a node or access point. In this case, when packets arrive at the node, they cannot be forwarded without being delayed at the queue. When the available buffer space is fully filled, upcoming packets will be dropped.
- Traffic fluctuation which may cause aggregation. Even if the outgoing bitrate is similar or the same as the incoming bitrate of a node, packets might get dropped at the queue when traffic from different sources all arrive at one time and there is no enough room for all packets. Besides, data processing takes time and might become the bottleneck in this scenario.

2.3.2.3 Jitter

Jitter is the undesired deviation from true periodicity of an assumed periodic signal in electronics and telecommunications. In computer networks, jitter, also known as Packet Delay Variation (PDV), often refers to the variability over time of the packet latency across a network. It is mostly often measured as the difference of delay between successive packets. The term is defined in [63] and can be caused by some primary reasons:

- **Variation in packet scheduling time**. For example, a multimedia transmission process has to contend for CPU time with other processes and hence there may be some transmit time jitter introduced by scheduling.
- **Network congestion**. Congestion in networks can hardly be predicted and this leads to varied delay in packet delivery.
- **External load sharing**. In order to improve resilience and provide more even net-

work load, the traffic is sometimes routed over multiple routes. Jitter is introduced if the delays across different routes vary significantly.

- **Internal load sharing**. This approach is employed by some routers to provide a multi-processing approach in which packets are processed by multiple parallel queues. The short term differences in queue size could result in low levels of jitter.

Despite the major aspects, other factors such as routing table updates and time drifting could also lead to serious jitter.

2.3.2.4 QoS for Multimedia Services

Although the metrics such as packet delay, jitter, and loss are commonly used for various types of applications, the requirements asked by each application service varies significantly. For example, applications such as email and file transfer allow longer delays and do not care about jitter levels much, while multimedia streaming services are much more sensitive to packet delay or jitter.

Table 2.4 lists the network performance objectives for IP-based applications, as suggested in [64]. QoS levels are categorized into 6 classes, with class 0 representing the best quality while class 5 indicating the worst. Class 0 and 1 can be used for real time data tranmission such as video coferencing and VoIP, while the other classes are more suitable for transaction data transmission, such as file transfer over the networks.

2.3.3 QoE Evaluation Metrics

In the area of multimedia applications, the measurement of QoE is mostly performed at the end devices and can conceptually be seen as the remaining quality after the distortion introduced during the preparation and delivery of content through the network until it reaches the decoder at the end device. Although QoE is subjective, it can also be estimated using the results of objective evaluation, by combining weighted QoS parameters, such

Table 2.4 Network Performance Objectives for IP-based Applications

QoS Class	Delay	Jitter	Loss	Application
0	100ms	50ms	1*10^{-3}	Real-time, jitter sensitive, high interaction
1	400ms	50ms	1*10^{-3}	Real-time, jitter sensitive, interactive
2	100ms	NA	1*10^{-3}	Transaction data, highly interactive
3	400ms	NA	1*10^{-3}	Transaction data, interactive
4	1S	NA	1*10^{-3}	Low loss only (short transactions, bulk data, video streaming)
5	NA	NA	NA	Traditional applications of default IP networks

as delay, jitter and loss. Subjective quality evaluation requires large amounts of human resources, being also time-consuming, while objective methods provide faster evaluation results, but sometimes require large amount of machine resources and sophisticated apparatus configurations.

2.3.3.1 Classificatin of Objective Methods

Audio and video quality test algorithms are divided into several categories depending on what information is made available to an algorithm:

- A Full Reference (FR) algorithm. RF has access to the original reference signal for a comparison. Specifically, it can compare each sample of the reference signal taken from the sender side to the corresponding sample of the degraded signal taken at the receiver side. This algorithm provides highest accuracy and repeatability, but can only be applied for dedicated tests in live networks.
- A Reduced Reference (RR) algorithm. RR uses a reduced side channel between the sender and the receiver which is not capable of transmitting the full reference signal. In order to predict the quality at the receiver side, parameters extracted from the sender side are utilized. The solution provides reduced accuracy and represents a

working compromise when there is limited bandwidth for the reference signal.

- A No Reference (NR) algorithm. NR does not have access to the original reference signal thus can only use the degraded signal for the quality estimation. A common variant of NR algorithms does not work on the decoded audio signal but analyzes the digital bit stream on an IP packet level.

2.3.3.2 E-Model

E-model [65] is a measuring tool which provides prediction of the voice quality. It has been standardized in the recommendation ITU-T G.107. The output of an E-model calculation is a single scalar, called an R-factor, which is derived from delay and equipment impairment factors.

The basic formula for the E-Model is shown in equation (2.4), where:

- R represents the overall network quality rating which is normalized to a value ranging between 0 and 100,
- R_o is the signal to noise ratio,
- I_s is the impairments simultaneous to voice signal transmission,
- I_d refers to the impairments delayed after voice signal transmission,
- I_e indicates the effects of equipment, representing impairments from low-bit codecs and packet loss,
- A represents the advantage factor which attempts to account for caller expectations. For example using mobile phone gives user mobility advantages which offset the impairment caused by errors in the radio interface.

$$R = R_o - I_s - I_d - I_e + A \tag{2.4}$$

Table 2.5 Mean Opinion Score(MOS) Rating Scheme

MOS	Quality	Impairment
1	Bad	Very annoying distortion which is objectionable
2	Poor	Annoying distortion but not objectionable
3	Fair	Perceptible distortion that is slightly annoying
4	Good	Slight perceptible level of distortion but not annoying
5	Excellent	Imperceptible level of distortion

In simple terms, R Factor is calculated by estimating the signal to noise ratio of a connection (R_o) and subtracting the network impairments (I_s, I_d, I_e) that in turn are offset by any expectations of quality had by the caller. Therefore if theres no network and no equipment, the quality is perfect, which leads to equation (2.5).

$$R = R_o \tag{2.5}$$

2.3.3.3 MOS

Mean Opinion Score (MOS) [66], which has been used for decades in telephony, is a test used to obtain the human user's view of the quality of the network. In MOS, listeners are asked to sit in a ”quiet room” and score call quality as they perceived it. IT is required that the room noise level must be below 30 dB with no dominant peaks in the spectrum. There will be both male and female speakers read the tested sentences, and a number of listeners will rate the audio quality heard over the communication medium being tested. The listener will give each sentence a rating using the rating scheme listed in Table 2.5.

The final MOS result is the arithmetic mean of all the individual scores, and ranges from 1 (worst) to 5 (best). Similar approaches can be used to evaluate subjective video quality.

Table 2.6 PSNR Mapping to MOS

MOS	PSNR
5 (excellent)	≥ 37
4 (good)	≥ 31 **&** < 37
3 (fair)	≥ 25 **&** < 31
2 (poor)	≥ 20 **&** < 25
1 (bad)	< 20

2.3.3.4 PSNR

Peak Signal-to-Noise Ratio (PSNR) [67] is a tool mostly used for measuring the quality of reconstruction of lossy compression codecs. It is calculated as the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. The signal refers to the original data being transmitted, while the corrupting noise is the error introduced by compression. PSNR approximates the perceived quality by the users and higher PSNR is an indication of better perceived quality in most cases.

Scaling from 0 to 100, higher PSNR values normally indicate better quality. The 5-point MOS scale and the PSNR mapping [68] are presented in Table 2.6.

2.3.3.5 PSQM & PESQ

Perceptual Speech Quality Measurement (PSQM), defined by the ITU in the standard P.861 [69], objectively evaluates and quantifies voice quality of voice-band (300 - 3400 Hz) speech codecs. It is a more objective voice quality measurement, compared with MOS. The solution can be used to rank the performance of these speech codecs with differing speech input levels, talkers, bit rates and transcodings. PSQM uses a scale of 0 to 6.5 in its rating scheme. It was withdrawn and replaced by Perceptual Evaluation of Speech Quality (PESQ) [70] which contains an improved speech assessment algorithm.

PESQ, as a full-reference algorithm, has become a worldwide applied industry standard for objective voice quality testing used by phone manufacturers, network equipment vendors and telecom operators. The results of PESQ principally model MOS which cover a scale from 1 (bad) to 5 (excellent).

2.3.3.6 PEAQ

Perceptual Evaluation of Audio Quality (PEAQ) [71] is a objective algorithm for the measurement of perceived audio quality. It was originally released as ITU-R Recommendation BS.1387 in 1998 and last updated in 2001. Software is used to simulate perceptual properties of the human ear and multiple Model Output Variables (MOV)are integrated into a single metric. PEAQ results principally model MOS which covers from 1 to 5.

2.3.3.7 PEVQ

Perceptual Evaluation of Video Quality (PEVQ) [72] is a standardized End-to-End (E2E) algorithm used for the measurement of the picture quality of a video presentation. The measurement paradigm is to assess degradations of a decoded video sequence output from the network (for example as received by a TV set top box) in comparison to the original reference picture (broadcast from the studio). It is based on modelling the behaviour of the human visual tract.

PEVQ is a full-reference algorithm and makes use of both the frames of reference and test signals. Similar to PESQ, MOS scores are provided as a measurement of the video quality for IPTV, streaming video, mobile TV and video telephony.

2.3.4 Quality Measurements Conclusion

Quality measurements of data transmission over wireless networks are mainly divided into QoS and QoE metrics: the former one focuses on the data delivery performance over the network and mainly measures the objective network-related metrics while the latter term often relates to the subjective user's perception of the service provided and is performed at the end devices.

Although either QoS or QoE can be used in the measurement of quality levels, there is a need to combine both objective and subjective meansuremnts in order to fully test the delivery quality. Therefore both QoS metrics including packet loss, jitter and delay and QoE metrics such as PSNR and R-factor are evaluated in the assessment of our proposed solutions presented in this work.

2.4 Summary

This chapter introduces some of the most important wireless technologies as part of the evolution in wireless communications. The generations that emerged in the cellular networks are presented with both their benefits and limitations. Wireless networks are classified based on range and discussed seperately, including WWAN, WMAN, WLAN and WPAN. The major technologies behind these technologies are then compared and illustrated in terms of coverage area, speed and mobility.

The network protocol stack is then introduced and all layers are further discussed in details. Protocols that have been standardized are categorized based on the layer they belong to. Energy efficiency at each layer is then presented. Each of the standards falls short in some aspects of energy efficiency or delivery performance, which makes the cross-layer solution presented in this thesis all the more important.

The last section of this chapter introduces the metrics used for quality of service measurement. The most common metrics such as packet delay, loss, jitter are introduced separately. The corresponding factors that lead to the phenomena are also listed. Following that, other tools used for QoS and QoE evaluation are explained in details.

Chapter 3

Related Works

This chapter presents energy efficient research works for each layer of the network protocol stack. Application layer solutions are divided into categories which include data compression, adaptive control and partial caching. Transport layer solutions are categorized into congestion control, reliability control and the combination of these two. Energy saving schemes at network layer include energy aware routing and QoS-oriented energy efficient routing. State-of-the-art techniques at MAC are also presented and analyzed. Cross-layer schemes are summarized and categorized into several categories based on the techniques they rely on including traffic shaping, packet prioritization and so on. The main challenges and open issues that need to be addressed are outlined at the end.

3.1 Energy Efficient Application Layer

Application layer solutions address host-to-host connectivity and service provisioning for different applications. Energy saving at this layer is more application specific and varies with different requirements, especially QoS constraints. For example, web traffic is more delay tolerant, while multimedia applications require real time communication service and tolerate a certain level of packet loss. Application level energy conservation techniques

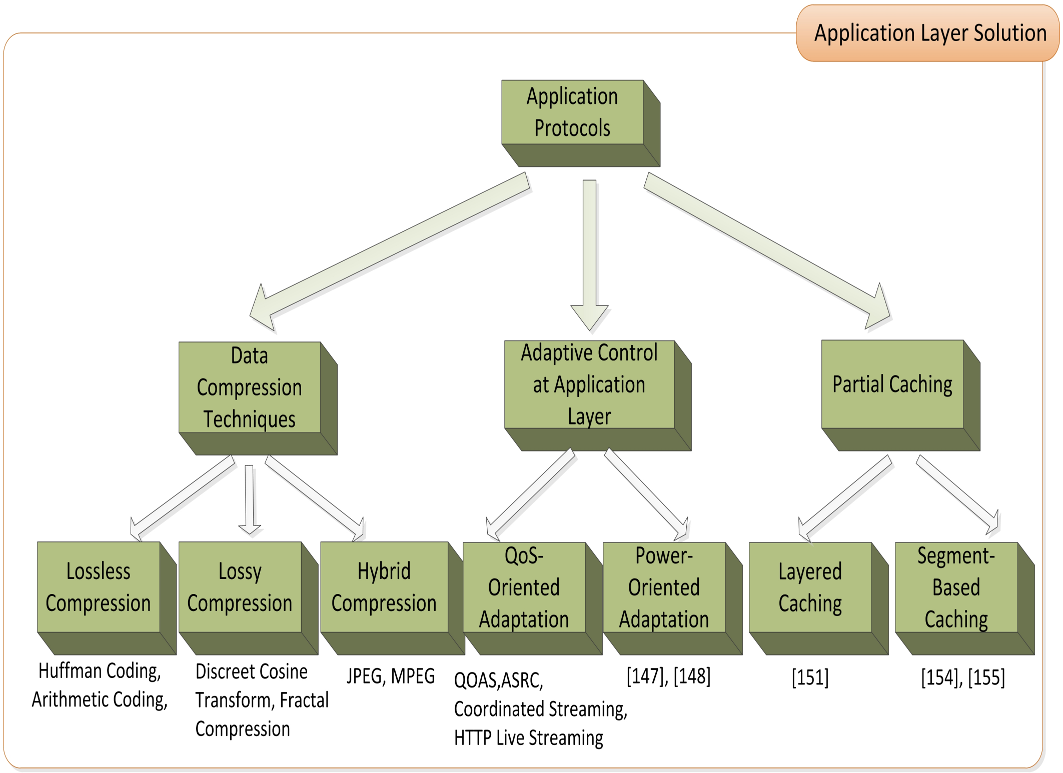

Application Layer Solution
Application Protocols
Data Compression Techniques
Adaptive Control at Application Layer
Partial Caching
Lossless Compression
Lossy Compression
Hybrid Compression
QoS-Oriented Adaptation
Power-Oriented Adaptation
Layered Caching
Segment-Based Caching
Huffman Coding, Arithmetic Coding,
Discreet Cosine Transform, Fractal Compression
JPEG, MPEG
QOAS,ASRC, Coordinated Streaming, HTTP Live Streaming
[147], [148]
[151]
[154], [155]

Figure 3.1 Classification of application layer energy efficient protocols

are categorized based on several aspects including data compression, adaptive control at application layer, data caching, and load partitioning. The first approach compresses data to decrease its size for transmission and storage efficiency. The second one adapts the behaviour of applications at runtime according to QoS or energy levels. Data caching schemes store a part of potentially demanded data in order to increase response time. The key idea of load partitioning is to migrate the computational pressure from client device to server. This is extensively studied in various research works including [73] [74] [75]. However, load partitioning is not widely used by multimedia streaming applications as they do not necessarily require intensive computation at the client side. The classification of application level QoS and energy control for multimedia streaming is depicted in Figure 3.1.

Table 3.1 presents an overview of the energy efficient solutions for networked applications discussed in this section.

Table 3.1 Energy Efficient Solutions at Application Layer

Protocol	Type	Description
Huffman coding [76]	Lossless compression	Frequently used characters are represented by fewer bits and other characters are encoded into longer bit sequences
Arithmetic coding [77]]	Lossless compression	Represent an entire message using a single number
DCT [78]	Lossy compression	A block of data is represented as a series of cosine waves
Fractal compression [79]	Lossy compression	Use a mathematical transformation for data manipulation
Delta modulation [80]	Lossy compression	Encode only the difference between consecutive data sets
CELP [81]	Lossy compression	Divide large vectors into groups and represent each group by a codeword
JPEG [82] [83]	Lossless and lossy compression	Enable selection of either high quality picture or low quality picture
Energy-aware MPEG-4 FGS [84] [85]	QoS-oriented adaptation	Utilize the difference between frames in the process of compression
QOAS [86]	QoS-oriented adaptation	Uses feedback from clients to dynamically adjust streaming behaviour at the server side.
ASRC [9]	QoS-oriented adaptation	Utilize ARQ and guarantee that data get to the destination within the delay bound.
Coordinated Streaming [87]	QoS-oriented adaptation	Use a pair of lower bound and upper bound values of the playout buffer to adjust the data rate
HTTP Live Streaming [88]	QoS-oriented adaptation	Chops the whole stream into a sequence of small HTTP-based file downloads
Energy-aware video streaming [89]	Power-aware adaptation	Adjusts video streaming strategy at runtime to prolong service time of a whole communication system
Energy-aware adaptation [90]	Power-aware adaptation	Adapt the behaviour of applications dynamically according to the energy level
Multimedia proxy caching mechanism [91]	Layered-caching	Each stream is distributed into different layers with priorities
Proxy prefix caching [92]	Segment-based caching	Store a large number of initial frames in the buffer and retrieve the remaining frames when a streaming request is made
Proxy caching mechanism [93]	Segment-based caching	Consider the video quality and the popularity of multimedia content when deciding whether to replace a video or to cache it.
Segment-based proxy caching [94]	Segment-based caching	The video stream is divided into variable-sized segments which are assigned different priorities in caching and replacement

3.1.1 Data Compression Techniques

There are many studies targeting data compression techniques required mostly due to limited network resources. Their main goal is to compress large size data content into smaller size messages, enabling the receivers to recover either the original content or similar content with minor losses. General compression techniques such as those presented in [95] [96] [97] compress data by applying reversible transformation, and other studies focus on specific type of content such as text and database queries [98] [99] [100]. Among all applications, multimedia streaming has the highest requirements for the compression techniques due to:

- Scarce power resources. Energy efficiency is critical for battery operated wireless devices, and the time and energy consumed on data transmission and processing needs to be controlled low.
- Large quantity of data required by images, audio and video content. In contrast to traditional text based applications, multimedia content is associated with large amounts of data.
- Relatively low bandwidth available in wireless network environments. Although there is ongoing progress in the development of high bandwidth wireless technologies, the increasing growth in wireless users and data traffic is much faster than the infrastructure and delivery protocols development.
- High QoS requirements. Multimedia applications especially audio and video streaming are more sensitive to delay and packet loss, which put higher requirements on compression techniques. This is because users expect higher quality of service.

Multimedia compression mechanisms are used to reduce the amount of data at the sender side; the original data is recovered through decompression at the receiver side. The energy cost related to the whole transmission process includes the energy consumed with compression/decompression and communication process. Although energy spent with communication is decreased for the wireless network interface (WNIC) due to the smaller data size and shorter duration for data reception, (i.e. the WNIC stays in active mode when receiving data) the technique introduces extra computational overhead to the processor, especially in the decompression process at the recipient side, and therefore it is crucial to balance these two aspects in the design of compression mechanisms.

Lossless compression techniques [76] [77] [96] [101] allow the receiver to recover the identical content transmitted from the sender, which leads to larger size of data after compression. The general mechanism is to map characters into binary codes through the use of a mapping table. **Huffman coding** [76] is one of the widely used coding technique, which uses a Huffman table at both sides of communication for encoding and decoding, and the variable length table is derived according to a prediction on the probability of occurrence of each value of the source. The most frequently used characters are represented by fewer bits and other characters are encoded into longer bit sequences, and thus a smaller size of data message is obtained. **Arithmetic coding** [77] works similarly using variable length codes depending on probability of characters. The difference is that it represents an entire message using a single number instead of dividing a message into component symbols and replacing each using a code.

Other techniques with higher compression ratio provide **lossy compression** by discarding data deliberately and are more practical for multimedia streaming. Transform-based techniques translate data into another mathematical domain for data compression. The most popular ones include **Discreet Cosine Transform** (DCT) [78] where a block of data is represented as a series of cosine waves, fractal compression [79] which uses a mathematical transformation for data manipulation, stores repetitive information, discard unmatched data and wavelets transform which describes an image using a wavelet and calculates detail coefficients repetitively until the required outcome is obtained. The work in [102] proposes a video compression method which uses DCT coefficients to detect blocks with moving edges. Instead of performing object or region segmentation which is a widely used technique, the solution used DCT coefficients so that complexity is reduced in moving edge detection.

Prediction-based solutions [103] [80] encode only the difference between consecutive data sets and are mostly applied in audio streaming applications where signals change smoothly. There is another type of layered-coding based techniques which prioritizes data into different tiers and applies different compression technique or adjusts parameters based on the priorities. In [104], layers are distributed to different multicast groups and receivers are enabled to adjust their receiving rate by joining or leaving a multicast channel. The mechanism introduced in [105] identifies important information of images and transmit data with different quality of service: virtual wire, regular best effort and etc. The mechanism proposed in [106] categorizes video content based on relative preference per packet in terms of loss and delay. Vector quantization is another mechanism used in compression techniques such as **Code-Excited Linear Prediction** (CELP) [81], the main principle of which is to divide large vectors into groups and represent each group by a codeword. The energy optimization technique proposed in [107] consists of two parts. The first part is to adapt the encoding rate of each frame to the degree of motion activity. The adaptive rate scheme means encoding rate is decreased by skipping redundant frames when motion level is low without degrading video quality seriously. Two techniques are adopted for frame skipping: direct comparison and forward prediction. Direct comparison compares the current image with newly captured images and does not encode a frame until the comparison result shows violation of quality level or latency constraints. So the quality requirement will be met with the least number of frames encoded. The main disadvantages of this technique lie in that look-ahead buffer is required for comparison and the encoding of frame is delayed until the next frame is selected. On the other hand, forward prediction adjusts the frame skipping interval based on the difference between current frame and the previous images. The second part of the solution is to buffer the input data in order to delay data processing which results in longer slack time between consecutive processing and thus achieves energy efficiency. Other hybrid solutions [108] make use of both techniques in order to achieve higher efficiency or better quality.

Joint Photographic Experts Group (JPEG) [82] [83] is designed for image processing and provides both lossless and lossy options. It enables the flexibility of selecting

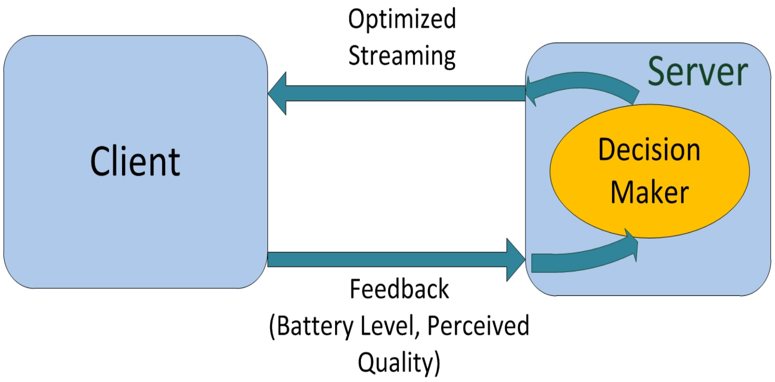


Figure 3.2 Illustration of adaptive control at application level

either high quality picture with low compression factor or low quality picture with high compression factor. Studies have demonstrated the high energy efficiency achieved by the technique [109]. **Moving Pictures Experts Group** (MPEG) group have developed several standards of compression techniques of moving pictures and audio.

MPEG-1 [84] and MPEG-2 [85] standardize compression of full motion video, inter-frame compression and utilize the difference between frames in the process of compression. MPEG-4 [110], standardised as ISO/IEC 14496, extends MPEG-2 by adding new features such as object-oriented composite files, error resilience, digital rights management support etc. MPEG-4 provides higher compression ratios and treats video as objects which could be handled both individually and collectively. **H.264/MPEG-4** Part 10 or Advanced Video Coding (AVC) [111] is a block-oriented motion-compensation-based codec standard. It is currently one of the most commonly used formats for the recording and compression of high definition video. The standard in [112] introduces XML for description of content and supports a broader range of applications [113].

3.1.2 Adaptive Control at Application Layer

With the goal to optimize energy efficiency or increase user quality of experience, one direction of designing application layer protocols is to dynamically adapts the behaviour of applications to channel environments, observed power level or obtained quality at user side, as shown in Figure 3.2.

Recent studies have focused on the interactive compression technique where the sender adaptively chooses the optimal compression policy for energy efficiency or quality of service. The solution presented in [8] adapts the compression strategy based on feedback from clients and guarantees the constrained minimum quality of service. The battery operated client sends to the server its maximum decoding capability, so the AC-operated server could calculate the optimal transmission rate. It is suggested that a match between decoding aptitude (M) and the number of correctly arrived packets at the client (A) results in best energy efficiency. If A is larger than M, the energy spent on handling A-M packets is wasted and on the other hand is A is smaller than M, the server should send more packets to improve video quality. Thus the feedback mechanism helps balance these two values and achieves energy saving. **QOAS** [86] uses feedback from clients which mainly includes estimation of user perceived quality and QoS metrics such as average loss rate to dynamically adjust streaming behaviour at the server side. Simulation results show significant increase in the number of clients that can be served simultaneously and meanwhile the quality of service is maintained at high level. The mechanism proposed in [114] uses a methodology to select the optimal image compression parameters at runtime to best balance the trade-off between energy/latency/image qualities. The methodology consists of two steps. In the first step, the average value of image quality and latency is calculated. In the second step, the data obtained from the first step is used to generate a table with quality and latency constraints and total energy consumption spent on computing and transmitting images. The table is then used to look up the optimal parameters for the desired energy/latency/image quality. **Adaptive Source Rate Control** (ASRC) [9] for video streaming applications is proposed to work with Automatic Repeat Request (ARQ) to take advantage of high throughput and reliability achieved by ARQ and at the same time to guarantee that data could get to the destination within the delay bound. Acknowledgement (ACK) packets received at the data source are used to calculate packet error rate which is an indicator of channel condition. The ASRC scheme forecasts the channel effective data rate based on the error rate before the next video frame is encoded. Finally the target number of bits for the next frame is calculated according to the channel condition, target delay bound so that the packet can be transmitted correctly and within the delay bound.

Rate adaptation techniques are widely used by some popular media services such as RealNetworks and Windows Media to support dynamic data rate adaptation. There are mainly three types of techniques used:

- **Stream Switch**. Stream switch supports multiple streams containing the same content but with different encoding rates. The technique enables users to switch dynamically among several streams based on the available bandwidth. Using RTSP, at the beginning when a media session is established, the client sends a DESCRIBE message to the server. In DESCRIBE response message the server informs the client about the description of each media stream encapsulated in the media object. The client then selects the desired media object and specifies it in the SETUP command. During playback, the client may generate a request to the server to switch to a lower stream rate if the available bandwidth drops down. This approach is deployed as Intelligent Streaming [1] in Windows' Media service and Sure Streaming [2] in RealNetworks' media service.
- **Stream Thinning**. In stream thinning, the data rate is decreased when the available bandwidth drops. Different from stream switch, the decrease is lowered through only delivering key frames to the client or reducing bitrate through transcoding.
- **Video Cancellation**. Video cancellation is used in the worst case where the available bandwidth is not capable of delivering the key frames, and only the audio stream is maintained. Using RTSP, the client may send a TEARDOWN command to cancel the video and may again setup and request video stream later again only if the bandwidth increases.

  However, stream thinning and video cancellation techniques are applicable only for video media streaming.

In [87], a pair of lower bound and upper bound values of the playout buffer is used to

[1] Fast Streaming-http://technet.microsoft.com/en-us/library/cc731688.aspx.

[2] Microsoft streaming media basics:real media surestream. http://service.real.com/help/library/guides/producerplus8/htmfiles/preparin.h

adjust the data rate. The lower bound of the buffer range helps to eliminate the latency for stream switching and the upper bound to prevent aggressive buffering. At the beginning of the streaming session, the server transmits data at the highest rate until the lower bound is reached. After that, the client begins the playback and data is buffered at the highest rate until the clients play-out buffer reaches the upper bound. At this point, the incoming data rate is set to the encoding rate and ideally the buffer is kept full. However, the buffer size might drop below the lower bound due to traffic fluctuation, and in this case the streaming rate is adapted for smooth playback.

HTTP Live Streaming [88] is an HTTP-based media streaming communications protocol implemented by Apple Inc. [3] in their QuickTime [4] software. The solution chops the whole stream into a sequence of small HTTP-based file downloads, each download loading one short chunk of an overall potentially unbounded transport stream. During the streaming process, the client is enabled to select from a number of different alternate streams containing the same content but with different encoding rate. Therefore it allows data rate adaptation.

Power-aware mechanisms are studied extensively, their goal is to maximize battery lifetime. The proposal presented in [89] adjusts video streaming strategy at runtime to prolong service time of a whole communication system in wireless environment. The authors observe that the video quality is determined by three aspects: encoding capability of the server, decoding capability of the client, and the channel. Therefore they propose a strategy where transmission power level at the server side and decoding scheme at the client side is adjusted to each frame based on the energy level at runtime, guaranteeing of minimum video quality levels. The adjustment is made with consideration of energy level at both sides, as the system life time is maximized if the server and client run out of energy at the same time. The solution proposed in [115] tries to monitor power level on mobile devices and decrease number of bits transmitted if power level drops as data transmission consumes a lot of energy in wireless communications. It introduced two techniques for re-

[3] Apple- http://www.apple.com/

[4] Quicktime- http://www.apple.com/quicktime/

ducing the amount of transmitted data. The first mechanism is to reduce number of bits in the compressed video stream if the video stream is encoded on the device instead of stored on it beforehand. The other technique discards some packets at the sender end. The main challenge for both techniques is to maintain reasonable video quality while conserving energy. The power-aware scheme introduced in [90] adapts the behaviour of applications dynamically according to the energy level in order to prolong the battery life for wireless devices. High quality of service is provided if the battery resource is plentiful and energy conservation is performed at the expense of user experience if a device is running out of energy. Experiments based on four different application types: video player, speech recognizer, map viewer and web browser are performed and testing results show that lowering data fidelity yields significant energy savings.

In general a media object is streamed at its encoding rate and at the client side a small amount of buffer is used to smooth the playback. However, the network conditions fluctuate and the playback buffer might get empty which affects user experience seriously via delays and loss. While streaming video is sensitive to bandwidth jitter, a receiver buffer can ameliorate the effects of jitter by adjusting to the difference between the transmission rate and the playback rate. Window Media services use a group techniques called fast streaming in order to provide high quality of stream, which includes Fast Start, Fast Cache, Fast Recovery, Fast Reconnect and Advanced Fast Start [5].

- **Fast Start**. Fast start refers to the procedure where the playout buffer at the client side is filled at a higher rate than the media encoding rate until the playout buffer is filled. The higher rate enables the client to reduce the startup buffering time. In addition, users can fast forward, rewind without any additional delay and re-buffering. Fast start only works with unicast TCP or UDP stream.
- **Fast Cache**. Fast cache, when enabled, refers to the procedure after fast start. It runs until the whole media project has been buffered or the media session is terminated by the user. It also streams media data at a higher rate than the encoding rate. While fast

[5] Windows Media-http://technet.microsoft.com/en-us/library/cc731688.aspx.

cache is running, client player also maintains a growing buffer for the early arrived data. The purpose of fast caching is to guard against network bandwidth fluctuations. It is extremely useful when the available network bandwidth of the client exceeds the required bandwidth of the content or the network latency is high or when the quality of the content received is of paramount importance.

- **Fast Recovery**. Fast recovery is used by the Windows Media player to recover the lost or damaged media without requesting that the data be resent by the Windows Media server. It is accomplished by using Forward Error Correction (FEC) on a Windows Media services publishing point.

- **Fast Reconnect**. Fast reconnect enables the client to reconnect to the server automatically and resume the streaming which reduces the impact of a temporary stream disconnection period. In case of on demand the stream is resumed from where it was interrupted by synchronizing itself with the content timeline. If video is included in the content, the video frame at which the connection was lost is estimated. However the broadcast clients may experience gaps in content reception as the client is reconnected to the broadcast in progress.

- **Advanced Fast Start**. Similar to fast start and fast cache, the media content is also being streamed at a higher speed than the encoding rate. However advanced fast streaming allows the client to play a media stream before its play-out buffer is full. The acceleration is terminated when the client buffer is full.

3.1.3 Partial Caching

Caching techniques are used in many applications to improve user experience as they can significantly decrease access time and delay. **Web caching** [116] [117] has been explored by a lot of researchers as web content normally is small in size and relatively static with small changes. Different from traditional web browsing or data downloading, multimedia streaming has tremendous amounts of data and asks for high user quality of experience. Download before watching is suggested and employed in some protocols, however

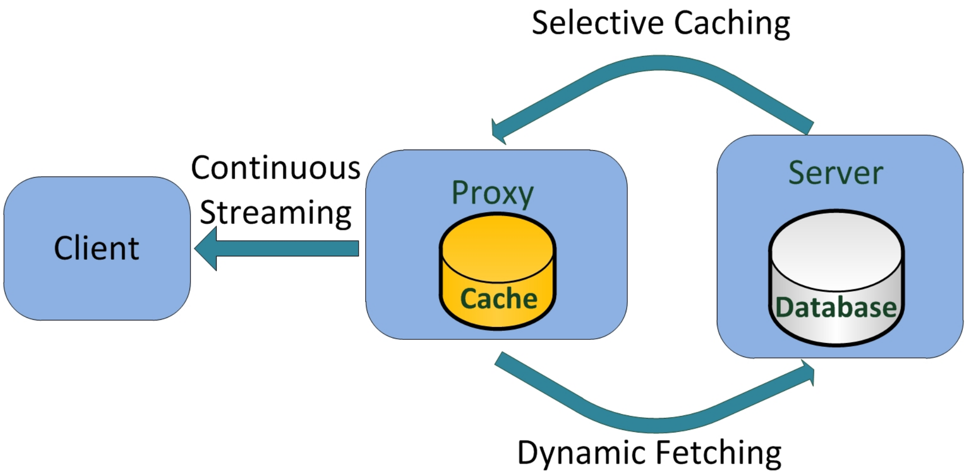


Figure 3.3 Illustration of partial caching

it is not feasible in many cases as it requires huge amount of buffer space at the client side, and a significant amount of start-up delay is introduced. Therefore partial caching of data using proxies is paid attention by many researchers in order to perform high quality streaming. The mechanism is depicted in Figure 3.3.

One way of partial caching is to perform layered caching based on different prioritizations of frames. A layered approach is assumed in [91] where each stream is distributed into different layers with priorities. The layering mechanism caches the base layer for each stream and discards the last segments of the least popular layer. It performs quality adaptation based on the variation in client bandwidth and lets the average quality of a stream be proportional to its popularity and the quality variation be inversely proportional to its popularity to finally achieve efficient cache state.

Some other approaches perform segment-based caching where objects are divided into segments and only a small part of them are cached to decrease start-up delay and the other part is fetched on demand. A prefix caching approach is proposed in [92] to store a large number of initial frames in the buffer and retrieve the remaining frames when a streaming request is made by the users. The size of cached data depends on both the physical constraints for example bandwidth and transmission distance, and the quality required by the user such as the maximum playback delay. The mechanism presented in [93] works in a similar way, while it also takes into consideration the video quality and the popularity of multimedia content when deciding whether to replace a video or to cache it. When the popularity of a video stream increases, the quality and size of cached frames increase. The solution proposed in [118] provides high quality of service to users and achieves high resource efficiency at proxies. It mainly consists of two parts: adaptive and lazy segmentation, and active pre-fetching. The adaptive and lazy segmentation adapts the segmentation function to user behaviours and tries to segment the video object as late as possible by adopting three functions. The first function is aggressive admission policy which caches the whole media object when it is accessed for the first time based on the assumption that the future access behaviour of a new object is unknown at the first access. An object is segmented adaptively according to the average client access length computed at runtime instead of before access. The two-phase iterative replacement policy will decide the candidate segments to be replaced based on the average number of accesses, the average duration of accesses, the length of cached data and the predicted future access probability which is calculated according to average interval between accesses and the current time. Continuous streaming is guaranteed by calculating the start point of a pre-fetching based on the streaming rate and pre-fetching delay at runtime. Some other protocols use the variable size segmentation technique. In the solution proposed in [94], the video stream is divided into variable-sized segments which are assigned different priorities in caching and replacement. The segmentation policy is based on the distance between a segment and the start point of the whole stream, which means the closer a segment is to the beginning of a frame, the smaller it will be. The reason behind this is to achieve high buffer utilization by discarding large size data chunks when replacement is performed. The replacement policy depends on the popularity of a segment and the distance to the beginning frame, where the popular segments and starting segments are treated preferentially. The mechanism used in [119] combines both uniformly sized segments and variable sized segments in dividing video streams.

3.2 Energy Efficient Transport Layer

Transport layer protocols provide end-to-end data transmission support and optionally offer functions such as congestion avoidance, reliability and flow control. **TCP** (Transport Control Protocol) [49] and **UDP** (User Datagram Protocol) [50] are two de-facto protocols employed at transport layer. These two protocols are designed and widely deployed in wired network communication environments. Although there are many issues related to using these two protocols in wireless communications, most of the transport layer solutions studied in this survey are based on either or both of these (i.e. TCP and UDP) and try to offset their weaknesses in order to make them suitable for wireless transmission in the context of applications such as multimedia streaming. Several variants of TCP have been proposed, each making improvements in terms of energy consumption, network throughput and reliability.

TCP Tahoe [120] [121] mainly contributes to the design of slow start, congestion avoidance and fast retransmission, and is the first protocol to include congestion control. This makes it energy efficient in case of bursty errors which happen quite often in wireless networks. **TCP Reno** [122] implements the three functions of Tahoe and adds additional fast recovery mechanism. **TCP New-Reno** [123] modifies the fast recovery scheme. The fast recovery function detects packet loss and initiates retransmission without the timeout signal required by traditional retransmission policies. In this case it provides shorter delay and better quality for multimedia streaming applications. **SACK** [124] uses selective ACK instead of cumulative ACK to indicate successful transmission of specific packets, and the sender is able to figure out which packets are lost and save the energy for redundant retransmission. Simulation results show that incorporating SACK in TCP achieves better performance in terms of packet delay and throughput [125]. SACK is supposed to be energy efficient as it decreases the number of unnecessary retransmissions, however the study in [126] points out that the energy gain is neutralized by the extra overhead. **Vegas** [127] modifies the congestion control scheme and adapts the transmission rate at sender side according to the observed Round Trip Time (RTT). **WestwoodNR** [128] differentiates the causes of packet loss, i.e. traffic congestion or error-prone wireless channel, and adapts the congestion window size at the sender side accordingly.

Performance concerns in terms of evaluating transport layer protocols for wireless networks mainly include the following:

- **Energy Efficiency**: energy consumption related to retransmissions, number of hops, and control overhead needs to be examined for wireless hosts in wireless networks.
- **QoS**: QoS includes packet delay; packet loss rate and delay jitter etc. Different applications have different requirements in terms of QoS.
- **Fairness**: fairness is especially required in wireless sensor networks as sensor devices are scattered and it is difficult for those far from sink to transmit data. Therefore the bandwidth should be allocated in a way that the sink can obtain a fair amount of data from all the sensor nodes.
- **Reliability**: reliability studied in most protocols includes reliable data transmission in either the downlink or the uplink or both. It mainly refers to the successful data transmission from source to sink, thus is examined as the success rate of a subset of devices or a whole network.
- **Congestion control**: congestion control metrics is further divided into centralized control and distributed control. Congestion control attempts to avoid over-subscription of any of the processing or link capabilities of the intermediate nodes and networks which leads to congestion and severe drop in delivery performance. For instance, automatic repeat requests may keep the network in a congested state; this situation can be avoided by adding congestion avoidance (e.g. slow start) to the flow control. In wireless sensor networks, upstream data from sensors to sink shows the feature of large quantity and burstiness while downstream from sink to sensors generally generates much less data, therefore congestion control is normally performed at upstream.

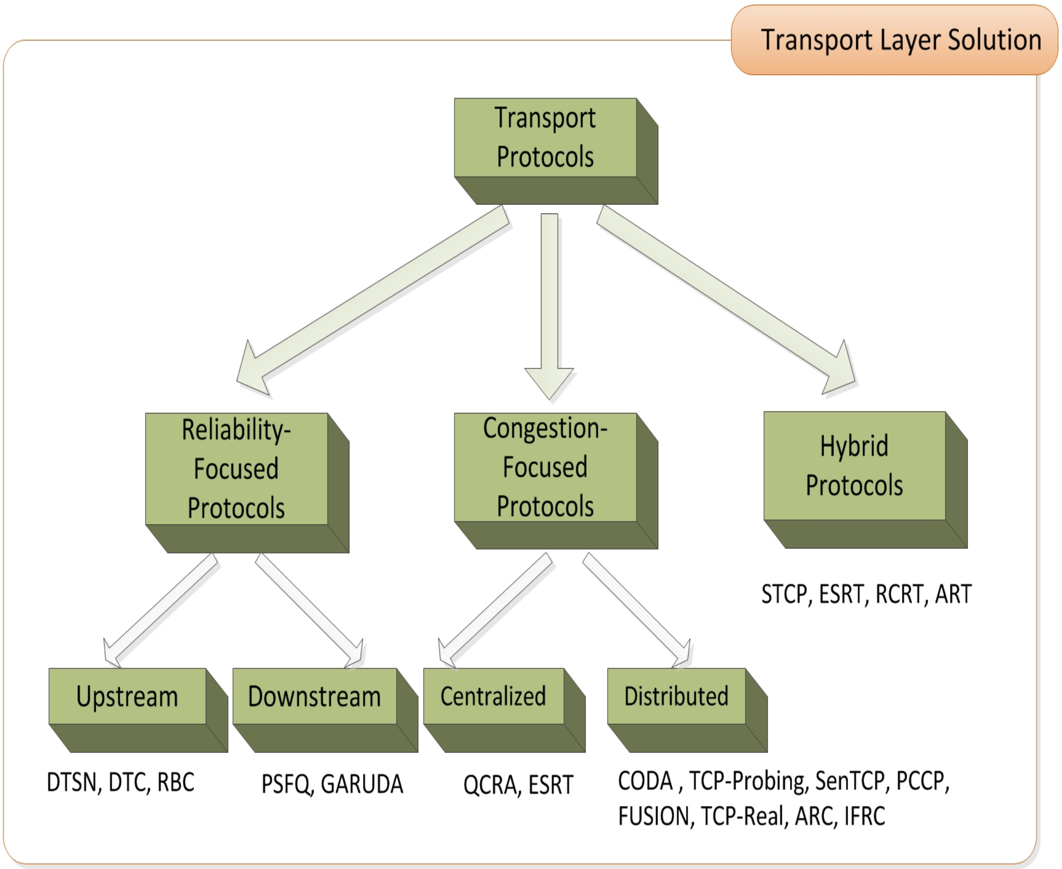


Figure 3.4 Classification of energy efficient transport layer protocols

The last two aspects are used in the classification of energy efficient transport layer protocols as most solutions improve either of them. Reliability-oriented solutions try to provide high reliability either for upstream or downstream data transmissions. Upstream transmission refers to data generated at wireless device and is collected at a central station, normally a base station in wireless networks. Most data flow in this direction is information detected from the environment by the devices and requires higher data rates, while downstream data often refers to control packets originated at base station and is relatively smaller in size. Therefore congestion control focused approaches normally target upstream and solve the problem of error-prone communication in this direction. Congestion control is then divided into centralized control where control behaviour is coordinated at the base station and distributed control where each host adjusts its own response to congestion. A classification of transport protocols is shown in Figure 3.4.

Table 3.2 presents an overview of the energy efficient solutions for the transport protocols which will be discussed in this section, according to the classification depicted in Figure 3.4.

Table 3.2 Energy Efficient Transport Layer Protocols

Protocol	Type	Description
DTSN [10]	Upstream reliability-focused	Differentiates reliability according to application requirements.
DTC [11]	Upstream reliability-focused	Uses distributed segment caching and local retransmissions.
RBC [129]	Upstream reliability-focused	Uses block acknowledgement instead of traditional sliding window and differentiates retransmission priority according to retransmission times.
PSFQ [130]	Downstream reliability-focused	Employs low rate message relaying at a regular basis, high rate data fetching when loss is detected, and selective status reporting to reduce overhead.
GARUDA [131]	Downstream reliability-focused	Divides nodes into core ones and non-core ones and employs local retransmission.
QCRA [132]	Centralized congestion control	Takes into consideration of channel capacity and network topology.
CODA [133]	Distributed congestion control	Notifies neighbours when congestion is detected.
TCP Probing [134]]	Distributed congestion control	Employs an error control scheme to determine whether to initiate congestion control or to resume transmission.
TCP-Real [135]	Distributed congestion control	Receiver-side information is used to measure channel condition and to adjust congestion control on the sender side.
IFRC [136]	Distributed congestion control	Uses average queue length and topology tree to determine congestion condition and adapts congestion control.
STCP [137]	Reliable and congestion-aware	Exchanges information before data transmission in order to provide required reliability and congestion control.
ESRT [138]	Reliable and congestion-aware	Adjusts reporting rate for ad-hoc hosts according to real time detected reliability and congestion condition.
RCRT [139]	Reliable and congestion-aware	Provides reliability through employing NACK-based scheme, and regulates total rate of all flows connected to a sink when congestion is detected at the sink.
ART [140]	Reliable and congestion-aware	Provides both event reliability and query reliability for sensor nodes, and adjusts sending rates through message relying among essential nodes and non-essential nodes, classified by remaining energy level.

3.2.1 Reliability-Focused Transport Protocol

1. **Upstream reliable transport**: **Distributed Transport for Sensor Networks** (DTSN) [10] provides reliability differentiation to adapt to different requirements of applications, so different needs on throughput latency and energy consumption could be met. For full reliability service, DTSN adopts the end-to-end Selective Repeat ARQ and employs caching at intermediate devices so the number of end to end retransmissions is minimized and packet delay is shortened in order for reliability to be improved. For differentiated reliability service, DTSN employs the Enhancement Flow Strategy where only the important data is buffered at the source and remaining data is only cached at intermediate hosts and delivered with best-effort reliability.

   Distributed Tcp Caching (DTC) [11] is built on TCP and mainly uses segment caching and local retransmissions to reduce the number of end-to-end retransmission in wireless sensor networks. In this caching scheme, packets are cached at intermediate hosts before they are successfully transmitted to the receiver end. Retransmission happens from the intermediate hosts instead of from the source if packet loss is detected so that retransmission could be reduced. Moreover, packets are cached at each host with a probability of 50 percent so they are better distributed among all intermediate hosts. Therefore the energy consumed on caching and retransmission is evenly distributed; otherwise some hosts, especially those close to the base station, will deplete their battery quickly. The distributed caching mechanism is depicted in Figure 3.5. The first two data segments are cached before being dropped at node 2 and node 3 respectively. When segment 3 reaches the destination, an acknowledgement (ACK 1) is sent to node 2, followed by retransmission of segment 1 from node 2 to receiver. Segment 2 is then retransmitted as soon as node 3 receives ACK 2 before the final acknowledgement is transmitted back to the sender.

   In **Reliable Bursty Convergecast** (RBC) [129], continuous packet forwarding is guaranteed by using block acknowledgement instead of traditional sliding window. Given that the sliding window and in-order delivery requires acknowledgement on

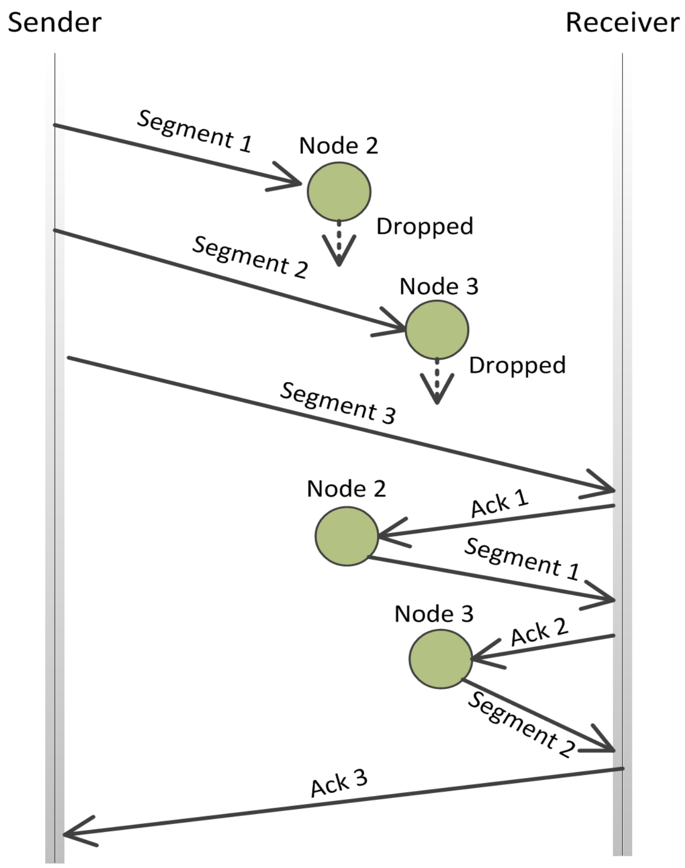


Figure 3.5 Illustration of distributed TCP caching

transmission of each single packet which leads to decrease in throughput and increase in packet delay, it is more suitable to detect the successful transmission of a sequence of packets. Moreover, RBC increases reliability by differentiating packets on their retransmission times. Packets that are transmitted for the first time or retransmitted less number of times are given higher priority for channel contention.

2. **Downstream reliable transport**: **Pump Slowly, Fetch Quickly** (PSFQ) [130] provides hop-by-hop downstream reliability for the purpose of re-tasking the wireless devices, or for control messages or network management. It mainly comprises of three parts: low-rate message relaying phase (pump), high-rate error recovery phase (fetch) and selective status reporting (report). The pump phase mainly functions as a message is broadcasted every T unless all fragments have been sent out. The T is designed to provide a host the opportunity to detect packet loss before the next packet

arrives, and it also helps reduce broadcasting redundancy. And a host switches into fetch mode once packet loss is detected, and it will try to retrieve all lost packets in a single fetch if possible. And the selective status reporting mechanism saves the overhead introduced in feedback procedure by letting intermediate hosts piggyback their report messages in an aggregated way.

Similar to PSFQ, **GARUDA** [131] detects packet loss by adopting NACK scheme and uses local retransmission. It achieves reliability mainly by dividing hosts into core ones which are guaranteed to receive packets from base station successfully and non-core ones which could recover lost packets from core ones.

3.2.2 Congestion-Focused Transport Protocol

Congestion control transport layer protocols aim at preventing, detecting, notifying and recovering from traffic congestion at intermediate hosts and links. Different form congestion control at downstream data flow which can be controlled by the based station, upstream congestion control presents bigger challenge and is extensively studied by various research works. Based on whether the control is taken individually by each host or is handled by a central base station, congestion control protocols are divided into two categories.

1. **Centralized upstream congestion control**: Most transport layer protocols for congestion control are designed and implemented in a distributed way where each host takes care of its congestion window and transmission behaviour, and the main centralized protocols include **Quasi-static Centralized Rate Allocation** (QCRA) [132] and **Event-to- Sink Reliable Transport** (ESRT) [138]. The rate allocation scheme in QCRA takes into consideration the channel capacity calculated by transmitting packet, the loss rate measured by sending probe packets, and the network topology. The network topology is used to divide devices into subsets that do not interfere with each other which means they are able to transmit at the same time. The bandwidth is then allocated according to the capacity of sets and also the expected number of retransmissions.

2. **Distributed upstream congestion control**: Protocols such as [133], [134] initiate probing or sampling sessions after a potential congestion is detected to determine the real channel condition and take corresponding actions according to the sample results. **COngestion Detection and Avoidance** (CODA) [133] mainly comprises of three components: congestion detection, hop-by-hop backpressure and multi-source regulation. The congestion detection scheme mainly lets a mobile to sample the channel regularly when a packet is waiting in the queue and the buffer length is high. A utilization factor is calculated based on the times when the channel is sampled and found busy, the congestion bit is set and a suppression message is broadcasted to neighbours if the factor is above a threshold value. Both upstream and downstream neighbours will regulate their rates to prevent congestion. CODA tries to execute congestion control over multiple sources from the sink if congestion is persistent. CODA solves the problem of ESRT where all source hosts are asked to regulate their on-going traffic by letting a host notify its neighbours only when congestion is detected.

   TCP-Probing [134] improves the congestion control scheme and also contributes to energy efficiency. It is deployed on top of the TCP protocol. TCP-Probing proposes an error control scheme where a probe session is adopted whenever an error occurs to check the link condition, normal congestion control will be invoked if persistent error is sensed, or the source will resume transmission when the link condition is fine.

   Some protocols adopt novel indicators other than packet loss to detect congestions. **SenTCP** [141] [142] use the local service time and packet inter-arrival time as congestion indication and FUSION [143] watches the buffer of a host to determine channel condition. **TCP-Real** [135] is a TCP friendly protocol and estimates contention level in wireless channels, distinguishes the reasons of packet loss, and takes appropriate actions accordingly. On the contrary of what TCP does, TCP-Real lets the receiver of multimedia data measure data rate and estimate channel condition and accordingly adjusts the congestion window at the sender side. TCP-Real introduces a concept of wave, i.e. a sequence of data, which is used to calculate the receiving rate of data and compared with previous waves in order to predict the traffic condition more accurately than ACK-based mechanisms. The reason behind this is that if a packet is lost due to transient wireless errors, the receiving rate will not be affected by the gap of packets, and therefore congestion window will not be affected. Adaptive Rate Control (ARC) [144] detects traffic congestion at intermediate hosts, which increase their packet sending rate packets are forwarded successfully by their parents and decrease sending rate otherwise. **Interference-aware Fair Rate Control** (IFRC) [136] uses the average queue length to determine congestion and it mainly consists of three parts. The first component detects congestion by looking at the average queue length as it is assumed that retransmissions take place when congestion happens. If the observed value is larger than a threshold, the congestion control scheme is initiated. The second component uses a topology tree to determine a host's potential interferers and the congestion status gathered by the first component is shared with those interferes. And the last component adapts sending rate of hosts based on the congestion status.

3.2.3 Reliability and Congestion Supported Transport Protocol

In **Sensor Transmission Control Protocol** (STCP) [137], an association is established between source and destination before data transmission to provide information such as type of data, transmission rate and level of reliability required. The packet drop policy and congestion control scheme are regulated during transmission to reduce energy consumption as much as possible and at the same time meet reliability requirements. Experimental results demonstrate lower delay and higher throughput than TCP [145].

Event-to-Sink Reliable Transport (ESRT) [138] intends to save energy and maintain reliability for ad-hoc based wireless hosts by regulating reporting rates. If real time calculated reliability is higher than the required reliability, the source device will reduce the reporting rates so energy could be saved without degrading performance. ESRT adopts a congestion control scheme so that when congestion is detected, the reporting rates are adjusted. To be more specific, if the buffer overflows in a wireless node, it notifies the sink about the congestion, and the sink will broadcast with high energy a signal letting all source devices to decrease their reporting frequency. The main problem existing in this protocol is that buffer overflow at one node would impact all source nodes which might not be responsible for the assumed congestion, which means any on-going transmission will be disrupted. Additionally, the impact on all source nodes could be completely unnecessary if the congestion is not persistent, which means interruption of on-going traffic is a waste of channel resources.

Rate-Controlled Reliable Transport protocol (RCRT) [139] mainly consists of four components: end-to-end reliability, congestion control, rate adaptation and rate allocation. End-to-End reliability is 100 percent guaranteed by the NACK-based loss recovery scheme to prevent the channel being overwhelmed by ACK messages. In contrast to most congestion control schemes where congestion is monitored at each host, RCRT measures congestion condition at the sink in order to get a more extensive view of the whole network. The time spent on recovering lost packets is used as an indicator of congestion as congestion could be predicted if the average time needed to recover lost packets is long. After the detection of congestion, network-based rate adaptation is used to adapt the total rate of all flows connected to the sink and rate allocation component will assign proper rate to each traffic flow.

Asymmetric and Reliable Transport (ART) [140] is an energy-aware scheme that provides both upstream and downstream reliability as well as congestion control mechanism in wireless sensor networks. Reliable data transport is based on classifying nodes into essential ones (E) and non-essential (N) ones. The essentiality is determined by its remaining energy level, which means nodes with higher residual energy level are selected in order to achieve overall energy efficiency. To achieve event reliability which is downstream reliability, a lightweight ACK mechanism is used and for query reliability, which guarantees reliable upstream transport, a NACK-based mechanism is adopted. In both ACK and NACK streams, all nodes participate in message relying if no congestion is detected while only the E nodes are responsible for packet acknowledgement and retransmission. The distributed congestion control scheme is handled by E nodes and the loss of ACK is used as congestion notion. When congestion is detected, the notification is sent by E to N nodes which then adapt their sending rate accordingly.

3.3 Energy Efficient Network Layer

Network layer is responsible for packet routing and forwarding in computer networks. It involves a process of selecting a route which consists of multiple intermediate nodes or even multiple networks to deliver packets from their source to destination. Often the route also meets certain quality of service requirements at the same time. The process of selecting routes is called routing and is a main function of network layer protocols.

Routing protocols can be devised according to different criteria, for example shortest path or maximum network throughput. Most traditional protocols try to calculate the shortest path from source to destination, and regularly provide minimum delay. However, another critical factor of route selection in wireless communications is energy efficiency, especially for sensor network where intermediate nodes are battery powered and the depletion of battery might lead to inactivity or partial activity of the network. Each host may have different energy constraint, for example energy depletion rate, battery capacity etc, which means energy efficient solutions can employ some of those metrics to obtain an optimal performance of the whole network. A classification of energy efficient network layer protocols is illustrated in Figure. 3.6.

Table 3.3 presents an overview of the energy efficient solutions for the network layer protocols.

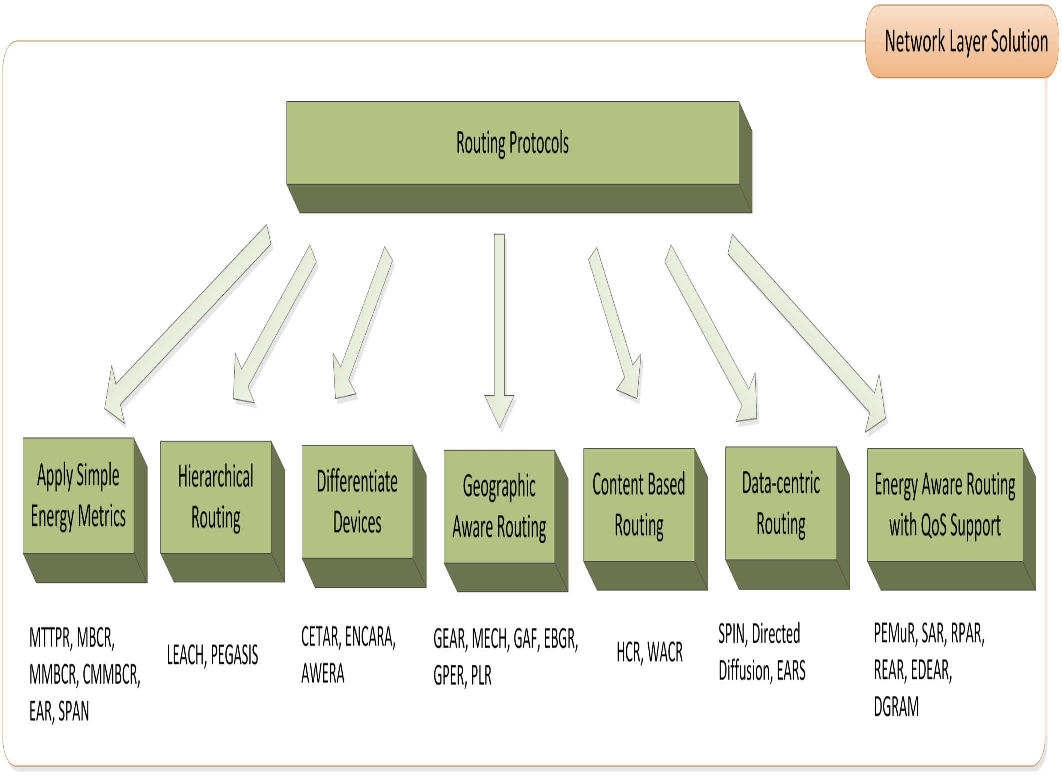

Network Layer Solution
Routing Protocols
Apply Simple Energy Metrics
Hierarchical Routing
Differentiate Devices
Geographic Aware Routing
Content Based Routing
Data-centric Routing
Energy Aware Routing with QoS Support
MTTPR, MBCR, MMBCR, CMMBCR, EAR, SPAN
LEACH, PEGASIS
CETAR, ENCARA, AWERA
GEAR, MECH, GAF, EBGR, GPER, PLR
HCR, WACR
SPIN, Directed Diffusion, EARS
PEMuR, SAR, RPAR, REAR, EDEAR, DGRAM

Figure 3.6 Classification of energy efficient network layer protocols

Table 3.3 Energy Efficient Network Layer Protocols

Protocol	Type	Description
MTTPR [146]	Apply Energy Metric	Select the route with the least total energy consumption.
MBCR [147]	Apply Energy Metric	Select the route with the most remaining battery capacity.
MMBCR [147]	Apply Energy Metric	Improve MCBR by checking remaining battery capacity across devices.
CM-MBCR [147]	Apply Energy Metric	Hybrid solution mixing MTTPR and MBCR.
EAR [12]	Apply Energy Metric	Consider both energy cost of each hop and the residual energy level of each host.
SPAN [148]	Apply Energy Metric	A distributed algorithm in which each node decides to sleep or forward by evaluating the available energy and its contribution to the route.
LEACH [13, 149]	Hierarchical Routing	A cluster based solution that most nodes only communicate with the cluster head, and only the head communicate directly with the base station.
PEGASIS [150, 151]	Hierarchical Routing	A chain based solution that improves LEACH by letting cluster member nodes only communicate with the neighbour instead of the head.
CETAR [152]	Differentiate Devices	Differentiate devices by their major role in the communications: sender or receiver.
GEAR [153]	Geographic Aware Routing	Forward packets towards the direction of the destination node.
MECN [154]	Geographic Aware Routing	Use location information provided by GPS to compute the most energy efficient route.
GAF [155]	Geographic Aware Routing	Use location information to divide the topology into grids, and use cluster based mechanism for intra-grid and inter-grid communications.
EBGR [156]	Geographic Aware Routing	Apply beacon less routing, where a localized routing decision is made when a host has a packet to transmit.
SAR [157]	Energy Aware QoS Support	Construct multiple routes with different QoS and energy constraints.
RPAR [158]	Energy Aware QoS Support	Energy efficient real-time communications: dynamic adaptation based on packet deadlines.
QoS support and local recovery Routing [159]	Energy Aware QoS Support	Make hosts query the bandwidth availability to estimate needed bandwidth.
Cross layer solution for TCP/RTP multimedia [160]	Energy Aware QoS Support	A cross layer solution for TCP/RTP based multimedia applications in heterogeneous wireless networks.
EDEAR [161]	Energy Aware QoS Support	Apply reinforcement learning to achieve energy and delay efficient routing.
DGRAM [162]	Energy Aware QoS Support	A TDMA based energy efficient and delay guaranteed routing solution.
EARS [163]	Energy Aware QoS Support	An energy efficient and reliable data-centric routing protocol uses MAC layer detection to reduce unnecessary overhead traffic in data-centric routing.
SPIN [164]	Data-centric Routing	Allow source node to broadcast advertise data before the actual data to improve flooding routing.
Directed Diffusion [165]	Data-centric Routing	Present data as attribute-data pair so that nodes only accept traffic they are interested in.
Routing using passive measurement [166]	Content-based Routing	Make routers content aware and monitor each traffic flow in routers for better performance, energy efficiency and load balance.
HCR [167]	Content-based Routing	Saves energy by organizing servers with replicated content in a hierarchical structure.
WACR [168]	Content-based Routing	Tag packets with corresponding routing policy according to the content, in order to save the effort for consequent packets in the same flow.

3.3.1 Energy Aware Routing

Limited battery capacity has brought energy aware routing protocols to research attention for the last few decades. The traditional shortest path criterion might not be the best solution as some intermediate wireless hosts may not be operational anymore if they are frequently picked by routing protocols or have low battery capacity. Therefore route selection algorithms have taken into consideration the energy consumption of each possible route for packet delivery.

Minimum Total Transmission Power Routing (MTTPR) [146] calculates the total energy consumption of each path when selecting routes. The information of energy consumption at each host is obtained and added up in MTTPR, and the path that consumes the least energy is chosen for packet forwarding.

Besides total energy consumption, some other metrics are evaluated such as battery capacity and remaining energy level. To prolong the life of the network as a whole, **Minimum Battery Cost Routing** (MBCR) [147] considers the remaining energy as the most important factor of routing algorithm. MBCR calculates the remaining energy of a whole path which is the sum of remaining energy of each hop and uses the path with device along it of most remaining energy. However, MBCR evaluates the energy of a path as a whole without considering individual nodes. Frequent choosing of a node as part of routing path may lead to quicker energy depletion and inactivity of part of or even the whole network death. MBCR avoids the hops with less energy to balance the remaining energy level of whole network and maximize the lifetime of whole network. **Conditional Min-MBCR** (CMMBC) [147] on the other hand combines these two solutions where both the remaining energy of a host and the total energy of a path are used for route selection.

Energy Aware Routing (EAR) [12] is a solution that normalizes both the value of energy consumption and residual energy level to achieve better energy distribution among nodes within a network. Each node obtains the processing energy value and remaining energy of its neighbours and a probability is given to each neighbour. Higher probability is assigned if the neighbour has more energy left and requires less energy for packet processing and forwarding. Neighbouring hosts with higher priorities are chosen to forward data as part of route.

Some of the routing protocols are based on centralized algorithms where the energy information is gathered at a host before the packet is forwarded, while other works distribute the decision of whether to participate in data transmission is made by the intermediate hops themselves. In **Span** [148], the available energy as well as the contribution of a host is evaluated by intermediate hosts to decide whether they should sleep or be active. The contribution of a host is a metric that refers to the number of pairs of hosts a node can help connect. The distributed feature of Span decreases routing overhead and increases scalability.

Hierarchical based routing protocols such as **Low Energy Adaptive Clustering Hierarchy** (LEACH) [13] [149] have been proposed to conserve energy. In this scenario, a host does not need to talk directly to the base station to exchange information about how far the node is from the base station. Instead, the network is divided into clusters where a host is elected as the head to communicate with the base station, and the remaining nodes only need to talk to the head which is closer than the central station which is more energy efficient. **Power-Efficient GAthering in Sensor Information Systems** (PEGASIS) [150] [151] saves more energy by forming chains within a cluster so that each node only communicates with two neighbours which are its previous and next nodes in a chain. However the cluster-based solutions have two obvious drawbacks. First, the battery of elected head tends to be drained as it is required to exchange information between all the remaining nodes within the cluster and the base station. Second, energy efficiency is achieved through short distance communication and the mechanism might not be effective when deployed in wide area. The problem is address in [169], where a multi-tier solution is proposed and the role of cluster head is dynamically chosen. In this solution, quick energy depletion is avoided as the role of head is rotated and short distance communication is guaranteed by introducing multiple tiers in clusters.

Geographic-based routing protocols make use of geographical information to choose routing path as it is more energy efficient to communicate with hosts that are closer as in cluster-based solutions and also as better performance is achieved if a packet is forwarded in the direction towards its destination. In **Geographical and Energy Aware Routing** (GEAR) [153] a path is selected from source to the target area which consists of a small range of hosts including the destination device before the packet is transmitted within the area to the exact destination. GEAR first defines the target area which is the preliminary direction for the routing path to follow as a set of hosts including the destination host, and then a refined routing algorithm is applied to deliver the packet to its recipient. **Minimum Energy Communication Network** (MECN) [154] and **Geographic Adaptive Fidelity** (GAF) [155] make use of GPS to track the address of each hosts location. The address information in GAF is used to divide the area into grids and each host is fitted into a grid. The hierarchical routing method is also applied in GAF where a head node is elected within each grid as a central host to forward data to other hosts.

Traditional geographic-based routing protocols may lead to control overhead caused by dissemination of beacon data which is used to maintain address information within a network. In order to address this problem, **Energy-efficient Beacon-less Geographic Routing protocol** (EBGR) [156] adopts a localized solution where a localized routing decision is made only when the host has data to send. The choice of path is based on calculation of the ideal position of the next hop based on the direction of sink and energy efficiency of path, and then the neighbour that is closest to the ideal position is chosen as the next hop with respect to the upper bound of distance and energy consumption.

Protocols such as [152] [170] differentiate devices according to their energy features as part of their routing algorithms. As differently devices may join a network with different features and energy requirements, Energy-oriented Node Characteristics-Aware Routing Algorithm (ENCARA) [170] evaluates both the hardware and software specifications of each device to prolong the battery life of energy critical hosts. **Energy Type Aware Routing** [152] categorized devices to senders and receivers. Devices that send packets most of the time are considered active senders and are not preferred as part of routing path as these hosts tend to generate traffic and require more energy transmitting and processing data.

3.3.2 QoS-oriented Energy Aware Routing

With the increasing popularity of multimedia applications, the high requirements on quality of service provided to users are another challenge faced by routing protocols, especially energy aware solutions which normally compromise the delivery performance.

Real-time Power-Aware Routing (RPAR) [158] is a protocol that guarantees the minimum required quality of service levels while decreasing energy consumption. Neighbour discovery in RPAR is not initiated until a packet transmission is required and there is no establishing route in the routing table. Therefore it achieves energy saving as periodical beacon is avoided and only neighbours that meet certain requirements such as high velocity are asked to reply to join routing table. It also dynamically adopts the transmission power using multiplicative increase and linear decrease to achieve maximum energy efficiency. When the required velocity of a packet is met, RPAR will decrease transmission power to improve energy efficiency.

Both energy and QoS constraints are evaluated in the **Sequential Assignment Routing** (SAR) [157] as part of the route selection process. SAR was designed for Wireless Sensor Networks (WSN) with low mobility. The main idea is to establish trees at a node which contains multiple paths to other hosts with the quality of service and energy constraint maintained. Each path is evaluated in response of the maximum number of packet forwarding before battery depletion, packet delay, and packet priority and the results are compared in order to select the optimal route.

Bandwidth allocation efficiency has been exploited in many energy aware routing protocols such as [159] [160]. In [159], hosts are required to send bandwidth availability query to neighbours and estimate the required bandwidth for effective packet transmission.[160] attempts to address the problem of varied bandwidth in heterogeneous networks. It adopts cross-layer solution where application layer provides information of device mobility pattern and estimation of next access point to connect, and physical layer assists detection of overlapped networks and possible hand-over decision. Transport layer creates two connections and sends dummy packets for the purpose of estimating the bandwidth of the new connection.

Besides bandwidth, other metrics are used to perform routing such as route quality. For example, **Energy Efficient and Reliable Routing Scheme** (EARS) [163] adopts cooperation between network layer with MAC layer, which gathers information of data rate and frame error rate, to choose the path with better quality in order to provide better quality in terms of reliability and energy efficiency.

Energy and Delay Efficient State Dependent Routing (EDEAR) [161] adapts its routing table to the network environment in order to satisfy QoS requirements. It applies reinforcement learning which collects information of environment parameters such as energy consumption of each link, remaining energy level of hosts, based on which the cost of each path and packet delay is calculated. Reinforcement learning enables adaptive routing and uses explore agent to learn from past experience in order to adjust the optimal path constantly.

Delay Guaranteed Routing and MAC (DGRAM) [162], as a TDMA-base solution, reuses slots to decrease packet delay. Hosts are required to be uniformly distributed and location information is exchange within networks to build a topology tree. The topology is established with different tiers so that packets are guaranteed to be transmitted from the inner tier to outer tier towards their destinations. According to mathematical analysis, DGRAM is supposes to provide delay guaranteed routing with energy efficiency.

Instead of traditional address-centric routing where routing is performed in an end-to-end manner according to IP address, some studies allow data-centric routing where routing is done based on the content of data packets. In **Directed Diffusion** [165], pairs of attribute-value are treated as basic units of data and hosts spread their interests for certain kind of data. Events as reply to the queries are generated and flow towards the originator through multiple paths, one or a small number of which are reinforced by the network. Directed Diffusion introduces the novel feature of in-network data aggregation and caching which means data is cached in intermediate nodes for further queries so that the amount of end-to-end traffic is decreased. **Sensor Protocols for Information via Negotiation** (SPIN) [164]

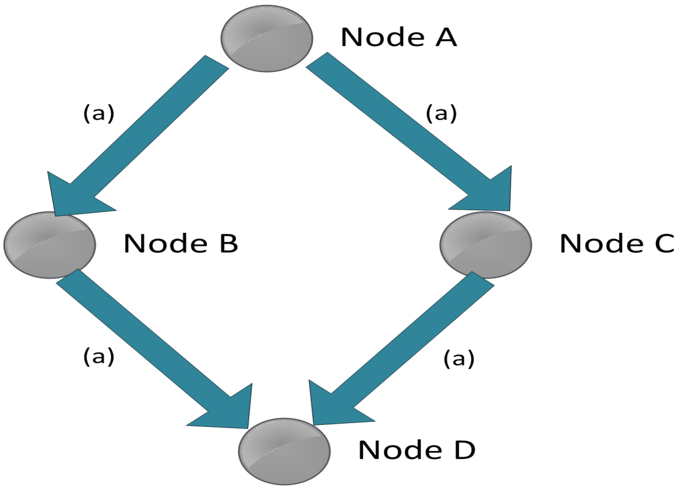


Figure 3.7 The implosion problem. A broadcasts data to B and C, and two copies data is sent from B and C to D eventually

is another data-centric protocol which allows broadcasting of advertising data before real data transmission to solve the implosion problem and the overlap problem. **Implosion**, as illustration in Figure 3.7 is a common issue in flooding algorithms and refers to the redundant data receiving caused by a node sending data to its neighbours regardless of whether or not the neighbour has already received the same copy of data from other sources. **Overlap** happens when two nodes cover an overlapped geographical area and send overlapping data to the same neighbours. Figure 3.8 illustrates what happens when two nodes send overlapped data to the same recipient. Both problems of implosion and overlap waste energy and bandwidth. Under these circumstances, SPIN instructs sensors to broadcast data advertisements before the actual data transmission, allowing neighbor hosts to check if the advertised data has been requested in order to conserve energy by avoiding unnecessary transmissions. It further saves energy by letting nodes check energy availability before participating in data transmission and only hosts with enough energy are able to request data.

The idea of **Content Delivery Network** (CDN) is derived from replicated server systems and cooperative caching/streaming systems due to increasing request of multimedia content. It consists of distributed servers caching parts of data. Queries are made to the clients nearby CDN server and the server check if it has the content. Content is sent back

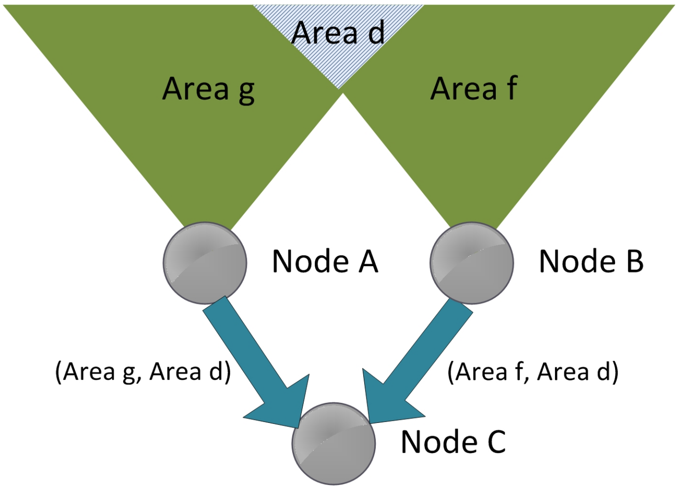


Figure 3.8 The overlap problem. C receives data from A containing information about area g and d, and C received data from B containing information about area f and d, two duplicate information about d are transmitted

to client if it is available or content routing is performed to locate and deliver the content in the network. **Hierarchical Content Routing** [167] is a CDN-based protocol and categorizes servers containing the same content into clusters to form a hierarchical structure. Both intra-cluster-wise and inter-cluster-wise routing are performed for content routing. For intra-cluster routing, each router maintains a hashing table for content cached by other servers to locate the nearby server when a piece of data content is required. For inter-cluster based routing, requests are made by cluster head to neighbour clusters for data needed. **Wide area content based routing** [168] uses tags to save energy for content-based routing. It constructs virtual content network between clients and servers. The first query from the client is tagged and all the following packets within the same traffic flow are then assigned the same tag. This tag is used for routing purpose so energy efficiency is achieved by saving the efforts of routing of packets within the same flow. Miura et al. [166] applies probabilistic server selection in case client requests concentrate at a server. It further reduces response time by measuring the RTT information at a router. The shorter RTT reflects quicker response as server latency and network delay are dominant elements of user response time.

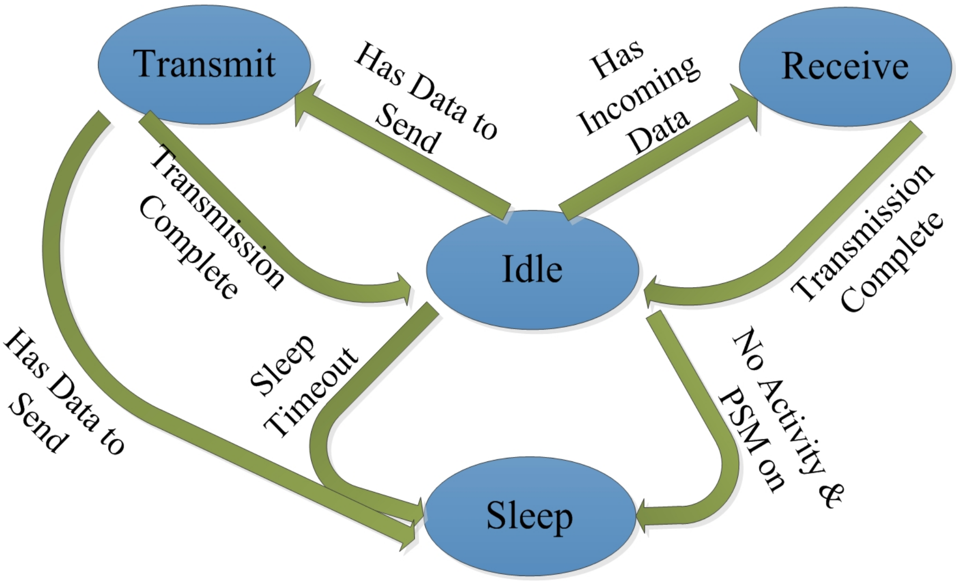


Figure 3.9 State diagram of WNIC

3.4 Energy Efficient MAC Layer Solutions

3.4.1 MAC-Layer Energy Efficiency Approaches

A typical Wireless Network Interface Card (WNIC) has several states, each involving different energy consumption levels: transmit, receive, idle, and sleep. A wireless host in transmit and receive mode consumes the most energy, while a host in sleeping mode which is also known as power saving mode (PSM) does not participate in network activities thus is far less energy demanding. While in idle state, a network device is neither transmitting nor receiving, however it is prepared to participate in communication. Although an idle host does not spend as much energy as in transmit/receive mode, it still requires significant amount of power. The state transition of a typical WNIC is shown in Figure 3.9.

The energy inefficiency of MAC-layer protocols is altered by five aspects:

1. **Idle Listening**

   Nodes in wireless networks can not know exactly when the next packet addressed to them will arrive. Therefore they may need to wake up very often or stay active to prevent packet loss due to sleeping. In this case, energy is wasted for listening to nothing of interest in the channel, which is called idle listening. This is one of the major sources of energy waste, as power consumed in listening mode is hundreds or thousands times bigger than that used in sleeping mode.

The easiest and most obvious method to reduce energy wasted by idle listening is to put a node into sleep node and make it active again, regularly. When a node is switched off, any packets reaching for it should be stored in a queue, buffer of the sender or access point and transmitted later when the node wakes up. In this manner, how much energy is saved depends on the cycle of listening and sleeping period. The longer a node sleeps the smaller chance that idle listening will take place. However, packets may easily get dropped if the queue is full, and serious delay may be introduced due to long sleeping periods.

Other solutions to reduce idle listening time are proposed in order to give a better performance in terms of energy saving and packet delay. For example, some algorithms ask nodes that connect directly with each other to form a cluster which share the same schedule of sleep and active period [171] [172]. Listening period would be reduced this way as when neighbor nodes are sleeping, there is no chance of data coming to the node. Some other protocols provide prediction algorithms that can foresee the upcoming packets arrival [173]. In this manner, nodes will wake up when it is likely that packets will arrive, and idle listening can be reduced based on how well the prediction works.

2. **Collision and Retransmission**

   Packet collision is introduced in the contention based medium access control protocols. As nodes have no idea if any other node is transmitting, two or more nodes may start to transmit at the same time. In this case, those packets transmitted will be corrupted, and the data needs to be retransmitted. However, in order to prevent continuous collision, retransmission should not be right after the previous packet collision. Most protocols have proposed a back off algorithm which specifies for how long a node should wait before the retransmission.

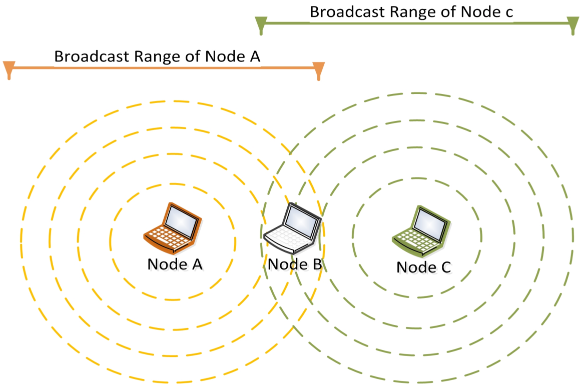


Figure 3.10 Hidden node problem

This problem can be solved or at least the situation can be improved through some collision avoidance methods such as the widely used concept of carrier node. Nodes will listen to the shared channel first to see if other transmissions are carrying on and, if there is any, no packets will be sent; otherwise transmission will begin. Although energy may be wasted in sensing the medium and waiting for another chance to transmit, radio can be turned off during the waiting period and real transmission will take place only once.

Another problem called **hidden node problem**, as presented in Figure 3.10, could arise if carrier node method is used. Suppose there are three nodes A, B, and C. A and B are within the same range, and B and C can also communicate directly. However, A and C cannot hear each other. If node C is communicating with node B, which cannot be heard by A, then node A may mistakenly think B is not active, and therefore sends out packets to B. In this case collisions occur. Request To Send/ Clear To Send (RTS/CTS) pair is often combined with carrier sense to address this problem. In the above scenario, node A will send a RTS frame asking for permission to communicate, and node B will not respond as it is participating in another activity. Then node A will back off and collision is avoided.

3. **Overhearing**

Overhearing in network communications means that unwanted packets are received by a node. As wireless link is radio-based link, which is shared between all the nodes in the same range, a packet destined for one node can easily be overheard by other nodes. These nodes will waste energy on receiving packets, and nodes in receiving mode will consume much more energy than in sleep or idle mode.

The best way to avoid overhearing must be turning the radio off when other nodes are transmitting packets, however difficulties exist as data for one node may get lost if going to sleep to fast. Some solutions use a special signal, which is sometimes described as beacon or preamble, to indicate which node is the destination of the next message (packet), and other nodes can immediately switch to sleep mode.

4. **Protocol Overhead**

   Only useful information should be transferred and exchanged among nodes in order to save energy and not to waste bandwidth. However, data besides message content itself are also transferred, and most of the additional information is used by protocols in order to perform appropriate functions. For example, each layer in the network stack needs to add its correspondent header or tailor for the purpose of transmission, reorganization and processing on the other side of communication. Most of this data is compulsory, but some overhead is optional and depends on the extra algorithms used, such as the RTS/CTS scheme introduced in the previous section. In the RTS/CTS scheme, even if only one packet is exchanged between two hosts, an extra pair of RTS/CTS needs to be exchanged before the packet is delivered, which will cause a lot of overhead, as can be seen in Figure 3.11.

5. **Over-emitting**

   The last major source of energy waste is caused by over-emitting, where packets are sent out when the receivers are not prepared, for example when they are still in sleep mode or when they go to sleep too fast. In this scenario, energy consumed in transmission and channel sensing is purely wasted. For some network applications with positive acknowledgement, the packet should be retransmitted later when there is no ACK packet received by the sender. On the other hand, for those connectionless applications, the packet is just lost and data will be missed at the receiver side.

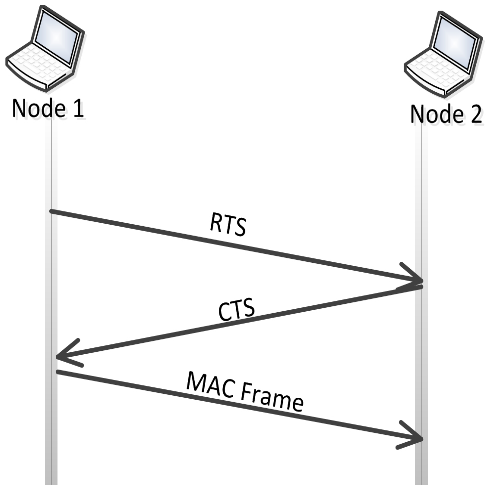


Figure 3.11 Illustration of RTS/CTS protocol overhead

An appropriate protocol will avoid this problem by making sure that packets will not be sent out unless the state of the receiver is in listening mode. For example, some of the MAC protocols will exchange a pair of messages, say, RTS/CTS, between the source and sink of communication in order to make sure packets will arrive at the destination properly. This process is like a handshake between communication parties. Some other protocols will let nodes run by a schedule, and only for a specific interval communication can take place. Every node should wake up and listen to the channel in case of an upcoming packet. Specifying the schedule will guarantee that packets will only be transferred when receivers are awake.

RTS/CTS also addresses the exposed node problem, as shown in Figure 3.12, if the nodes are synchronized. In Figure 3.12, nodes A and B are in the same range, and nodes C and D are in the same range. While the two receivers A and D are out of range of each other, the two transmitters B and C in the middle are in range of each other. In this context, if a transmission between node A and B is taking place, node C cannot communicate with node D, as it concludes after carrier sensing that interference will take place if it starts transmission. However, node D can receive data from node C without interference because it is out of range of node B. On the contrary, with RTS/CTS employed, when a node hears an RTS from a neighboring node, but not the corresponding CTS, it is permitted to transmit data to other neighboring

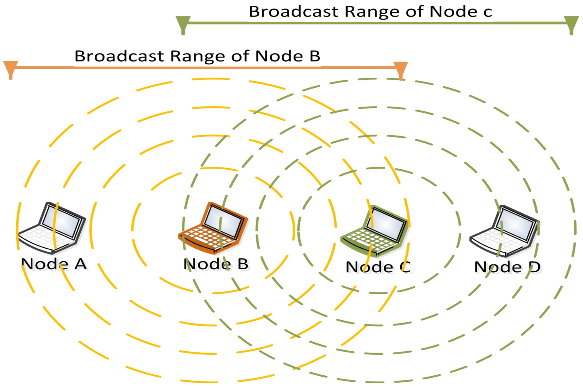


Figure 3.12 Exposed node problem

nodes as it is an exposed node.

Energy efficiency at MAC layer is normally achieved by switching the WNIC card into low energy states (i.e. sleep) when the device does not have any data to send or receive. Studies of power consumption at network interface level in hand-held devices show that the critical factor in saving energy for these devices is to switch the interface off for as long as possible as opposed to reducing the number of packets sent and received [174]. Table 3.4 presents an overview of the energy efficient solutions for MAC protocols discussed in this section.

Table 3.4 Energy Efficient MAC Layer Protocols

Protocol	Type	Description
IEEE 802.11 [27]	Contention-based	Access Point beacons regularly, the mobile station wakes up for each beacon.
S-MAC [171]	Contention-based	Nodes within a same virtual cluster adopt the same schedule, in-channel signaling is employed to avoid overhearing and message passing is used to reduce latency.
TMAC [172]	Contention-based	Similar to S-MAC, but with dynamic duty cycle.
Aloha with Preamble Sampling [175]	Contention-based	Adds preamble in front of packet header to notify the receiver of upcoming traffic.
B-MAC [176]	Contention-based	Similar to Aloha with Preamble Sampling, except that the preamble is not fixed but could be provided as a parameter to upper layers.
PAMAS [177]	Contention-based	Adopt two channels with different frequencies for data and control packets. Energy is saved through turning off the data channel periodically.
TRAMA [178]	TDMA-based	Dynamically assigns time slot according to real time information including the amount of traffic processed by one node and the topology of neighbor nodes.
LMAC [179]	TDMA-based	Divides time slot into control section for message exchange and data section for data transmission. Data transmission starts after the control section and nodes with no data can be switched off.
MMSN [180]	FDMA-based	Allocates different channels to neighboring devices and the least used frequencies are picked to minimize collision and interference.
NOGO-MAC [14]	OFDMA-based	Divides nodes into groups based on the distance to sink. Nodes closer to the sink are allocated higher frequencies.
TFO-MAC [181]	OFDMA, TDMA and FDMA-based	Applies three accessing schemes for frequency and time allocation.
Z-MAC [15]	Hybrid	Slots are assigned but can be contended for transmission if not being used by slot owner.
EQ-MAC [182]	Hybrid	Control messages are transmitted on a contention basis, while data packets are transmitted according to their time slots.
PARMAC [183]	Hybrid	Time is divided into intervals which consist of reservation period when hosts make slot reservation on a contention basis, and contention-free period when data is transmitted according to the reserved schedule.

There are several challenges to be overcome in the design of energy-efficient MAC protocols:

- Increased delay and degradation in quality of service may be introduced by switching off the WNIC card for longer periods of time, especially in the context of delay sensitive multimedia streaming applications.
- Computing the sleep/wakeup schedule may lead to increased processing power requirements and packet overhead. Some protocols look up the communication history and try to predict the next packet arrival.
- Frequent switching between different states leads to energy waste as well due to the fact that a certain amount of energy is consumed during the transition process.
- Complexity is a practical issue when setting up a MAC layer in real network. Due to the nature of a specific network, some complicated protocols can hardly be deployed. For example when in ad-hoc network, where there is no access point to manage the whole traffic, and nodes should be able to be configured by themselves, some TDMA based schemes are not workable.
- Scalability should be considered as well. On the one hand, time is divided into short slots in TDMA based protocols, nodes are often assigned a time slot in each round, and the rounds repeat from time to time. The scalability of this scheme is limited as new devices are difficult to be configured and deployed. On the other hand, with contention based protocols, nodes are free to contend for the resource whenever they have packets to send, there is no existing schedule established, and therefore it is much easier to deploy more wireless hosts after the network is set up.

Energy efficient MAC protocols are classified according to their contention level. Some are based on **contention free infrastructure** where time or frequency is divided and allocated to each mobile station and a schedule is established to prevent overlapping. **Time Division Multiple Access** (TDMA) is one of the most widely used resource allocation policy. As mobile stations have their own time slot and can only transmit or receive packets within this predefined slot, solutions of this type are inherently collision free. However, complexity exists as in most cases there should be some controlling devices which take care of the schedule. Otherwise an algorithm needs to be designed in order to form clusters among mobile hosts themselves. These protocols have limited scalability as schedules needs to be determined before allocation. Other commonly used channel access method include **Frequency Division Multiple Access** (FDMA) which divides access by frequency and **Code Division Multiple Access** (CDMA) which employs spread-spectrum technology and a special coding scheme.

Another category consists of **contention based protocols**. These protocols do not employ any schedule established for accessing the shared medium. Instead, devices can contend for the resource when needed, for example when a device wakes up and tries to send a packet. Collision is introduced to this scheme and energy may be wasted on collision and backing off, as multiple hosts may transmit packets at the same time.

Therefore one common issue of all these contention based protocols is to prevent collision or provide an appropriate back off policy when collision occurs. One obvious advantage of contention based protocols is the scalability of networks, as new mobile devices may easily be added to the network without affecting the complexity of multiplexing schemes required in contention free protocols.

Hybrid protocols using these two schemes in conjunction to define channel access have also been proposed in the literature. The classification of energy efficient MAC protocols is depicted in Figure 3.13.

3.4.2 Contention Based Power Saving Mechanisms

IEEE 802.11 [27] has a built-in power saving mode and is the basis of most contention based power saving schemes. The destination node's Association ID (AID) provides the logical link between the frame and its destination. Each AID is logically connected to frames buffered for the mobile station that is assigned that AID.

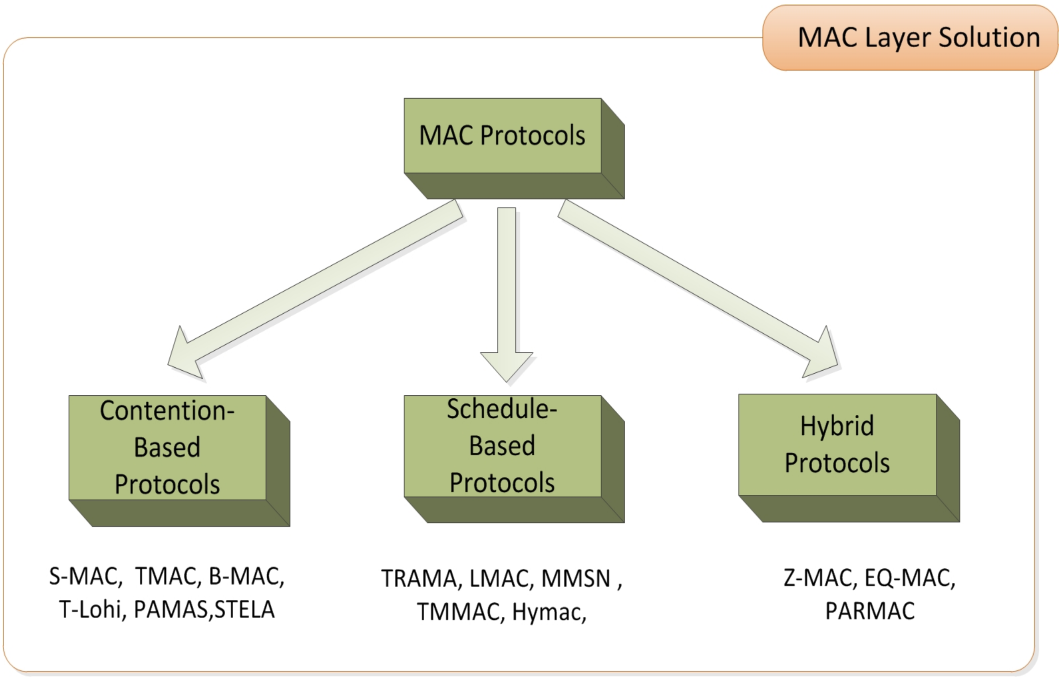

MAC Layer Solution
MAC Protocols
Contention-Based Protocols
Schedule-Based Protocols
Hybrid Protocols
S-MAC, TMAC, B-MAC, T-Lohi, PAMAS,STELA
TRAMA, LMAC, MMSN , TMMAC, Hymac,
Z-MAC, EQ-MAC, PARMAC

Figure 3.13 Classification of energy efficient MAC layer protocols

On the other side, access point buffers packets for sleeping hosts when they are in sleeping mode. The access point periodically assembles a Traffic Indication Map (TIM) and transmits it in Beacon frames to notify the associated mobile hosts. The TIM is a virtual bitmap composed of 2,008 bits; offsets are used so that the access point only needs to transmit a small portion of the virtual bitmap. Therefore it can conserve network capacity when there are only a few stations which have data buffered at the AP. Each bit in the TIM corresponds to a particular AID; setting the bit indicates that the access point has buffered unicast frames for the station with the AID corresponding to the bit position.

On the mobile stations side, WNIC is allowed to sleep for a fixed duration, i.e. one or several rounds of beacon intervals for energy saving. In order to receive buffered data from the AP, the host wakes up regularly and switches to active mode to listen for beacon frames. By examining the TIM, a station can determine if the access point has buffered traffic on its behalf. Once it finds there are packets addressed to it, mobile stations use PS-Poll Control frames. For each poll frame, one buffered frame will be transferred to the receiver. Each frame must be positively acknowledged before it is removed from the

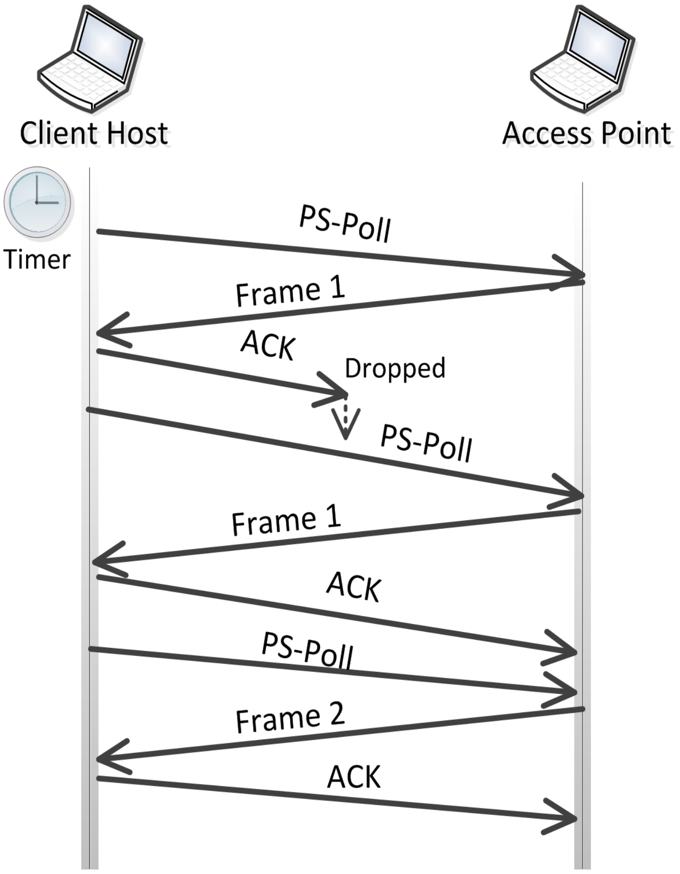


Figure 3.14 Polling process in contention-based PSM

buffer. Positive acknowledgment is required to keep a second, retried PS-Poll from acting as an implicit acknowledgment. The polling process is depicted in Figure 3.14. If multiple frames are buffered for a mobile station, then the More Data bit in the Frame Control field is set to 1. Mobile stations can then issue additional PS-Poll requests to the access point until the More Data bit is set to 0. The whole process ends when all buffered data is retrieved by the wireless device.

Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) which can be deployed in IEEE 802.11 and other IEEE 802.11-based protocols is used to solve the collision problem. Devices wishing to send packets wake up and listen to the channel, if the medium is free, transmission will start immediately, and otherwise they will back off and wait for another chance. If collision occurs as simultaneous transmission of packets happens, the back off procedure is also initiated. This process repeats until packets are transmitted successfully. Power saving under this circumstance can be achieved through preventing packet collision, and turning off the radio while packets transmission is happening among other devices.

The solution presented in [184] not only adopts the same mechanism but also counts the number of contenders in order to determine the back off duration.

Ye et al. propose **Sensor-MAC** (S-MAC) [171] that enables synchronization between mobile hosts so those belonging to the same cluster wake up at the listening period at the same time and a short duty cycle is implemented to guarantee energy efficiency. **Timeout-MAC** (TMAC) [172], similar to S-MAC, makes improvement by employing dynamic duty cycle which means a WNIC is put to sleep mode as soon as it predicts no incoming data.

Aloha with Preamble Sampling [175] and **Berkeley Media Access Control** (B-MAC) [176] utilize the idea of preamble which is added in front of packet header in the physical layer in order to notify the receiver of upcoming traffic. Mobile stations wake up periodically and sample for the incoming message. When a device finds out there is a preamble, it continues listening and waits for the real transmission of packets. If it does not detect a preamble, it turns of its radio until the next preamble. The difference lies in whether the length of preamble is fixed or not. The technique proposed in [185] is based on Markov decision process. It models the power tuning process and calculates the duty cycle adaptively according to its power management policy associated with the Markov process.

The idea of multiple channels with different attributes is adopted in [177] and [186]. **Power Aware Multi-Access protocol with Signalling** (PAMAS) [177] uses two channels with two different radio frequencies for data and control packets respectively. This protocol simply conserves energy by turning the radio off and switching to sleep mode when a device is not transmitting or receiving any data. However, the inefficiency is significant when the traffic load is high, as the switch of WNIC between sleep and active modes is more often, thus wasting a lot of energy. The solution proposed in [186] consists of a primary channel for sending data and control information, and a wakeup channel to wakeup neighbours. Energy is saved as the wakeup channel consumes much less power than the main channel, and it adopts a low duty cycle. A wake up signal is sent from a host to all its one hop neighbours when the queue size reaches a threshold value, and all hosts need to wake up to check if there is data for them. This protocol allows the WNIC to sleep once data communication is finished and wake up at the predicted time calculated via previous traffic patterns.

The feature of traffic burstiness is utilized in some proposals. The mechanism proposed in [173] assumes the burstiness of multimedia traffic and mathematical correlation of the time interval between bursts time series with long memory. It tries to calculate the next interval before arrival of packets based on the previous intervals and corresponding predictor coefficients. A negative bias is used in conjunction to determine the sleep interval of wireless interface. Experimental results show that it outperforms a history-based strategy providing lower data drop rate and lower energy consumption.

3.4.3 Schedule Based Power Saving Mechanism

TRaffic Adaptive MAC protocol (TRAMA) [178] assigns slots according to current traffic information instead of determined beforehand. Information such as the amount of traffic processed by one device and the topology of neighbours will be used together during election process. The elected host will have priority to send packets in a time slot. This protocol renders high channel utility. However, it may cause serious delay due to long period of sleep.

Lightweight Medium Access Protocol (LMAC) [179] proposes a solution where each slot consists of two sections, one is traffic control section, and the other is fixed-length data section. Every host has its own slot, and at the beginning, say, the traffic control section, of its slot, it will broadcast a message containing the destination of packet length if it has any data to send. Real transmission will start after the first section. Other nodes will wake up at the beginning of every slot, listen to the channel to check if there is any data for them. If there is no upcoming event, the WNIC can be switched to the sleep mode.

Traditional frame scheduling mechanisms adopted by most TDMA divide time into three phases for the whole network: uplink phase, downlink phase, and reservation phase and allow all mobile stations transmit, receive data or make reservation in the corresponding phases. Unlike this, the approach proposed in [187] introduces a TDMA scheme where time is divided for each mobile host, so that all activities of one device are grouped into one slot. In this manner the channel capacity decreases due to the time wasted on frequent switch of operating mode of transceivers. However, the battery life is prolonged as a WNIC is allowed to stay in low power mode for longer time periods.

As a typical mobile host is equipped with one singe radio transceiver and is supposed to process smaller packets, FDMA which divides frequency into multiple ranges is not widely supported in wireless networks. In order to solve the problem and better apply FDMA in wireless networks, some mechanisms [180] [188] [189] try to combine it with other schemes in order to suit the requirements.

Multi-frequency Media access control for wireless Sensor Networks (MMSN) proposed in [180] allocates different channels to neighbouring devices in order to avoid collision and minimize interference. Different frequency assignment policies are designed to meet different needs and make sure the least chosen frequencies are picked for the purpose of low interference. It achieves energy efficiency by minimizing overhead packets and introducing less packet loss through low interference. Some other protocols [188],[189] adopt similar FDMA schemes to increase bandwidth utility and energy efficiency.

The solutions introduced in [190] and [191] adopt both TDMA and FDMA or even CDMA to assign frequency and time slots. The main idea is to utilize the combination of different slots and frequencies for data transmission without interference and packet collision, which means if two devices are assigned the same frequency and may interfere with each other, they will be assigned to different slots. The obtained high throughput and short delay provides high quality for real time streaming applications.

Zhang et al. [192] propose an OFDMA-based MAC solution for optical line terminals. Statistical multiplexing gain is achieved through the division of data channel and control channel. Fixed sub-carriers are dedicated for controlling messaging and dynamic sub-carriers are allocated according to real time traffic for data transfer.

In **Node Grouped OFDMA MAC** (NOGO-MAC) [14], nodes are divided into groups based on their distance to the sink in order to maintain high signal-to-noise ratio and low energy consumption. Nodes that are far from the sink are allocated lower frequencies and nodes that are close to the sink use high frequencies. The reason for this approach is the fact propagation loss is more dependent on distance at high frequency. Moreover, an adaptive sub-channel adaptation is employed by the sink to improve the transmission rate.

Uplink-based **TDM with FDM over OFDM MAC** (TFO-MAC) [181], uses FDM to divide bandwidth into sub-bands, and employs OFDM for data transmission. Moreover, TDM is used to divide time into slots, and therefore each node is able to transmit data using different channels in different slots. A greedy algorithm is also proposed to achieve optimal channel allocation, and reduce the transmit power.

3.4.4 Hybrid Power Saving Mechanisms

Despite the collision free nature of schedule based energy efficient MAC layer mechanisms and lower delay and better throughput potential of contention based MAC solutions, these two categories are not absolutely opposite to each other. In fact, some hybrid solutions combine the advantages of both and try to address their shortcomings.

Zebra MAC (Z-MAC) [15] uses a TDMA channel to assign time slots. Although the time schedule is fixed, the novel idea behind this is that a device can not only send packets in its own slot, but also transmit during other hosts' time slots. However, a host has higher priority to send packets in its own time slot than others which borrow its slot. Under this circumstance, when the traffic load is not high, other devices having packets to send will contend for a slot if the slot owner has no data to send, and energy saving can be achieved according to different traffic conditions. The solution presented in [193] is similar to [15] whilst it introduces a local framing pre-schedule slot during which the source broadcasts the destination address of the buffered data so that potential receivers could decide the sleep/wakeup schedule and minimize the energy consumption.

Other approaches such as the ones discussed in [182],[183] divide traffic into two types: control packets and data content and treat them differently with different access methods. **Energy Efficient Hybrid Medium Access Control Scheme for Wireless Sensor Networks with Quality of Service Guarantees** (EQ-MAC) [182] is a priority based solution and mainly consists of two sub-protocols: Classifier MAC (C-MAC) and Channel Access MAC (CA-MAC). C-MAC takes care of classification of data based on its importance, while CA-MAC employs both contention-based mechanism (CSMA) and schedule-based (TDMA) mechanism for medium access control. Short control messages are assigned random access slots while data messages are assigned proper time slots based on their priorities. EQ-MAC achieves energy efficiency and provides high quality of service to multimedia streaming due to its high priority assigned by C-MAC. The protocol proposed in [194] uses a similar idea with the preamble mechanism proposed in [171] to minimize energy costs during idle listening.

Power-Aware Reservation-based Medium Access Control (PARMAC) [183] divides time into equal length intervals, and each interval consists of two parts: reservation period and contention-free period. During the reservation period, hosts try to make slot reservation or cancel reservation on a contention basis. And the reserved slots in the contention-free period are used by corresponding devices for the purpose of data transmission. The protocol is specifically designed for real-time traffic such as multimedia streaming. In PARMAC, only several control packets are used for a whole traffic session instead of frequent exchange of control messages required in some protocols such as IEEE 802.11.

3.5 Cross-Layer Approach to Energy Efficiency

The layered architecture has become the de facto standard for network communication. It is pointed out that the success of Internet is mainly determined by its layered architecture [195]. The benefits and efficiency are due to the simplicity brought by modular-based approach. In the layered protocol stack, each layer is defined to provide specific service to its upper layer and is dependent only on its lower layer. The interaction between adjacent layers is independent of implementation details thus makes the design process much easier. Despite the wide use of layered architecture, there is an ongoing trend of cross-layer approaches towards quality of service and energy saving. Protocols and algorithms have been widely exploited at each layer, but they are designed in an independent way without consideration of inter-relationship among the protocol stack component. However, the proper behaviour of each single layer depends on the functionality of other layers, especially adjacent layers. Energy saved through modification at a single layer could possibly incur energy waste at another layer, for example due to extra control overhead. On the other hand, the dependence between layers infers that inter-layer information exchange and variable shared across the protocol stack can significantly benefit in terms of optimization in the network performance. Therefore cross-layer approaches are designed and implemented to take a more holistic view of the network stack. Moreover, the unique feature of wireless link creates problems that could not be well handled through layered approaches. For example, the congestion control scheme at transport layer could use information from MAC layer to prevent inappropriate detection of congestions.

A typical methodology for cross-layer design takes several steps:

- First, the relationship between layers is identified in terms of both adjacent and non-adjacent layers. The main factors that need to be examined are layer dependencies and data flow through the protocol stack.

- Second, it is required to identify the features of each layer that contributes to the problem to be solved, which in this thesis refers to energy efficiency in multimedia transmissions. The contribution is classified into three types [196]: parameters that only have local effect on individual layer, controllable parameters that have direct effect on the performance of multiple layers and uncontrollable parameters that directly affect multiple layers.

- Third, there is a need to optimize the coupling parameters through modification of parameters at each layer. Instead of focusing on a single perspective, cross-layer approach improves the collective performance of several layers which might even sacrifice performance metrics at one layer.

Although there are many possible benefits of cross-layer approaches, there exist many challenges that should be taken care of in the designing process. Authors in [197] present the potential problems cross-layer design could cause. It is pointed out that fragmentation could be caused by violating the layered structure and there is possibility of inadvertent performance loss due to conflicting cross-layer solutions. Energy-efficient cross-layer approaches for multimedia streaming are divided into four groups: general approaches, traffic shaping, joint routing, sleep scheduling and QoS through packet prioritization.

Table 3.5 presents an overview of the cross-layer energy saving solutions discussed in this section.

3.5.1 General Approaches

Some cross-layer multimedia streaming approaches introduce parameter abstraction methods which absorb different parameters or policies at several layers of the protocol stack and organize a table according to the observations on the quality level or energy consumption. And the table is used in the strategy adaptation at each layer for quality or energy optimization. The general approach towards cross-layer cooperation is depicted in Figure 3.15.

A cross-layer solution is proposed in [198] to optimize user experience for wireless video streaming. It mainly involves three steps: first, parameter abstraction at application layer, data link layer and physical layer are generated respectively, then they are integrated to be optimized and finally the optimal values are distributed to each layer. The abstracted parameters introduced in this mechanism avoid large amount of information that could be gathered at each layer for wireless communication and at the same time hides the technical details. Therefore this mechanism could be implemented in a more general way and deployed in multiple different systems. Experiments on wireless video streaming have been performed and several parameters, i.e. video source rate at application layer, time slot al-

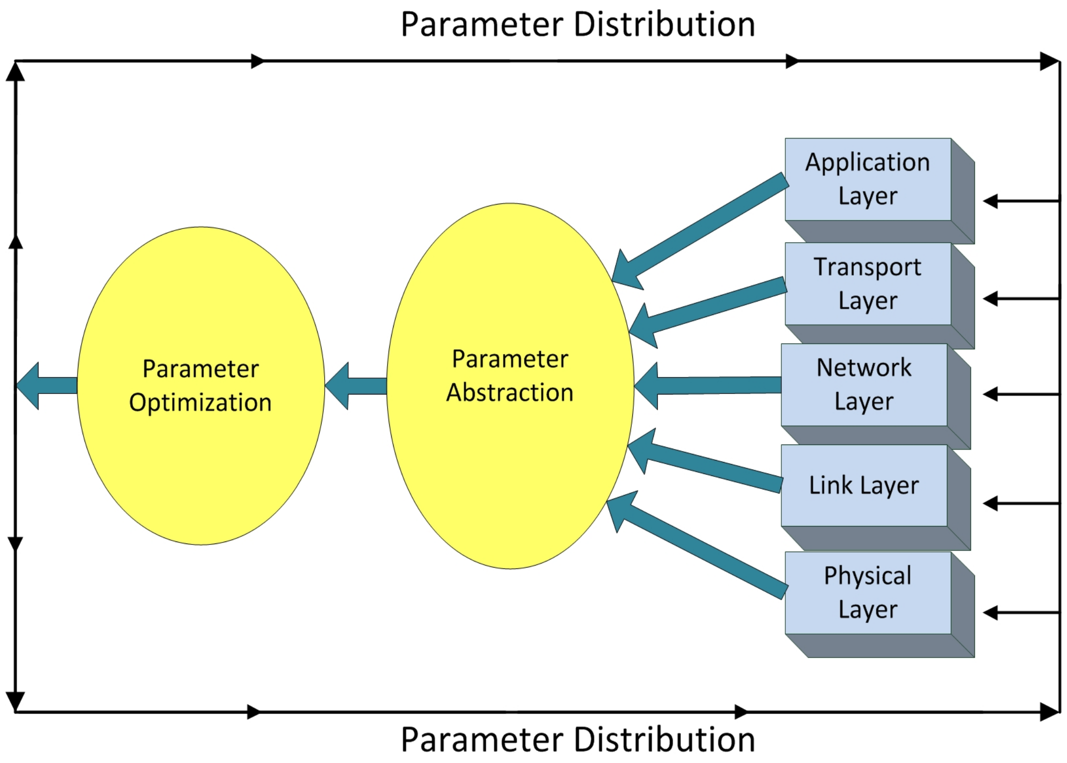


Figure 3.15 Illustration of general cross-layer approach

location at data link layer and modulation scheme at physical layer, were optimized for maximum user satisfaction. The solution in [199] optimizes end-to-end quality of video streaming service in a multi-user environment. It mainly collects status information from application layer, data link layer and physical layer which is called parameter abstraction and optimizes a single objective function through the use of cross-layer parameter tuples. The strategy proposed in [200] evaluates different strategies adopted by different layers in order to enhance robustness and efficiency of scalable video transmissions. The packet loss ratio and throughput efficiency based on a multipath model is derived and used to characterize the video distortion model and estimate the channel condition in terms of SNR, and therefore the protocol is able to select FEC at application layer, retransmission strategy at MAC layer and optimal packet size in order to maximize quality performance. In contrast to most power saving schemes for infrastructure based wireless network, [210] introduces an energy efficient proposal where the decision of when to transmit packets and when to suspend the wireless card is made by the hosts instead of the central access point. There is no beacon scheme in this proposal, and it is up to the mobile host to start a transmission to and to fetch data from the base station. It is also the responsibility of the mobile host to balance the trade-off between energy efficiency and packet delay. Another feature of this protocol is that the decision of switching between modes is application-driven, which means different types of applications will have direct impact on the power management policy. Liu et al. [201] uses mathematical analysis to combine energy efficient routing and sleep scheduling in one optimal framework where an mathematical analysis is provided to give near optimal solution that balances traffic load across the network and periodically sleeps to reduce idle listening time at the same time, instead of fixing one factor and tune the other one, like most other works do. Alicherry et al. [211] proposes to combine efficient channel assignment and routing solution for wireless mesh networks to avoid interference and conserve energy by construct a formula considering the interference constraints, the number of available radios and channels for every router to develop a solution that achieves optimized network throughput and energy efficiency.

3.5.2 Traffic Shaping

Energy saving can be achieved at MAC layer by maximizing sleep duration of WNIC. A major problem is that packets may get lost if the recipient device is in sleeping mode when they arrive, which means a mobile host needs to wake up frequently to check if there is data arriving. Thus it is more energy efficient to form traffic burst deliberately at upper layers and inform a host of the next burst to prolong time spent on low power states. A transparent proxy is normally used to receive the continuous streaming from server and form bursts in its buffer, and a releasing policy determines the time to transmit the next burst to the client. Burstiness not only helps prolong sleeping interval of WNIC, but is also studied by some researchers [212] [213] [214] to exploit the charge recovery effect with traffic patterns, and experiment results show efficient energy use and longer battery lifetime by shaping traffic into bursts. The basic mechanism of traffic shaping is shown in Figure 3.16. Web traffic oriented protocol **Power Aware Web Proxy** (PAWP) [202] schedules incoming traffic into intervals of high and no communication to prolong sleeping interval

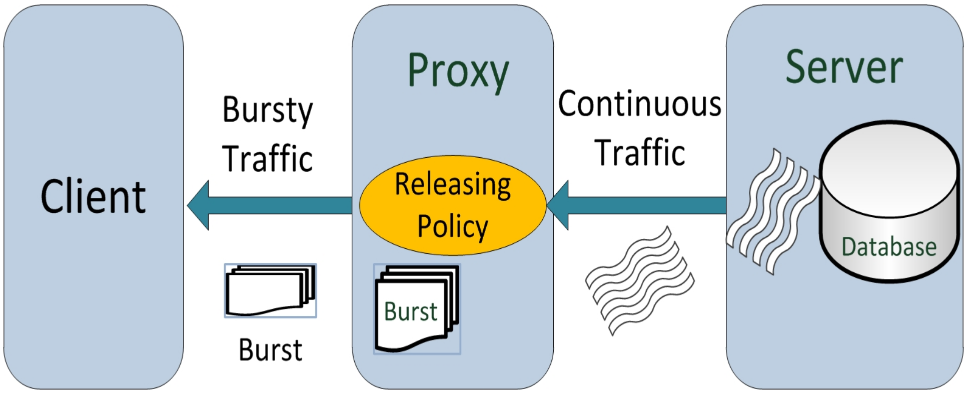


Figure 3.16 Illustration of traffic shaping principle

of WNIC between consecutive data receiving. Data is buffered to form bursts, and the data releasing follows several rules: data is released to the client if the WNIC is in low power state; data should be transmitted within a bound; data transmission gets started if more than a predetermined value of objects are buffered; data forwarding has to be initiated whenever the overhead of switching WNIC to active mode is justified. Catnap [215] is another data oriented protocol that let the proxy decide the best opportunity to start data transmission.

Some other traffic shaping mechanisms are designed specifically for streaming applications, such as **Power Aware Streaming Proxy** (PASP) [216] which works similarly to [202]. The multimedia oriented mechanism in [203], as shown in Figure 3.17, introduces the use of proxies at both server and client sides. Server side proxies or local proxies mainly shape data traffic generated from the server into bursts and exchanges information, for example the next scheduled data burst with the client side proxies. The client side proxy will inform the wireless interface about the future traffic so the interface could switch to sleep mode before the arrival of data in order to save energy. Experimental results show how up to 83 percent of energy is saved for receiving data as long intervals between bursts are guaranteed by the traffic shaping. However, traffic shaping on proxies might mislead the behaviour of stations and incurs unnecessary congestion control. Another proxy-assisted protocol [217] works in a similar way with the support of more streaming formats, and employs another approach where the proxy does not provide smoothing function, instead sends control packets which indicate the arrival time of the next data burst. In [218], a

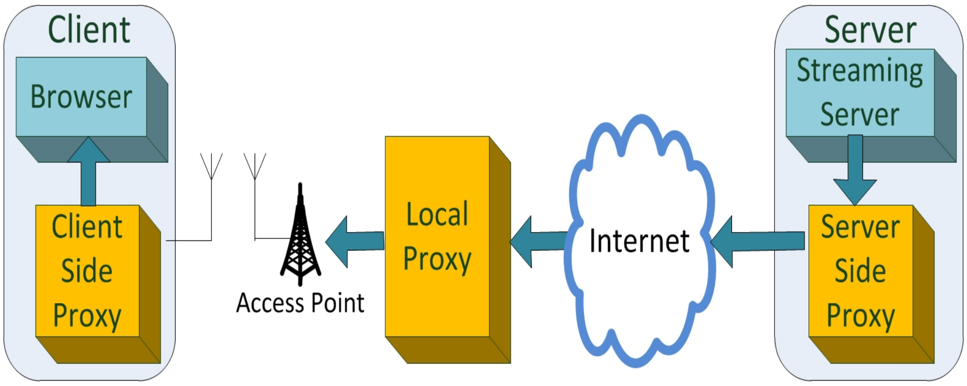


Figure 3.17 Illustration of application-specific network management for energy-aware streaming

proxy between client and server is implemented which is transparent to both sides. Data is buffered at the proxy before being sent to clients and proxies broadcast messages regularly of the next round of traffic schedules. The schedule information includes the start time of transmission to each client and the length of buffered data so corresponding WNIC is able to wake up before the arrival of the next burst. The idea of Multimodal transport layer is explored in [219]. It is designed to adapt its behaviour to different environments with the goal to increase battery lifetime. Traffic is transmitted in forms of bursts through manipulation of ACK messages as it is observed that delayed ACK results in bunching of packets. The feature of slow start and short transfer of traffic which means traffic bursts are separated by the order of the RTT is utilized in the prediction of data arrival time and powering off of WNIC. The solution proposed in [204] employs a traffic shaping algorithm at the proxy which adopts a fast streaming technique. During the fast start period, data is transmitted from the server to the client at a higher rate than the encoding rate until the playout buffer is full. After this the data rate is set back to the encoding rate and at the same time the proxy buffers the data periodically for a period short enough to guarantee the playout buffer is not emptied during playback. Moreover the PSM as proposed in IEEE 802.11 is adopted at the client side MAC layer.

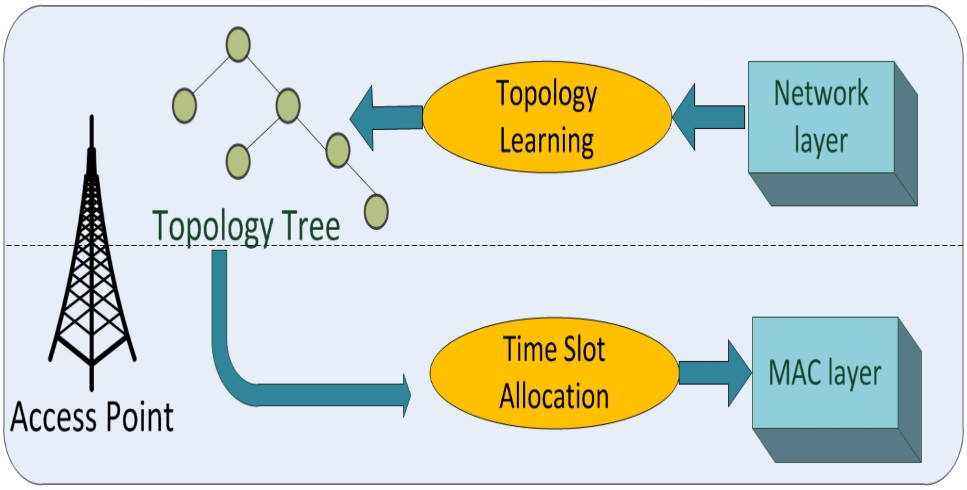


Figure 3.18 PEDAMACS: Joint routing and MAC layer sleep scheduling

3.5.3 Joint Routing and Sleep Scheduling

Another direction in manipulating the behaviour of wireless interface is through the usage of routing information. The main idea behind this is to enable wireless hosts obtain their position in the network or obtain the whole topology especially their parent and child hosts in order to wake up for data transmission with neighbours. At the same time, routing topology is used to study potential interference among wireless hosts and the information can be wisely used to achieve high utilization of bandwidth resource through simultaneously transmission of non-interference hosts.

In some protocols, routing information is learned at a central host to perform scheduling. **Efficient and Delay Aware Medium Access Control** (PEDAMAC) [205] is a TDMA based protocol with an access point determining which mobile host should occupy what time slot. Figure 3.18 illustrates how PEDAMACS learns the topology from the network layer, and schedules the sleep time slots for nodes to construct an energy efficient route. The first phase of this protocol is to construct a routing tree at the access point by broadcasting topology learning packets. Knowing the topology of hosts attached to it, an access point determines when a host will be able to use which slot, and the schedule will be broadcasted to other hosts.

Other researches let mobile hosts have knowledge about their position to perform self-scheduling. A TDMA based protocol [220] assumes a small number of wireless hosts rooted to the sink and topology awareness could be acquired and degree of interference from neighbouring hosts could be known and controlled. In this scheme, time is divided into epochs, and each host has *k* slots in an epoch for the purpose of retransmission. A packet is retransmitted for *k* times at most if not successfully acknowledged by the receiver, thus the delay is controlled below the duration of an epoch. Energy efficiency is achieved by assigning different duty cycles for each host according to their position in predetermined data gathering tree. Routing information is required by each host to get the data gathering tree. **D-MAC** [206] includes a mechanism that builds a data gathering tree which describes the depth of a host in multi-hop paths. An offset is specified as *u*, and any hosts that reside at the depth of *n* in the tree will wake up *du* ahead of the destination, which is the top host of the three, of the path. In this case, every host is assigned its own time slot, and the node at the next hop of a path will be able to wake up after the previous hop host wakes up and sends out the packet. This protocol efficiently saves energy by letting only the hosts on the path wake up. It also helps reduce packet delay by letting hosts waking up sequentially. However, collisions may still occur when different branches of the same host try to send packets at the same time, although a back off scheme is included in this protocol. In [221], each host wakes up when it is time for it to sense the environment and when it expects packets from neighbouring hosts, which means it has to route the packets to the next host. To be specific, if there is a route starting from host *a* to host *c* via host *b*. host *a* samples and transmits a packet, both of which process take 5ms, then host *b* has to wake up at 10ms when host a starts to transmit. Wu et al. [222] proposes to organize a tree of hosts, which allows data aggregation to perform along the tree structure in energy efficient way than normal tree structure. Furthermore, from the viewpoint of MAC layer, wireless nodes consume different energy in different radio states and state transition. It makes sure every host has to wake up only two times in one scheduling period for receiving data from the children and sending data to the parent in order to reduce the frequency of state transition which causes energy waste. This enhanced wake-up scheduling in MAC with the energy efficient data aggregation tree achieves better energy efficiency.

Some joint protocols utilize information from network layer in MAC protocols, while others are wise routing protocols which adjust routes dynamically according to the schedules of hosts to avoid sleeping hosts and decrease end-to-end delay. Bernardos et al. [223] introduces a TDMA communication scheme allowing neighbour hosts cooperatively find required communication time slots and avoid redundant sending of messages. In this manner, mobile hosts will schedule their wake up procedure according to predetermined timetable and transmit packets to the neighbour only if both parties are awake allowing self to sleep for the rest of time. With the information of hosts wake up schedule, this routing solution carefully selects the path from the source to the destination that includes as many hops as possible in one time frame so mobile hosts do not have to wait too long before it forwards it to the next hop in order to achieve short delays.

3.5.4 QoS through Packet Prioritization

Although quality experience and energy efficiency are not necessarily opposite concepts, normally there is normally a trade-off between them. High quality of multimedia streaming requires reliable data transmission with low delay, low jitter and smooth playback, which implies huge amount of data transmission, high bandwidth and in time response to packet loss. On the contrary, energy conservation is achieved through long period of inactivity, ignorance of decrease in performance, or preference of selective hosts, which leads to delayed transmission and drop in throughput. Meanwhile, different applications ask for varied type of services for example web content is tolerant of delay thus could be served with best effort and multimedia streaming requires real time transmission. In order to utilize limited network resource and balance QoS among applications, researchers have explored the idea of assigning different priorities to packets according to application type, channel condition, energy level and take into account different priorities when making decision on channel contention.

Protocol [207] integrates application layer, MAC layer and physical layer in order to maximize quality of service. It classifies users to different classes based on both the network condition and the associated QoS. And different transmission format, i.e. channel coding schemes, power control patterns, and priorities are scheduled among users at both uplink and downlink by MAC. Different QoS states are defined by user priority/pricing and different characteristics of data traffic. The prioritization rules in [208] are based on both their application and MAC layer information, where application type and along with the number of hops that a packet has gone through determine the level of emergency. An inter-host scheduling mechanism tries to minimize collision and idle listening in order to achieve energy efficiency by dividing a frame, which represents a round of RTS-CTS-DATA-ACK message exchange into contention period (CP) and transmission period (TP) and allocates these period, to packets according to their priorities. [224] adjusts the contention window according to traffic types and real-time information. Real time information specifically refers to packet collision rate which is gathered at each host so that packets could be sent in time if channel is expected to be idle and kept in the back-off procedure if channel is busy. Traffic type is differentiated so that real time traffic such as multimedia streaming packets is served with higher reliability and quality. The solution proposed in [225] mainly employs a priority function which is based on frames types, channel conditions, buffer space and multiplexing gain. To be more specific, MPEG video data is divided into three categories: I-frames (intra-coded frames), P-frames (predicted frames) and B-frames (bidirectional frames). Frames belonging to different categories are assigned different priorities. Besides frame time, video streams are given higher priorities if they have better channel conditions. Moreover, users with more buffers that are empty are given preference and higher priorities are given to those streams that have started transmissions to gain multiplexing. Protocol [209] guarantees certain QoS levels in terms of data throughput, packet error/loss rate and average delay and at the same time provides efficient bandwidth utilization by introducing a scheduler which takes consideration of estimated channel condition at physical layer and the queue status at MAC layer. NExt, decision on the number of time slots is made to each user according to the type of service they belong to: QoS-guaranteed or best-effort. Users of the former type are assigned a reserved amount of bandwidth while the best-effort users are served with best effort.

3.6 Summary

In this chapter, some of the most important solutions implemented at each layer of the protocol stack are outlined and discussed. Moreover, the state-of-art review of solutions proposed to contribute towards achieving energy conservation and improving quality of service on each layer of the protocol stack is presented followed by discussions about some important cross-layer solutions. The benefits and drawbacks of existing solutions are analyzed and compared, the results of which show the need for a cross-layer solution and adaptive adjustment of the behaviour of wireless interface of devices through cooperation of different layers in order to maximize the energy saving without compromising significantly the user experience.

Data compression techniques are used in some application layer solutions which squeeze large chunks of data into smaller pieces of information, reducing also the energy consumed on data transmission. Adaptive control is applied in some works which adjusts the behavior of applications to the environment, such as the remaining battery power level. In order to save access time, partial caching techniques are used in many existing solutions, as well.

Energy efficiency can be significantly improved if the number of retransmissions required for packet transmission is reduced. Moreover, the elimination of channel congestion also saves energy as data can be sent out as quickly as possible. Transport layer solutions are categorized into reliability-oriented, congestion-oriented and hybrid schemes. The first category is then categorized according to the flow direction: upstream and downstream. Congestion control transport layer protocols are further divided into centralized solutions and distributed solution. Moreover, the hybrid schemes provide control for both reliability and congestion.

In order to conserve power, the essential idea of MAC layer schemes is to reduce the time spent on idle listening and state switch and to put the WNIC into sleeping mode as long as possible. Solutions at this layer are divided into contention-based approaches, schedule-based approaches and hybrid solutions.

Some cross-layer solutions are proposed to maximize energy efficiency through cooperation between modules reside at different layers. Parameter abstraction method can be used to abstract parameters from each module and provide a table to look up for the best results. Traffic shaping is employed at application layer or transport layer to help prolong the sleeping interval of WNIC at MAC layer. Other methods such as packet prioritization or joint routing and sleep scheduling also benefit from cross-layer cooperation.

However, there is no such solution which fully utilizes the bursty nature of network data traffic in the wise scheduling of WNIC. Moreover, it is missing in the literature a solution that provides maximum energy efficiency through cooperation between MAC layer scheduling and higher layer data shaping without compromising delivery performance. In this context, we propose in this work three solutions which exploits the MAC layer WNIC scheduling or application layer data shaping and adaptively controls the WNIC in an effective way to provide the balance between energy efficiency and high QoS levels.

Table 3.5 Energy Efficient Cross-Layer Approaches

Protocol	Type	Layers	Description
Application-driven cross-layer optimization [198]	Parameter abstraction	Application layer, data link layer, physical layer	Provides a parameter abstraction and distribution mechanism.
Cross layer optimization [199]	Parameter abstraction	Application layer, data link layer, physical layer	Optimizes a single objective function through the use of cross-layer parameter tuples and parameter abstraction.
Adaptive cross-layer protection strategies [200]	Parameter abstraction	Application layer, MAC layer	Evaluates different strategies and applies optimal parameter configuration at each layer.
Joint routing and sleep scheduling [201]	General cross-layer corporation	Network layer, MAC layer	Balances traffic load for optimal routing and adjusts sleep schedule using mathematical analysis.
PAWP [202]	Traffic shaping	Application layer, MAC layer	Schedules traffic into bursts, which are released according to MAC layer information.
Application-specific network management [203]	Traffic shaping	Application layer, MAC layer	Shapes traffic into bursts and informs clients the arrival of data.
Fast start-enabled data buffering [204]	Traffic shaping	Application layer, MAC layer	Utilizes fast start to smooth out the playout buffer .
PEDAMACS [205]	Joint Routing and Sleep Scheduling	Network layer, MAC layer	A routing tree is built so that sleep schedule can be obtained at the sender.
D-MAC [206]	Joint Routing and Sleep Scheduling	Network layer, MAC layer	Builds a data gathering tree for time slot assignment.
Cross-layer design for QoS wireless communications [207]	QoS through packet prioritisation	Application layer, MAC layer, physical layer	Differentiate nodes by network condition and QoS, and associates different transmission format according to priorities.
Energy-efficient QoS-aware MAC control [208]	QoS through packet prioritisation	Application layer, MAC layer	Application type and MAC layer information, i.e. the number of hops that a packet has gone through are used to determine the level of emergency.
Cross-layer scheduling with QoS guarantees [209]	QoS through packet prioritisation	MAC layer, physical layer	Utilizes channel condition at physical layer and the queue status at MAC layer to adjust appropriate QoS.

Chapter 4

Proposed System Architecture and Algorithms

In this chapter, the system architecture and details of the proposed algorithms are presented. Two architecture options are described first, highlighting their benefits and drawbacks, and the reason behind selecting an infrastructure-based network for the proposed algorithms is explained. Data flow and block-level architectures at both client and server sides are provided. Next, detailed descriptions of the proposed solutions are provided: the energy efficiency oriented MAC layer solution– static version of Slow sTart Exponential and Linear Algorithm (STELA), the balanced performance oriented cross-layer solution– dynamic STELA, which employs the Packet/ApplicaTion manager (PAT) at application layer to interact with MAC layer to provide in time wakeup of WNIC, and the quality-oriented cross-layer solution Q-PASTE which adopts an extra traffic shaping scheme as part of PAT for the purpose of maximizing energy efficiency for multimedia delivery in WLAN with dynamic STELA employed at MAC layer.

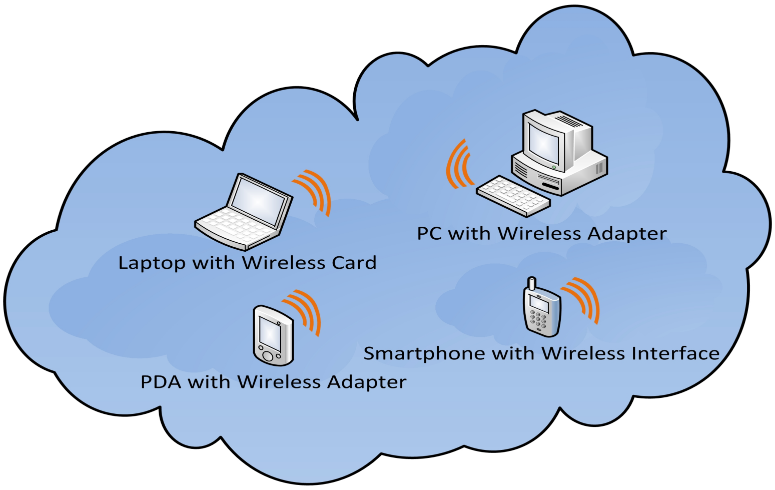

PC with Wireless Adapter
Laptop with Wireless Card
Smartphone with Wireless Interface
PDA with Wireless Adapter

Figure 4.1 Wireless LAN in ad-hoc mode

4.1 Network Architecture

Generally speaking, a wireless network is composed of *access points* (AP) (i.e. routers), and *clients* which are normally referred to as mobile stations in the context of wireless communications. The Base Service Set (BSS) is composed of a group of stations which are able to communicate with each other. Based on the existence or not of the access point, BSS can have a centralized architecture, or a distributed architecture.

4.1.1 Distributed Network Architecture

A distributed network architecture, also known as Independent BSS (IBSS) is an ad-hoc network with no access point; all data is transmitted without the control of a central node. Ad-hoc refers to the fact that there is no existing infrastructure of the network, no planned router or access point is provided, and the wireless nodes in the system form a network by themselves and may even participate in routing and forwarding packets. This type of infrastructure free architecture is typically used for a variety of applications including in the areas of military, scientific research and etc. The major advantage is the possibility to collect and pass information fast, it has quick deployment and minimal configuration. The structure of IBSS is illustrated in Figure 4.1.

A special type of wireless ad-hoc networks are Wireless Sensor Networks (WSN). WSN involves sensor devices embedded in wireless nodes, and it is often deployed in special areas, where people can hardly go and collect information themselves, for example, data collected by acoustic devices under water, or data detected in hazardous environments. In sensor networks, most of the nodes are battery operated and they are not easy to be recharged or changed. When a battery is dead, the node will not be able to function; what is worse is that the performance of whole network may be affected as all nodes are highly inter-liked with each other, a node will not only receive and transmit its own packets, but also work as router connecting other devices. Therefore, energy management is critical in this situation. However, this type of wireless networks without the coordination of a central station is outside the scope of this thesis.

4.1.2 Centralized Network Architecture

In contrast to the wireless networks based on a distributed architecture, centralized wireless networks are infrastructure-based, which means all packets within a BSS are received and forwarded via the access point. In traditional wired network, routers are devices used as interconnection between different computer networks, and manage the packet transfer among these networks. However, as an access point, the router has another functionality added and it can be configured as a central traffic coordinator which works mainly at MAC layer of a network stack. The structure of a centralized BSS is shown in Figure 4.2.

In an infrastructure-based wireless network, the mobile stations do not communicate directly with each other. Instead, all packets transmitted are received by the access point and then forwarded to their destination. Moreover, the access point allocates resources to each participant within its service set as part of medium access control (MAC) function. Often other functions are also deployed such as power saving (e.g. the power saving mechanism (PSM) in IEEE 802.11) and prioritization (e.g. class-of-service based approach in IEEE 802.11e). In these situations, the existence of an access point is essential for the implementation of all these functions.

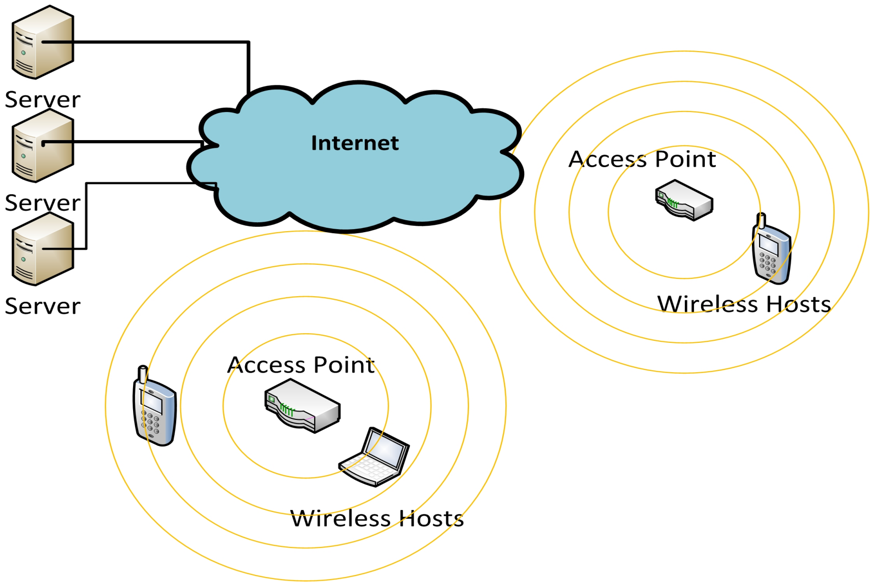


Figure 4.2 Wireless LAN in infrastructure mode

In particular, infrastructure mode is the most suitable for MAC-layer power saving schemes due to its centralized coordination. Due to the existence of the access point, it is less likely for packets to get lost when the recipients are in sleep mode, and wireless devices are able to fetch buffered data whenever they are ready for receiving. The problem of synchronization which exists in ad-hoc networks is avoided.

4.1.3 Power Saving Infrastructure-based Network Architecture

Infrastructure-based network architecture provides an efficient and convenient way of deploying power saving schemes for wireless devices. The main reason is that in order to save energy, the most common practice is to switch the WNIC into lower power consumption sleeping mode, during which period the packets addressed to the mobile host cannot be received. An ad-hoc based network requires accurate synchronization between the sender and the receiver, so that both sides are awake when the data transmission takes place, otherwise data will get seriously delayed or even lost. On the other hand, infrastructure-based network allows the AP function as intermediate node which buffers the data when the receiver is in sleeping mode, and therefore there is no requirement for synchronization and the sender and receiver do not need to know the state of each other.

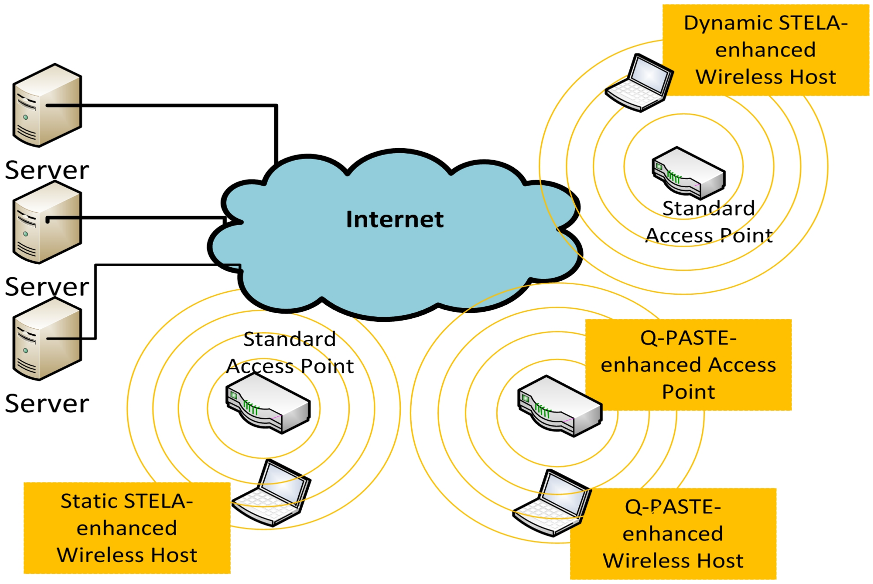

Dynamic STELA-enhanced Wireless Host
Server
Server
Server
Internet
Standard Access Point
Standard Access Point
Q-PASTE-enhanced Access Point
Static STELA-enhanced Wireless Host
Q-PASTE-enhanced Wireless Host

Figure 4.3 Power saving infrastructure-based network architecture

The proposed solutions including static STELA, dynamic STELA and Q-PASTE are also targeted at infrastructure-based networks, as shown in Figure 4.3. Both static STELA and dynamic STELA are implemented at the client side, while Q-PASTE is deployed on both the AP and the client side. The reason behind choosing infrastructure-based networks is that the proposed power saving schemes require all packets be temporarily stored in the central station while the destination devices are put on sleep mode and released once the radio transceiver is on and able to receive any buffered data. The AP beacons regularly to the associated stations, and the sleeping intervals of wireless devices are recorded in terms of the beacon intervals, namely sleeping window size.

4.2 Assumptions

The proposed solutions are based on several assumptions: burstiness of data in network traffic, high probability of regularity in data traffic, and infrastructure-based network topology.

4.2.1 Bursty Data Traffic

Network traffic has been proved to exhibit relative high burstiness [16] [17]. Bursty traffic refers to the characteristics of packet exchange with short inter-arrival duration within groups of packets and relatively longer gaps between these groups. As TCP and UDP are the most widely used transport layer protocols, next we highlight several reasons that cause burstiness of both UDP and TCP traffic [226].

- **Bursty applications**. Some applications exhibit more bursty traffic patterns which transmit packets sporadically, such as FTP file transfer and video streaming. [227].
- **UDP message segmentation**. Message segmentation is used by UDP to chop a message that is larger than the paths Maximum Transmission Unit (MTU) into smaller chunks or by the operating system to segment a message into multiple IP packets. In this case, a flow of message instead of a whole piece of information is transmitted.
- **TCP slow start**. During the slow start phase, the congestion window is increased by one Maximum Segment Size (MSS) for every new Acknowledgement (ACK) leading to fast increase. The data transmission burst length could double every Round Trip Time (RTT) especially when the receiver does not use delayed-ACK.
- **Self-clocking**. Self-clocking means that TCP should send a new packet when it receives a new ACK [121]. In the case of Delayed-ACK, a pair of packets is sent every time an ACK is received, as an ACK is generated for every second received packet. More importantly, the feature of self-clocking indicates that a sender has no direct control on the timing of packets departure, instead, packet departures are initiated by the receiving ACK events. Therefore it is possible that under certain conditions a sender transmits large number of packets at a time.
- **ACK compression**. Due to congestion of traffic in the reverse path of a TCP flow, successive ACKs might arrive at the sender at the same time.
- **Fast retransmission**. Fast retransmission as a recovery of segment loss may lead

to rapid increase of the ACK number, and consequently the sender can send large amounts of data back-to-back.

- **Useless congestion window increase**. When protocols that include control message exchange at the beginning of the content data transmission, the congestion window increases rapidly without being used by the sender to send useful data. Therefore, the sender will send a long burst of data to the network when the content transmission begins.

- **Idle restart timer bug**. Although the TCP congestion window is supposed to return to the initial value after a long time without any data transmission at the sender, some operating systems do not support this feature properly [228]. Consequently when new data transmission starts, a burst of traffic will flow into the network.

- **ACK reordering**. Reordering of ACKs is likely to trigger packet bursts at the sender.

Traffic burstiness is exploited by STELA to balance the trade-off between energy saving and quality of service levels. The adaptive scheme of STELA attempts to turn off the radio transceiver during the long interval between packet bursts and switch on the radio transceiver during each burst.

4.2.2 Regular Traffic Arrival Pattern

Extensive studies have been made on the arrival patterns of network traffic, results of which have shown relative high levels of regularity [16] [19]. Large levels of regularity is exhibited in the data traffic in terms of packet length, inter-packet arrival and bit rate. In our study, the assumption of regular traffic pattern is established on the fact that the data arrival interval is more or less consistent for the duration of the same data flow as it is highly likely that the application specification and system configuration remain unchanged during the application running interval.

4.3 Static STELA

This research proposes static STELA, whose goal is to prolong the battery life for mobile devices. Static STELA was introduced at the MAC layer to reduce power consumption at the client host by providing wise scheduling of Wireless Network Interface Card (WNIC), one of the most important energy consumers, especially for multimedia streaming applications which tend to be long running, on battery-constrained mobile devices [229] [230]. In static STELA, the WNIC state can be wisely switched from time to time so that the time spent in the sleeping mode is maximized and the number of state switches is minimized. Four states of WNIC are considered: sleeping, data transmitting, data receiving and mode switching.

4.3.1 Static STELA Architecture

Figure 4.4 illustrates the proposed static STELA solution, located within the TCP/IP protocol stack model. Static STELA is a MAC layer solution which controls in an innovative way the sleep/wakeup schedule of WNIC. The aim of STELA is to adaptively adjust the size of the sleeping window of the wireless interface in order to conserve power as much as possible while at the same time to maintain high QoS levels.

This infrastructure-based system which considers wireless multimedia content delivery is composed of three major components: multimedia server, access point, and one or multiple client side hosts. The server responds to client requests and sends corresponding media content over UDP to clients. UDP, instead of TCP, is used as the transport layer protocol to reduce overhead. The response data could be of different formats as tested with different intervals between data packets. The server requires no modification for static STELA deployment.

The AP is another important device which enables deployment of power saving schemes for most infrastructure based energy saving solutions. It has three major contributions in terms of energy conservation: data storage, regular beaconing and data forwarding. When

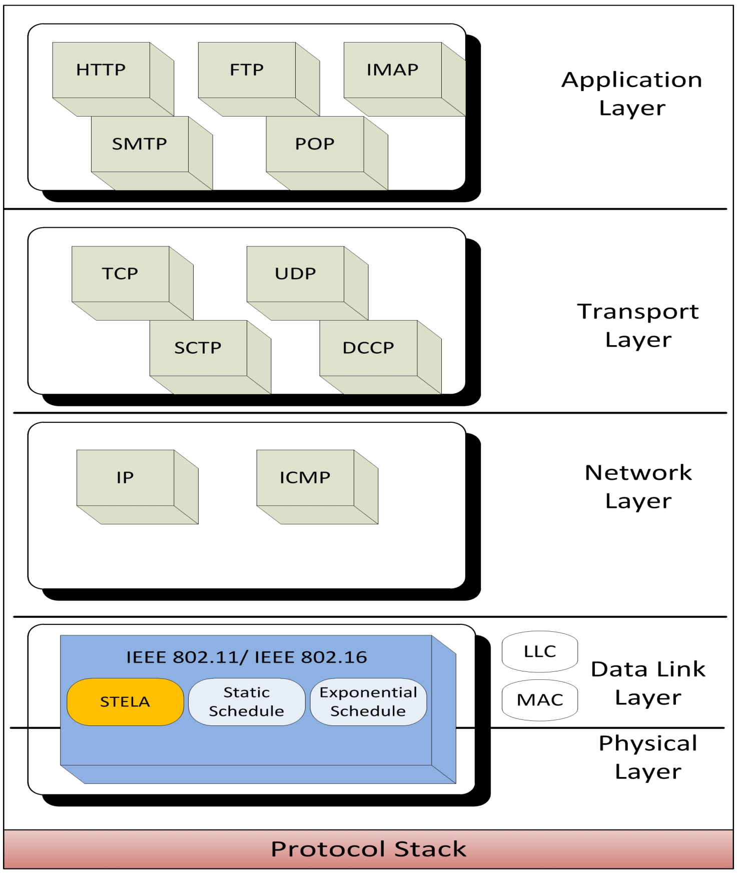


Figure 4.4 Static STELA architecture overview

packets are received by the access point, they are stored in the AP buffer before being forwarded to the recipient. The access point beacons regularly to broadcast information about which connected devices have buffered information. When a device indicates that it is ready to receive data, the access point will forward all the buffered data to the device. The architecture of server and access point is depicted in Figure 4.5.

The client side of static STELA is composed of one or more wireless devices that are able to send and receive data. Static STELA deployment requires modifications on the client side to be performed in terms of the power saving scheme at the MAC protocol. A power saving scheme works with the WNIC and decides the time when the radio transceiver should turn off and when it should be turned on. The goal is to allow the WNIC sleep longer, and also to avoid packets being delayed too much or even lost due to the sleep/wakeup pattern. The architecture of the proposed static STELA at the client device

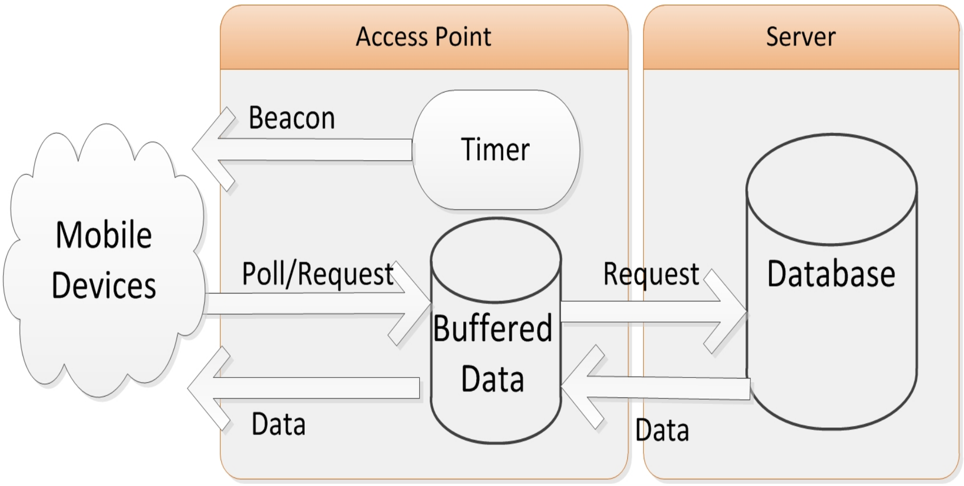


Figure 4.5 Server side and access point architecture of static STELA

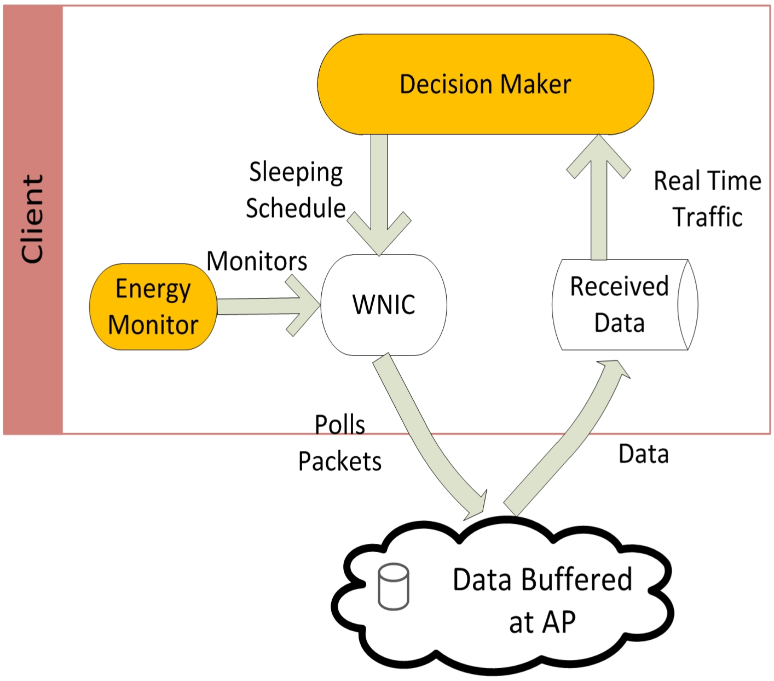


Figure 4.6 Client side architecture of static STELA

is shown in Figure 4.6. Static STELA is composed of three phases: slow start phase, exponential increase phase and linear increase phase which decides WNIC'S sleeping window size. There are mainly two components that the functional static STELA consists of: the *Decision Maker* and the *Energy Monitor*.

The *Decision Maker* hosts the main algorithm that controls the network interface. The input of the module is the current phase that the WNIC is in and the perceived traffic

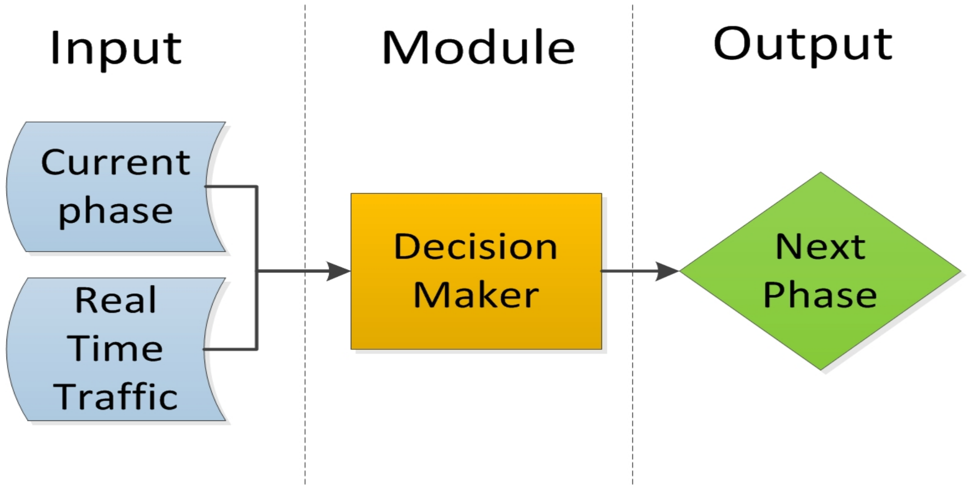


Figure 4.7 Decision maker in static STELA

conditions, while the output is the next phase the WNIC should choose. This module in fact decides whether the WNIC stays in the current phase or moves on to the next phase, and which one that is, as shown in Figure 4.7. When data is received by the device, the Decision Maker will analyze the real time traffic and decide the following status of the WNIC: to wait for another round of beacon or switch to sleep mode. Moreover, if a decision on sleeping is made, the decision maker will also determine the appropriate sleeping schedule the WNIC should follow, which means the time period the user device sleeps for and ignores all traffic.

The *Energy Monitor* computes the total energy consumption of WNIC. The output of Energy Monitor is the energy consumed by the WNIC during the testing period, while the input of the module is obtained through monitoring every state change of the WNIC, as shown in 4.8. For every state switch, the Energy Monitor calculates the duration spent in the last state and records the current state.

4.3.2 Static STELA Algorithm

The main operational principle of STELA is illustrated in Figure 4.9. As noted in the 802.11 and 802.16 standards, the access point beacons regularly. Static STELA adjusts the sleeping window of the client wireless network (radio) interfaces based on the observed real time traffic. Static STELA mechanism consists of three phases: slow start, exponential

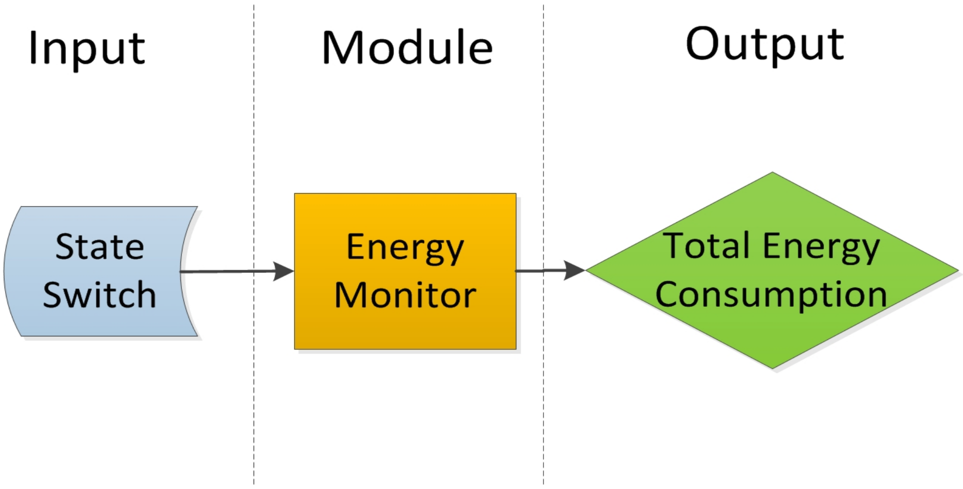


Figure 4.8 Energy monitor in static STELA

and linear increase of sleeping window. The *slow start phase* starts when a node receives one packet from the access point or a request to send data is made by the node, and it ends when no packet is detected during one beacon interval. During this stage, the sleeping window will be kept at one beacon period. The next phase is the *exponential increase* of the sleeping window, which will double the sleeping window every time the wireless interface wakes up unless the wireless node receives packets and goes back to the slow start phase. During this phase of the algorithm, sleeping window grows fast until it reaches a predetermined threshold value. Finally the *linear increase phase* is introduced after the threshold value is surpassed. In this mode, the listening period will increase by one beacon period each time the node wakes up until a specified maximum value is reached.

According to the bursty nature of traffic, response packets from the server will only arrive and probably will arrive soon after a request is made. Thus the slow start phase which requires the WNIC wake up after one beacon interval will lead to a quick response to arriving packets. Moreover, response packets are likely to form a continuous stream which can be processed by the client node during one wakeup period as the sleeping window does not grow when one packet is received. If an expected packet does not arrive in time, the second phase gets started with sleeping window growing exponentially. In this manner, energy is not wasted even for traffic patterns like Constant Bit Rate (CBR) where packets are transmitted regularly at constant bit rate. To be more specific, when a packet arrives,

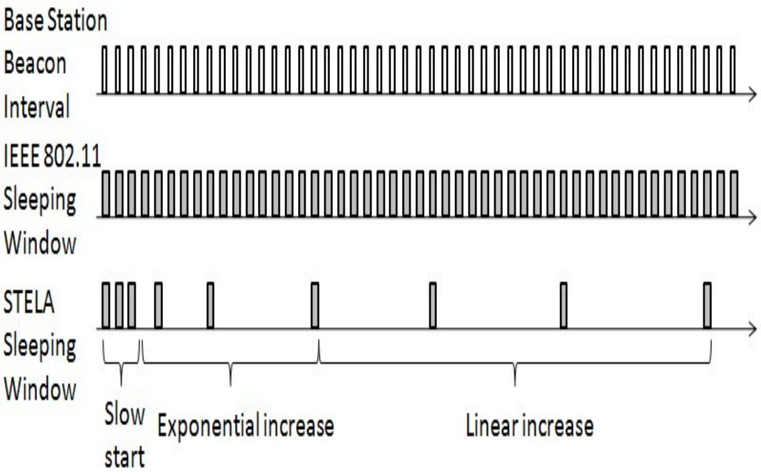


Figure 4.9 Illustration of static STELAs energy saving approach

the slow start process is carried on, and the exponential increase will begin after no more packets are received within one beacon period only.

Normally, a specified maximum value for the sleeping window is set beforehand. A small value would lead to frequent waking ups, while a large value will lead to long packet delays. Using a predetermined threshold value does not take into account the real time traffic. In STELA, a threshold is set for the exponential increase function only. When the threshold is reached, the sleeping window keeps increasing by one each time the node wakes up in order to avoid a fast growing pattern. Thus if there is little traffic, wireless nodes can sleep longer, and they can go back to the slow start phase quickly if a packet arrives, as the linear increase will lead to slow growth of the sleep duration. Above all, three phases of increase of sleeping window will generally save power without increasing packet delays too much. The three phase algorithm of STELA is described in Algorithm 1 using pseudo-code.

4.3.2.1 The Slow Start Phase

The slow start phase is the first stage of STELA and refers to the scenario where the wireless device wakes up regularly to listen to every beacon. It is adopted when the algorithm predicts that some data packets will be arriving within the next beacon interval in

Algorithm 1: Static STELA

```
Input: Threshold value $W_{thre}$
Initialize the sleeping window of the WNIC, $W_s$= 1;
Initialize the sleeping intervals counter $W_l$= 1;
for every beacon interval do
    if $W_l$ = 1 then
        wake up, listen to beacon;
        if buffered data is detected then
            Retrieve data;
            $W_l$ = 1;
            $W_s$ = 1;
        else
            if $W_s < W_{thre}$ then
                $W_l$ = $W_s$ = 2*$W_s$;
            else
                $W_l$ = $W_s$ = $W_s$++;
    else
        $W_l$–;
```

order to reduce packet delays. This phase contributes in several aspects. On one hand, most protocols without considering any characteristics in terms of traffic burstiness turn off the transceiver radio immediately after finishing transmission. In this circumstance, packets that could arrive at the client within a short period of time will instead be stored in the access point's buffer and will be transferred after the node wakes up next time. Unnecessary delay is therefore introduced. On the other hand, some other protocols, e.g. BSD [231], introduce a method of reducing the delay by not going to sleep at all after sending out a request, and waiting instead for some time before going to sleep. However, these protocols focus on delay reduction without saving any energy. This approach is practical in HTTP traffic only, and may waste energy for other types of traffic without reducing the delay (for example for CBR traffic, where the server constantly generates traffic with the same rate).

This phase is triggered by either of the two events: sending data request and receiving packets. **Sending request** refers to the mobile host sending a request to a remote server asking for data content. This initiates the delivery of content including texts, images, audio, video, etc. On one hand, it is highly likely that one or multiple pieces of content will be generated by the server and transmitted to the requesting node via the network. Furthermore, any piece of data content that is longer than the Maximum Transmission Unit (MTU) will be chopped into a group of segments prior to the transfer of data. This is due to the packetized communication process in today's communication systems. Therefore a request by the client normally determines a series of data packets to be transmitted in the reverse direction, which can be seen in Figure 4.10.

On the other hand, Round Trip Time (RTT) which is the time between a client sending a request and the server response being received over the network. This has been observed to be very short according to [231] due to the fast deployment of Content Distribution Networks (CDNs), and increased bandwidth of the latest network infrastructure. For example, measurement results have shown that the round trip time from both east coast and west coast of the U.S. to some popular sites such as Google, Yahoo, CNN, etc. is less than 30 ms most of the time. Additionally there are many other applications where data transfer happens between a client and a local server, and RTT is only a few milliseconds.

In this context, the RTT is much shorter than the typical beaconing interval of the access point, i.e. 102.4 ms. Data segmentation and fast RTT together help support the assumption that a series of data packets arrive at the mobile host within one beacon interval, and therefore the wireless interface should be turned on at the beginning of each interval.

Receiving packets refers to the event that the mobile host receives a data packet from the server after sleeping for more than one beacon interval. According to the bursty nature of traffic, packets often arrive in the form of bursts which implies that a packet received by the client device will be followed by multiple packets within the same burst. Under this assumption, waking up at the time when the next beacon is generated leads to quick response to packet arrival and shorter packet delays.

The end of the slow start phase is signalled by the absence of incoming packets when the transceiver radio turns on and listens to the beacon. Based on the assumption of that packet arrive in bursts, the absence of packets indicates that the last packet of a traffic burst was received, and very likely there are no more packet incoming for the immediate a short

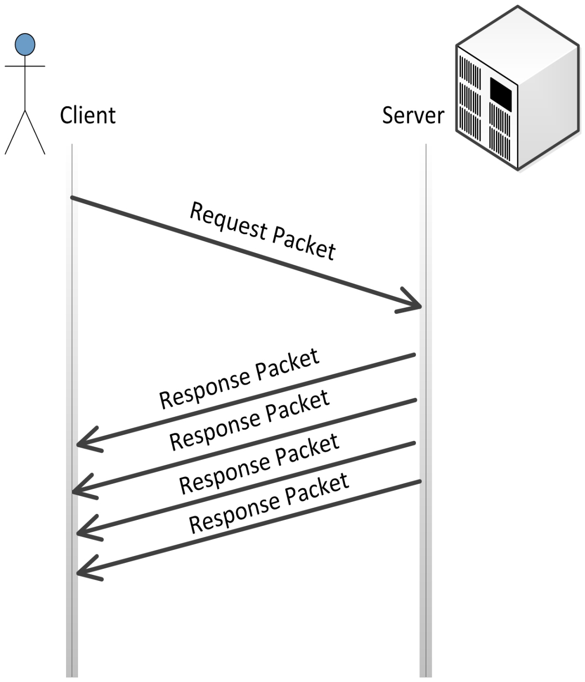


Figure 4.10 Packet flow between client and server

period of time. In these conditions, switching on the wireless radio interface for every beacon interval would be useless and wastes energy without any effective packet reception. Static STELA design suggests the power saving process should go to the next stage, i.e. exponential increase of the sleeping window.

The slow start phase helps increase the response time by predicting packet's arrival based on network activity and at the same time does not waste energy spent on idle listening. Furthermore, energy conservation is obtained through reducing the frequency of waking ups by switching to the exponential increase stage.

4.3.2.2 The Exponential Increase Phase

The exponential increase phase involves two major tasks: sampling the radio channel sampling and updating the sleeping window. The size of sleeping window is doubled every time no data packet is detected during radio channel sampling. As already described, fixed size sleeping window is a simple but unrealistic idea which does not consider the nature of the network traffic. It is more likely in network that packet exchanges happen in bursts, which means when a communication begins, packets are continuously sent out then transmission stops, and the link may stay silent for a while before the next transmission takes place. This can also happen in real life where people interact with computers intensely over short periods with breaks in between instead of regularly sending requests over long periods. The exponential increase of the sleeping window is adopted by STELA to fully exploit the long interval between bursts and increase sleeping interval of the WNIC.

The exponential increase phase starts as soon as the slow start phase ends. At the end of the slow start phase, it can be assumed that most of the response packets have arrived at the client side, and it is very likely that the communications link will stay silent before the next transmission is performed. The sleeping window size is doubled each time the client listens to the beacon and does not receive any buffered data.

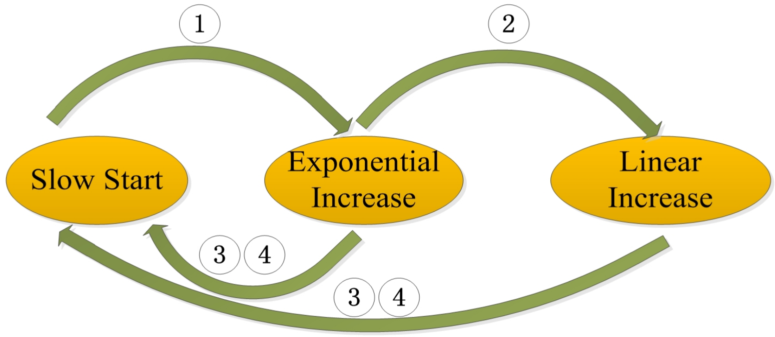
Another situation occurs when some packets do not follow immediately the previous ones, and there is a time gap between these two subsets of packets. The main reason causing this situation is that there might be variable processing time at the server side, or variable delivery delay. The node which has received the previous subset of packets will wait for the sleeping period, find out that no more packets are arriving, and therefore will switch from the slow start phase to the exponential increase phase. This situation will introduce some delay; however the next subset of packets will arrive at the access point very soon, and the client will wake up to receive them in this short period of time, as the sleeping window has not increased too much before noting there are incoming packets from the beacon information. The a slow start phase will be employed again until the last packet of this subset is transmitted. And therefore the delay introduced has acceptable values.

It is possible that after sending out a request, the wireless node does not receive any reply packets in its slow start phase. This phenomenon is mainly caused by long delays between client and server, and relatively short listening periods. In this case, the slow start increase function is turned off in order to ensure better energy performance, and the exponential increase phase is triggered. However, the packets are likely to arrive at the access point before the sleeping window size gets too large, and the slow start receiving process for a sequence of packets is triggered once again. This process gets repeated until all response packets are received by the client.

The exponential increase phase is terminated when the size of the sleeping window reaches a threshold value W_{thre}. The reason behind setting a threshold value is to prevent any aggressive increase of the sleeping period to violate the potential QoS constraints. Inappropriate W_{thre} values determine unbalanced tradeoff between energy efficiency and delivery performance. Large W_{thre} values result in aggressive increase of the sleeping window causing unacceptable QoS degradation. Small W_{thre} values guarantee better performance in terms of network QoS, but at the expense of frequent switching the WNIC between states and thus reducing battery lifetime. The exponential increase phase is also replaced by the slow start phase when any packet is received at the client side. The exponential increase phase is depicted in Algorithm 1 where the sleeping window W_s is doubled.

4.3.2.3 Linear Increase Phase

Once the threshold value W_{thre} is reached, the linear increase phase is triggered to perform a moderate increase of the sleeping interval. During the linear increase phase, the size of the sleeping window is increased with one beacon interval only in each step, as shown in Algorithm 1, where W_s++ indicates this increment. Different from IEEE 802.16 where the sleeping window stays static when the maximum value is reached, this thesis is innovative and considers that it is not fair to specify an exact value at which the listening intervals to stay at without considering the real time traffic. For example, if the client has just turned its wireless device on and left the room to attend other business, no data request will be made within a long period of time, and no packets or other information will arrive at the device. The static threshold approach will let the sleeping window stay at fixed size, which will consume much energy for regular wakeup and going back to the sleep mode. On the other hand, the proposed linear increase phase allows the sleeping window grow at a moderate pace with one beacon interval at each step, which introduces benefits in terms



1	Absence of incoming packets
2	Sleeping window size reaching threshold value
3	Receiving packets
4	Generating new data request

Figure 4.11 Phase transitions of STELA

of energy efficiency without a significant growth in delay and jitter.

Furthermore, increasing the sleeping window after reaching the specified maximum value W_{thre} is a good energy saving choice. The size of the sleeping window is not limited by the parameter set, actually adapts to the real life situations. When there is no event happening, the sleeping window will keep increasing, but not too much at any moment of time, which result in acceptable values. In this case, even if there is data incoming to this node during the linear increase function, there will be some start-up delay, but will not have excessive values. In conclusion, the linear increase of sleeping window size provides a good compromise balancing the energy saving and delivery performance.

This stage terminates and the slow start phase starts again when any packet is received at the client side signaling a potential new data burst. The events that triggers phase transitions are illustrated in Figure 4.11.

4.3.3 Static STELA Performance Analysis

In this section, we evaluate the performance of static STELA compared with IEEE 802.11 and IEEE 802.16 in terms of energy efficiency and QoS (i.e. packet delay).

4.3.3.1 Packet Delay

As part of network QoS, packet delay refers to the time interval elapsed from the moment a packet is sent to the time the packet successfully arrives at the destination. Packet delay D_t in wireless networking is measured as the sum of the following components:

- **Sender-side delay**: D_{tr} refers to the time taken to put all packets' bits to the wireless medium. D_{tr} depends on the packet length and channel bandwidth.
- **Access point (service gateway)-side delay**: D_p+D_{qu}. In infrastructure based networks, packets are first received and processed by the access point (service gateway) before being forwarded to the receiver. Therefore the gateway incurs processing delay D_p. Moreover, if multiple packets are buffered, queuing delay D_{qu} is introduced.
- **Receiver-side delay**: D_{sl} refers to the WNIC sleeping delay.
- **Propagation delay**: D_{pr} refers to the time it takes for a packet to be transferred over a medium. It is the distance between the server and client divided by the propagation speed.

When using the same network architecture and configuration, transmission, processing and propagation delays remain the same. Queuing delay depends on the size of the burst set in the traffic shaper and can be controlled during configuration. Consequently the main variable of the total delay is D_{sl} which depends on the duty cycle of the radio transceiver. The value of D_{sl} for each packet is dependent on the packet position in a packet burst and the regularity of the data flow. In our analysis, we focus on D_{ex}, which is the extra delay introduced by static STELA, as shown in equation (4.1).

$$D_{ex} = D_{sl} - I_{ac} \tag{4.1}$$

I_{ac} is the time taken by a packet to arrive at the mobile host if no power saving is employed and is fixed for all compared schemes.

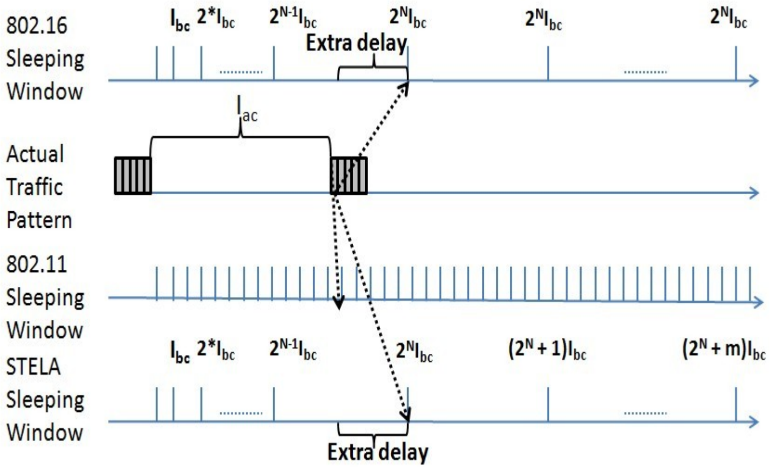


Figure 4.12 Illustration of paceket arrives during exponential increase phase.

Due to the bursty nature of the data traffic, packets can be divided into two types: the first packet in a burst denoted as $TypeI$ and the rest of the packets within the burst referred as $TypeII$. $TypeII$ packets are not affected by any extra delay caused by static STELA as the slow start phase ensures all such packets are received with the shortest delay by increasing the WNIC duty cycle. Two scenarios are considered for the $TypeI$ packets each having a different effect on the performance of static STELA and IEEE 802.16.

The first scenario occurs when a packet arrives at the mobile host during the exponential increase phase of static STELA, as shown in Figure 4.12. Static STELA and IEEE 802.16 introduce the same amount of extra delay D_{ex} as the exponential increase phase issued by both algorithms. As shown in equation (4.2), D_{ob} is the observed delay, and I_{ac} is the time it should take the packet to arrive at the mobile host if no adaptive sleep/wakeup algorithm is employed. I_{ac} is calculated as the packet delay observed when IEEE 802.11 is employed.

$$D_{ex} = D_{ob} - I_{ac} \tag{4.2}$$

We assume that the packets arrive when the sleeping window equals 2^N, and N can be determined using equation (4.3), where I_{bc} is the beacon interval.

$$D_{ob} = (2^1 + 2^2 + \ldots + 2^N) * I_{bc} \tag{4.3}$$

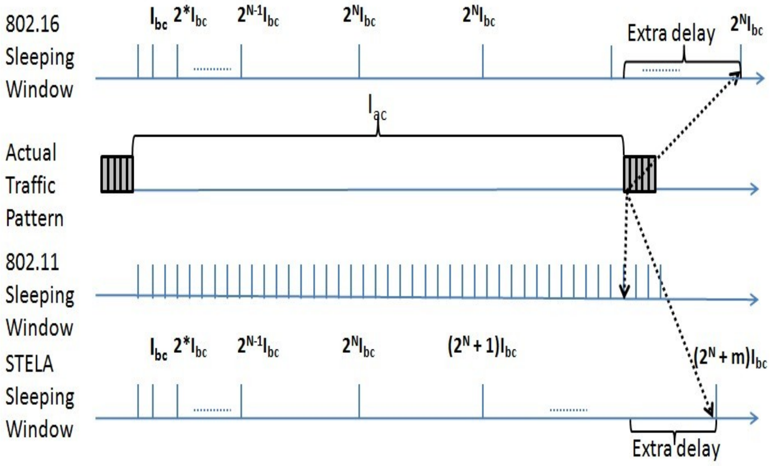


Figure 4.13 Illustration of paceket arrives during linear increase phase.

This can be simplified as presented in equation (4.4).

$$D_{ob} = \sum_{n=1}^{N} 2^n * I_{bc} \tag{4.4}$$

As the packet arrival time must be between D_{ob} and the last wake up time when the sleeping window equals 2^{N-1} when IEEE 802.11 is employed, then we can derive equation (4.5).

$$\sum_{n=1}^{N-1} 2^n * I_{bc} < I_{ac} \leq \sum_{n=1}^{N} 2^n * I_{bc} \tag{4.5}$$

Which further leads to equation (4.6):

$$\begin{aligned} D_{ex} &= D_{ob} - I_{ac} \\ &\leq \sum_{n=1}^{N} 2^n * I_{bc} - \sum_{n=1}^{N-1} 2^n * I_{bc} \\ &\leq 2^N * I_{bc} \end{aligned} \tag{4.6}$$

The second scenario refers to the situation when the packet arrives during the linear increase phase, when static STELA is employed, while the sleeping window keeps growing linearly, as shown in Figure 4.13.

We assume the packet arrives when the sleeping window of static STELA equals $2^T + M$. T refers to the threshold value and the value of M can be determined as soon as the condition in equation (4.7) is met.

$$\begin{aligned} D_{ob-stela} &= \sum_{n=1}^{T} 2^n * I_{bc} \\ &\quad +(2^T + 1 + 2^T + 2 + \ldots + 2^T + M) * I_{bc} \\ &= \sum_{n=1}^{T} 2^n * I_{bc} + \sum_{m=1}^{M} (2^T + m) * I_{bc} \end{aligned} \tag{4.7}$$

In order to analyze the extra delay introduced by static STELA, we assume that the packet is ready for transmission immediately after the previous wakeup of WNIC, i.e. the sleeping window equals $2^T + M - 1$, as shown in equation (4.8).

$$I_{ac} = \sum_{n=1}^{T} 2^n * I_{bc} + \sum_{m=1}^{M-1} (2^T + m) * I_{bc} \tag{4.8}$$

Therefore the extra delay caused by static STELA $D_{ex-stela}$ can be calculated based on equation (4.9).

$$\begin{aligned} D_{ex-stela} &= D_{ob-stela} - I_{ac} \\ &= \sum_{n=1}^{T} 2^n * I_{bc} + \sum_{m=1}^{M} (2^T + m) * I_{bc} \\ &\quad - \sum_{n=1}^{T} 2^n * I_{bc} - \sum_{m=1}^{M-1} (2^T + m) * I_{bc} \\ &= (2^T + M) * I_{bc} \end{aligned} \tag{4.9}$$

On the other hand, if IEEE 802.16 is employed, we assume the first packet of a burst arrives at the mobile host when the sleeping window has remained at the value of T for L rounds of sleep/wakeup schedule. $D_{ob-expo}$ is then calculated as in equation (4.10).

$$D_{ob-expo} = \sum_{n=1}^{T} 2^n * I_{bc} + L * 2^T * I_{bc} \tag{4.10}$$

The extra delay introduced by IEEE 802.16 is shown in equation (4.11).

$$\begin{aligned} D_{ex-expo} &= D_{ob-expo} - I_{ac} \\ &= \sum_{n=1}^{T} 2^n * I_{bc} + L * 2^T * I_{bc} - \sum_{n=1}^{T} 2^n * I_{bc} \\ &\quad - \sum_{m=1}^{M-1} (2^T + m) * I_{bc} \\ &= L * 2^T * I_{bc} - \sum_{m=1}^{M-1} (2^T + m) * I_{bc} \end{aligned} \tag{4.11}$$

When a smaller threshold value is set, $D_{ob-stela}$ is more likely to be greater than $D_{ex-expo}$. However, the difference is expected to be small as the linear increase phase does not introduce too much delay. This can also be seen in the experimental testing results, presented in Chapter 6. Moreover, the difference gets even smaller if the threshold is set to larger value.

4.3.3.2 Energy Consumption

The WNIC's energy consumption E_t is the sum of the energy consumed in data transmitting mode (Tx), data receiving mode (Rx), sleeping mode (Sl) and state transition (Sw) among different modes. Each mode is associated with a different level of energy consumption per time unit U_m, as shown in equation (4.12), where T_m stands for the time spent in each mode.

$$\begin{aligned} E_t &= \sum_{m \in M} E_m \\ &= \sum_{m \in M} T_m * U_m \end{aligned} \tag{4.12}$$

As the time spent in data transmitting and receiving modes remains constant, for all the three solutions discussed, as long as the total amount of data transmitted is the same, the only variables are the time spent in sleeping and transition mode. The longer the sleeping periods and the less state transitions occur, the more energy is saved as U_{sw} is much greater than U_{sl} [231] [232].

In the first scenario when data is being delivered during the exponential phase, the number of state transitions triggered by IEEE 802.11 is represented in equation (4.13), as the WNIC is switched on every time when the AP beacons, shown in Figure 4.12.

$$\begin{aligned} T_{sw-fix} &= 2^1 + 2^2 + \ldots + 2^N \\ &= \sum_{n=1}^{N} 2^n \end{aligned} \tag{4.13}$$

As the number of state transition triggered by static STELA and IEEE 802.16 is N, the total energy saved by these two solutions through reduced state switching is calculated as $\sum_{n=1}^{N} 2^n - N$, which grows exponentially with the increase of N.

In the second scenario, shown in Figure 4.13 when the packet arrives during the linear increase phase, T_{sw-fix} can be represented by:

$$\begin{aligned} T_{sw-fix} &= \sum_{n=1}^{T} 2^n + 2^T + 1 + 2^T + 2 + \ldots + 2^T + M \\ &= \sum_{n=1}^{T} 2^n + \sum_{m=1}^{M} (2^T + m) \end{aligned} \tag{4.14}$$

The number of state switching is $T + M$ for static STELA and $T + L$ for IEEE 802.16. Compared with IEEE 802.11, the total state transition number of which grows exponentially with the increase of N and T, both static STELA and IEEE 802.16 are capable of saving energy as the total times of state transitions grow linearly. Moreover, when compared with IEEE 802.16, due to the novel introduction of static STELA's linear increase phase, the sleeping window keeps growing after the exponential increase phase, and therefore M is smaller than L which leads to greater energy saving achieved by static STELA.

In conclusion, static STELA, as a MAC layer solution, aims at providing energy efficient data delivery over WLAN and at the same time balancing the delivery performance. It mainly consists of three phases: slow start phase, exponential increase phase and linear increase phase. The initialization and termination of each phase is triggered by real time data traffic so that the sleeping window size of the WNIC is adapted accordingly. One key parameter that defines the boundary between the exponential increase phase and the linear increase phase is the threshold value, which can be configured to obtain either maximum energy savings or best QoS.

However, static STELA does not provide a balanced performance between energy efficiency and delivery quality, as the threshold value is configured without any information of real time traffic. Therefore, dynamic STELA is proposed in this work and introduced in the next section, which aims at balancing the two factors through the introduction of a self-configured threshold value.

4.4 Dynamic STELA

The energy efficiency achieved by static STELA is determined by the system configuration of W_{thre}, which is the sleeping window threshold. If large energy saving is expected by the user, for example in the case of an almost empty battery, W_{thre} can be set to a large value so that the state switching of WNIC can be minimized and long sleeping periods will occur. On the other hand, if QoS is considered more important than the battery life, which is often required for multimedia applications, the user can set W_{thre} small so that the WNIC is able to wakeup more frequently to fetch data from the access point.

There is also another scenario in which the user asks for a more balanced performance between energy efficiency and QoS, for instance the user desires to get good quality multimedia playback with the limited battery life left. Therefore we propose **a self-adjusted version of STELA**, i.e. dynamic STELA, to provide a **well balanced performance without the need of direct user configuration**.

Dynamic STELA consists of four phases: slow start, exponential increase, linear increase (i.e. the three of which are the same as in static STELA), and an extra threshold adjusting phase.

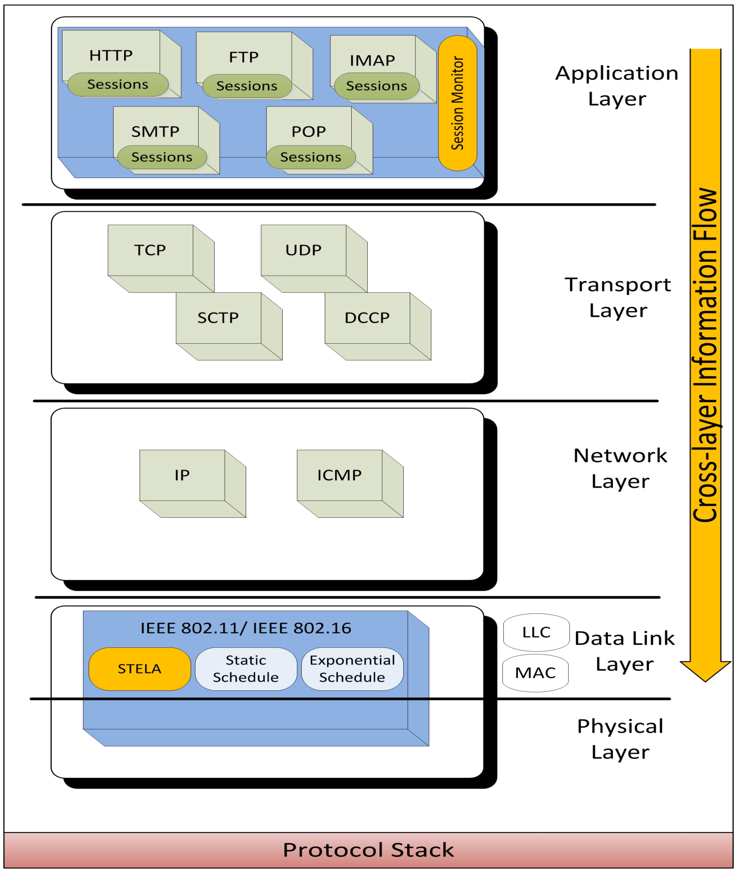


Figure 4.14 Dynamic STELA architecture overview

4.4.1 Dynamic STELA Architecture

Different from MAC layer static STELA, dynamic STELA is a cross-layer solution which requires cooperation between application layer and MAC layer. At application layer, a session monitor is employed to watch for any changes made to the application sessions, while at the MAC layer, the four-phase decision maker is used to determine the sleep/wakeup schedule of the WNIC, as shown in Figure 4.14.

4.4.1.1 Architecture Overview

The server and access point function the same way in dynamic STELA as in static STELA, as shown in Figure 4.5. The client side architecture of dynamic STELA is shown in Figure 4.15.

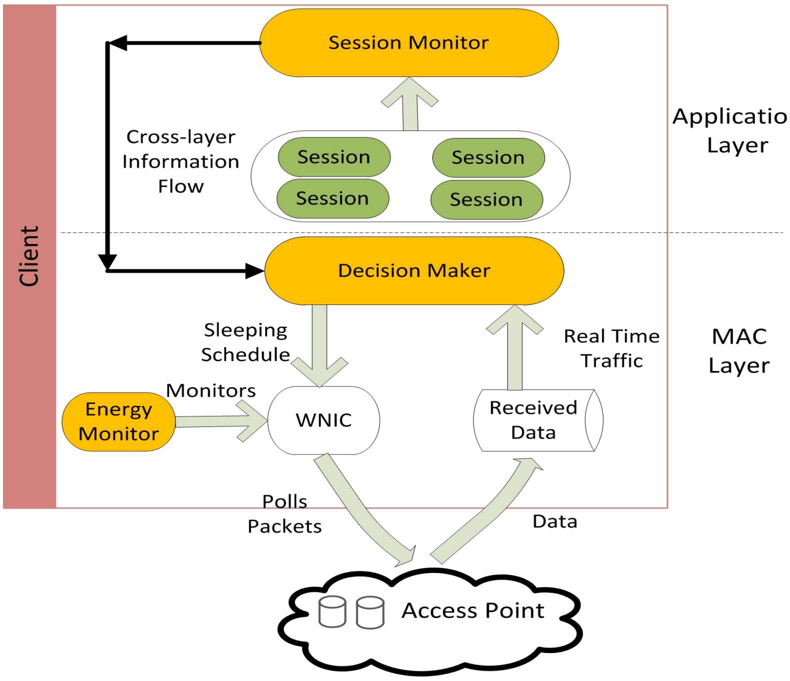


Figure 4.15 Client side architecture of dynamic STELA

The major components of dynamic STELA consist of a *Decision Maker*, an *Energy Monitor*, and a *Session Monitor*. The former two components are employed at the MAC layer of the mobile device while the session monitor is deployed at the application layer. The input of the decision maker module, as shown in Figure 4.16 consists of the current phase of the WNIC, and the real time packet arrival information, which is similar to static STELA. However, additional information related to the application sessions is required as dynamic STELA introduces an extra sleeping window management phase. Being a cross-layer solution, dynamic STELA collects real time session information from application layer through the information flow. The output of the module is the next phase that the WNIC should switch to. The energy monitor module remains the same as static STELA, introduced in Section 4.3.1.

4.4.1.2 Cross-layer Information Passing

This section presents the cross-layer information passing mechanism, which collects the information on the termination or initialization of any session at the application layer,

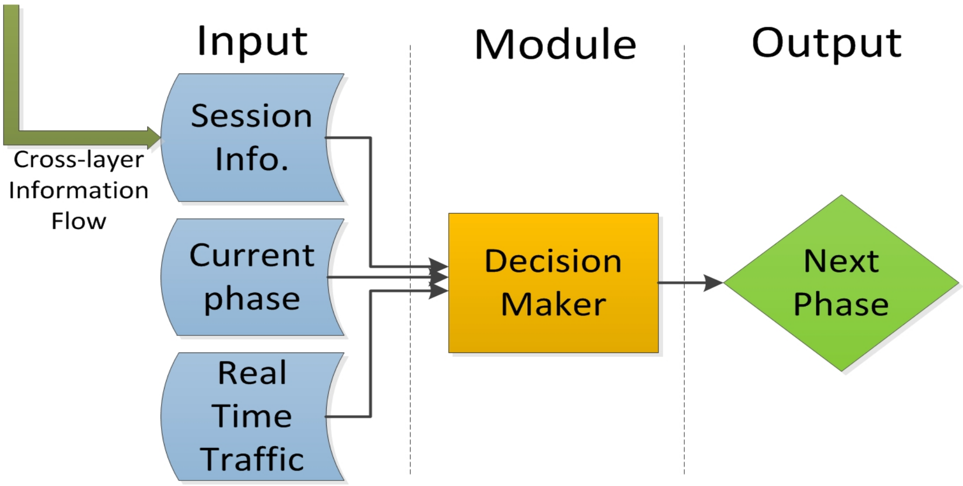


Figure 4.16 Decision maker in dynamic STELA

and sends it to the MAC layer dynamic STELA to assist its adaptation.

At the application layer, a Session Monitor is introduced at the client host, as shown in Figure 4.17. The input of the module is the real time information of application sessions, i.e. the termination or initialization of a session, and the output is the notifier sent to MAC layer through a field in the packet header. The main function of this module is to monitor active sessions and notify the MAC layer whenever a new session is established or an active session is terminated.

Control packets are generated at the application layer for notification purposes. The application layer Session Monitor at the client host informs MAC layer about traffic pattern related activities, and a notifier is sent to the MAC layer if changes happen. There is no influence to the intermediate layers including transport layer and network layer.

4.4.2 Dynamic STELA Algorithm

As already mentioned, dynamic STELA consists of four phases: *slow start*, *exponential increase*, *linear increase* and *parameter tuning*. The first three phases are the same as in static STELA, while the parameter tuning phase is newly introduced to provide balanced performance between energy efficiency and QoS. This phase is the configuration of W_{thre} value. The algorithm of dynamic STELA is described in Algorithm 2. The slow start

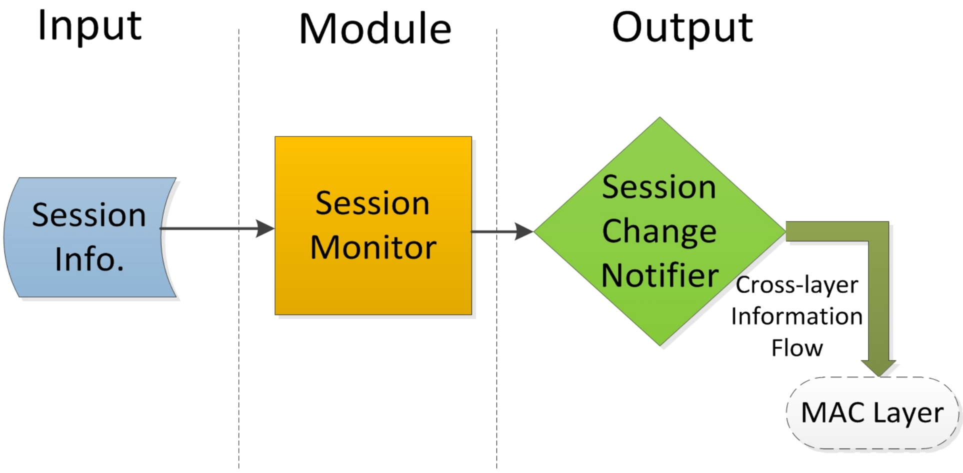


Figure 4.17 Session monitor in dynamic STELA

phase is triggered when any data is received and ends with absence of data. After the termination of the slow start phase, the exponential increase phase is initiated and repeated until the threshold value W_{thre} is reached, after which the linear increase phase is started. The parameter tuning phase is responsible of configuring W_{thre}, and the phase takes place when there is any change in application session activities.

W_{thre} has significant impact on the performance of static STELA in terms of both energy efficiency and QoS as it determines when W_s should stop increasing fast and the more cautious linear increase phase should start. In order to provide a balanced performance between energy efficiency and QoS, W_{thre} is dynamically adjusted in dynamic STELA according to the real time traffic pattern. The *parameter tuning* phase is activated every time an application layer multimedia session is established or terminated, as changes in the active sessions are highly likely to incur changes in the traffic pattern. The initiation of parameter tuning is enabled by cross-layer control packet passing. The field SESSION_INFO is added to the control packet header, the value of which is set to 0 by default. At the client side, the monitor generates a control packet whenever an application session is established or terminated, which indicates expected changes in traffic pattern. The information is passed down from application layer to MAC layer through the Boolean field SESSION_INFO, which is set to 1 in case of any session change. The MAC layer monitor probes the control packet and if the value equals 1 initiates the parameter tuning phase, which monitors the next two rounds of data transmission for future traffic prediction based on the feature of high regularity in data arrival patterns [16] [17].

Parameter tuning works as follows. When the parameter tuning phase is triggered, the first two rounds of data transmission and data arrival patterns are monitored to measure the interval between bursts and predict the arrival time of the next burst, as in general data arrival patterns are highly regular [16] [17]. The parameter tuning is triggered again whenever STELA receives a signal from the Session Monitor. To be specific, W_{thre}, by default, is set to one beacon period, while W_s is operated by STELA based on real time traffic. Whenever W_s is increased during the exponential increase or linear increase phases, it is compared with W_{thre}. W_{thre} is doubled if W_s is greater than twice the W_{thre}. W_{thre} keeps growing until the first packet of the next burst is received. At this point, the value of W_{thre} is smaller than W_s. We do not double W_{thre} at this point to enable early wakeup of WNIC for the next burst in case the burst arrives earlier than expected. The interval between these two bursts is denoted as Burst Gap I_{ob}.

The estimated W_{thre} is set to a smaller value than the sleeping window W_s when the first packet of the second data bursts arrives. The reason behind this is that a relatively conservative W_{thre} guarantees shorter delays and better QoS levels. Ideally, no extra delay is incurred in the best conditions when the traffic pattern follows the same rules as in the current session so far, and the WNIC is switched on immediately after the data burst arrives at the AP and is ready for transmission. However, fluctuations exist in the real life traffic and it is highly likely that the inter-burst interval is either a little bit longer or shorter than the monitored I_{ob}. In this case, a conservative value of W_{thre} guarantees earlier termination of the binary exponential increase of the sleeping window size and therefore the next data burst will be received in time.

4.4.3 Dynamic STELA Performance Analysis

Similar to static STELA, two scenarios are considered, each having a different effect on the end-to-end packet delay. The first scenario, Case 1, involves a packet being the first

Algorithm 2: Dynamic STELA

```
Initialize the sleeping window of the WNIC, $W_s$= 1;
Initialize the threshold value $W_{thre}$= 1;
Initialize the sleeping intervals counter $W_l$= 1;
for every beacon interval do
    if $W_l$ = 1 then
        wake up, listen to beacon;
        if buffered data is detected then
            Retrieve data;
            $W_l$ = 1;
            $W_s$ = 1;
        else
            if $W_s < W_{thre}$ then
                $W_l$ = $W_s$ = 2*$W_s$;
            else
                $W_l$ = $W_s$ = $W_s$++;
                if First Burst Gap and $W_s > 2 * W_{thre}$ then
                    $W_{thre}$ = 2*$W_{thre}$;
    else
        $W_l$–;
```

packet of a burst and the arrival pattern of each burst follows the exact same pattern. Under these circumstances, the radio transceiver wakes up when the sleeping window reaches the threshold value W_{thre}, which is set smaller than the recorded sleeping interval between bursts. Consequently the first packet of a burst could be detected in time after being processed by the gateway. Therefore, D_{sl}, i.e. the delay caused by WNIC being switched to sleeping mode, is smaller than the interval between consecutive data bursts I_{ob}. This scenario is the ideal situation and has low probability. The second scenario, Case 2, involves a packet being the first of a burst and the bursts arrival pattern fluctuates, either arriving early or late. In this case, the extra delay introduced by dynamic STELA, D_{ex} depends on the actual arrival time of the burst I_{ac}. This situation is further divided into three cases. To simplify the analysis, we assume that during the threshold adjusting process, the second burst arrives at the client host when the sleeping window has just exceeded W_{thre} (i.e. W_s equals $W_{thre} + 1$).

In the first case, Case 2.1, the burst arrives earlier than the interval observed in the

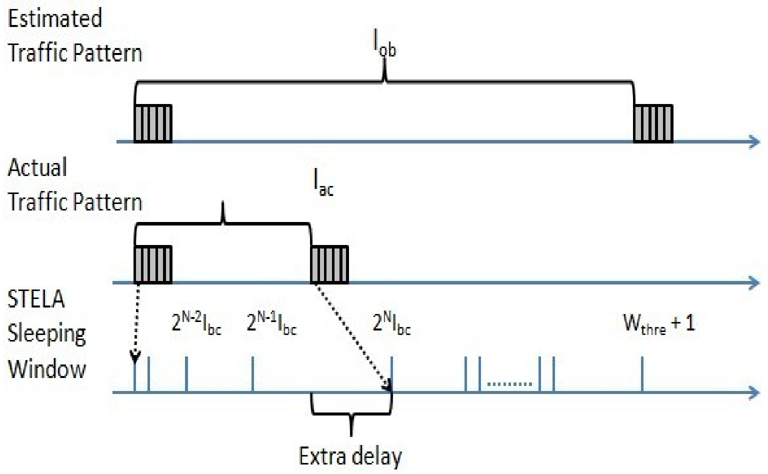


Figure 4.18 Illustration of case 2.1, data burst arrives early

first round for at least the interval specified by W_{thre}, as presented in Figure 4.18. In this situation, the packets must arrive at the receiver when the exponential increase phase is active as the sleeping window is smaller than W_{thre}. We assume that the packets arrive when the sleeping window equals 2^N, and N can be determined using equation (4.15), where I_{bc} is the beacon interval.

$$D_{sl} = (2^1 + 2^2 + \ldots + 2^N) * I_{bc} \tag{4.15}$$

This can be simplified as presented in equation (4.16).

$$D_{sl} = \sum_{n=1}^{N} 2^n * I_{bc} \tag{4.16}$$

As the radio transceiver is switched on when the sleeping window reaches W_{thre}, the delay is computed between the times of the last burst and when W_s equals 2^{N-1}, giving:

$$I_{ac} = \sum_{n=1}^{N-1} 2^n * I_{bc} \tag{4.17}$$

Therefore the extra delay introduced by dynamic STELA can be calculated based on equa-

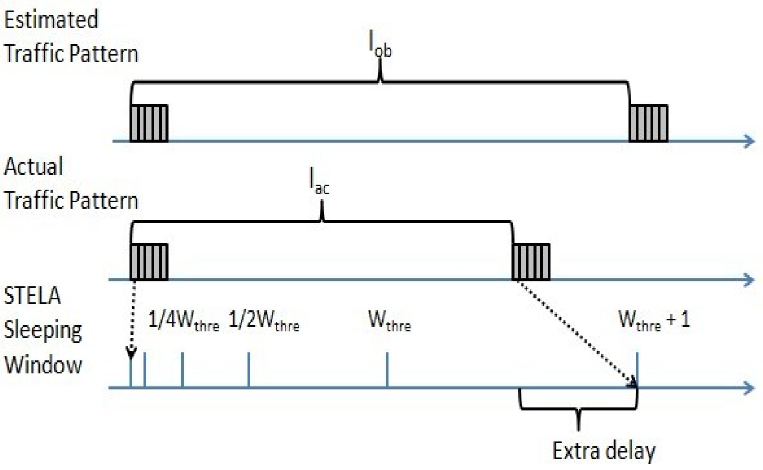


Figure 4.19 Illustration of case 2.2, data burst arrives early

tion (4.18).

$$\begin{aligned} D_{ex} &= D_{sl} - I_{ac} \\ &= \sum_{n=1}^{N} 2^n * I_{bc} - \sum_{n=1}^{N-1} 2^n * I_{bc} \\ &= 2^N * I_{bc} \end{aligned} \tag{4.18}$$

However, as the binary exponential increase phase is not finished before the packet arrives, the sleeping window is smaller than W_{thre}, which means the condition in (4.19) is valid.

$$D_{ex} < W_{thre} * I_{bc} \tag{4.19}$$

Case 2.2, as shown in Figure 4.19 represents the scenario where the burst arrives later than in the first case, but earlier than the observed interval I_{ob}. When W_s reaches W_{thre}, the WNIC samples the radio channel and switches back to the sleeping mode, as no packet arrives. Then, the linear increase phase is initiated and W_s is incremented by one and becomes $W_{thre} + 1$. The burst is ready for reception when the radio is switched on the next time, and therefore only W_{thre} beacon periods are added to the overall packet delay. In this situation, we have the delays from equations (4.20) and (4.21).

$$D_{sl} = I_{ob} \tag{4.20}$$

$$D_{ex} = D_{sl} - I_{ac} = I_{ob} - I_{ac} \tag{4.21}$$

As we have the inequality from equation (4.22):

$$I_{ob} - W_{thre} * I_{bc} < I_{ac} < I_{ob} \tag{4.22}$$

then:

$$I_{ob} - I_{ob} < D_{ex} < I_{ob} - I_{ob} + W_{thre} * I_{bc} \tag{4.23}$$

Equation (4.23) leads to equation (4.24), indicating how D_{ex} is bound by:

$$0 < D_{ex} < W_{thre} * I_{bc} \tag{4.24}$$

Figure 4.20 illustrates the last scenario, Case 2.3, in which the burst gets to the gateway later than the interval observed during the first round. The radio transceiver wakes up when W_s reaches W_{thre} and receives no packet, then sleeps for $W_{thre} + 1$ beacon intervals and samples the radio channel until the delayed burst is ready for transmission. We assume the size of the sleeping window is $W_{thre} + N$ when the burst is detected and the value of N can be determined as soon as the condition in equation (4.25) is met.

$$D_{sl} - I_{ob} = (W_{thre} + 1 + W_{thre} + 2 + \ldots + W_{thre} + N) * I_{bc} \tag{4.25}$$

This can be simplified as presented in equation (4.26).

$$D_{sl} - I_{ob} = \sum_{n=1}^{N} (W_{thre} + n) * I_{bc} \tag{4.26}$$

Next the delay is calculated based on equation (4.27).

$$D_{sl} = I_{ob} + \sum_{n=1}^{N} (W_{thre} + n) * I_{bc} \tag{4.27}$$

The burst arrives at the client side when sleeping window equals $W_{thre} + N$. However, in order to analyze the worst case scenario in terms of delay, we assume the burst is ready for transmission immediately after the previous wakeup, i.e. when the sleeping window is equal to $W_{thre} + N - 1$. This leads to equation (4.28).

$$I_{ac} = I_{ob} + \sum_{n=1}^{N-1} (W_{thre} + n) * I_{bc} \tag{4.28}$$

The extra delay D_{ex} introduced by dynamic STELA can be calculated based on equation (4.29).

$$\begin{aligned} D_{ex} &= D_{sl} - I_{ac} \\ &= I_{ob} + \sum_{n=1}^{N} (W_{thre} + n) * I_{bc} \\ &\quad - I_{ob} - \sum_{n=1}^{N-1} (W_{thre} + n) * I_{bc} \\ &= (W_{thre} + N) * I_{bc} \end{aligned} \tag{4.29}$$

Although N depends on the inter-arrival pattern of bursts, the extra delay is a linear function of the beacon interval as appose to the binary approach used in the IEEE 802.16 standard.

Above all, it has been proved that for packets that are not burst starting packets, which is true in most cases, the end-to-end delay is very short with an upper bound. Regarding the packets at the beginning of each burst, the extra delay introduced by dynamic STELA should be smaller than the observed interval between bursts in the first round of traffic or should be a linear function of the beacon interval, which lasts far less than one second in most cases. Experimental results also prove that dynamic STELA achieves high energy

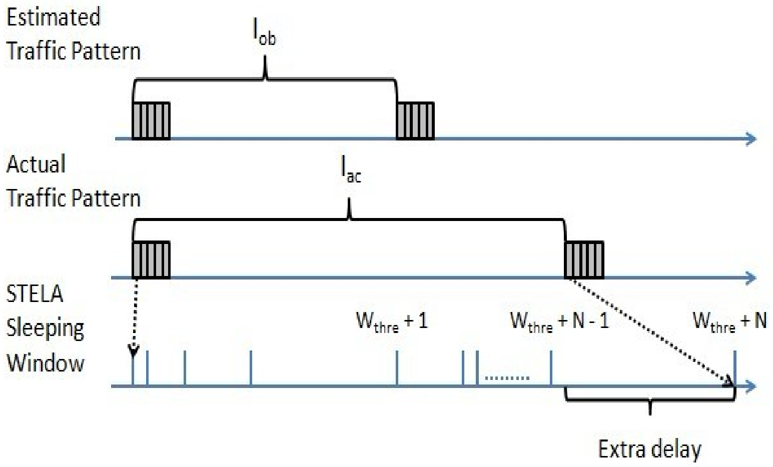


Figure 4.20 Illustration of case 2.3, data burst arrives late

efficiency while maintaining short delays. Although dynamic exploits the bursty nature of data traffic in order to save energy, it does not fully exploit the energy-related benefits that can be achieved through deliberate packet shaping at higher layers. In this context, Q-PASTE is proposed and discussed in the following section.

4.5 Q-PASTE

Both static STELA and dynamic STELA utilize the bursty nature of data traffic to extend the sleeping interval of the WNIC. A natural direction to further improve energy efficiency is to shape the data packets into bursts before they arrive at the MAC layer so that the duration between bursts is increased.

The immediate benefit of shaping packets into bursts before their delivery is obvious: the frequency of WNIC waking up is decreased as the wireless interface can now be switched only once during the reception of the whole burst of packets instead of being turned on each time for every single packet. Therefore the WNIC can spend longer time in sleeping mode instead of state switching. In this way power conservation is achieved.

However, there is a big challenge associated with deliberate traffic shaping: how to make sure it does compromise QoS levels significantly? When packets are grouped into bursts, they need to stay in a buffer until a certain requirement is met before being released to the client. The critical issue then becomes the choice of data releasing policy. Should the data be released when the number of packets exceeds a threshold or when packet delay is about to be unacceptable for the users? In this thesis, the data releasing policy is quality-oriented which makes sure the results of the traffic shaping do not affect QoS levels significantly, especially for delay-sensitive multimedia streaming applications.

Therefore, Q-PASTE is proposed in this work to increase energy efficiency and at the same time maintain high QoS levels. On one hand, Q-PASTE shapes relatively sparse multimedia data packets into bursts at the application layer and employs an adaptive sleep/wakeup WNIC scheduling scheme at the MAC layer to utilize the intervals between bursts in order to save energy. On the other hand, the burst releasing scheme takes into consideration the multimedia playout buffer at the client side so that smooth playback is provided.

4.5.1 Q-PASTE Architecture

Static STELA requires MAC layer modification while dynamic STELA requires both MAC and application layer modifications, both at the mobile client side. Although cross-layer solutions normally indicate that extra overhead is introduced as cross-layer information exchange is needed, their complexity is significantly decreased when applied in the layered model. Upper layer information, i.e. application layer in Q-PASTE, contained in a control packet is passed down to the lower layer, i.e. MAC layer in Q-PASTE, via the transport layer and the network layer. Moreover, only two control packets are needed in Q-PASTE to pass information about session information and the multimedia streaming status. Therefore the resulting overhead is not significant.

Q-PASTE involves modifications at both the client device and service gateway. Q-PASTE consists of two major components: an application layer module **Packet/ApplicaTion manager (PAT)** and MAC layer module **STELA**.

PAT is employed at the application layer of both gateway and client device. The **client side PAT** keeps track of all active application sessions and informs STELA ,the MAC layer module, through a cross-layer information flow whenever an application session ends or a new session starts. On the other hand, the **gateway side PAT** collects real time information about the number of packets ready to be transmitted to the clients and does not allow data releasing until the packets can be shaped into new bursts.

Although data scheduling can also be fulfilled at the server/client side, the packet shaping function of PAT is implemented as part of gateway functions in our energy efficient solution due to several reasons. First, client side shaping requires extra cost in terms of both hardware and software complexity, as burst should be formed before being received by the WNIC and thus an extra receiver unit should be installed. Second, if continuous data is manipulated at the server side to form bursts, the client node might be communicating with several servers at the same time, and therefore needs to wake up for each burst from each server. Moreover, as a burst is released as soon as the buffer size reaches the threshold, extra delay is introduced when packets from individual servers are considered separately as they will wait longer before being delivered. On the contrary, a service gateway could be either a media gateway which already exists on the market or a simple router that has built-in application layer control. The gateway can act as a centralized controller and gather data from all the servers for the client. This approach is the most efficient in terms of both energy saving and QoS level. The architecture of Q-PASTE shown in Figure 4.21 is composed of gateway side and client side components. The two components are described in details next.

4.5.1.1 Gateway Side Component Architecture

The Q-PASTE service gateway is composed of classic components at all layers except application and MAC layers where PAT and STELA modules were added. A cross-layer information flow is employed at both client side and service gateway to pass information from application layer to MAC layer. At the client side, real time session information is carried in the flow, while at the service gateway the information about fast start phase involved in traffic shaping is carried in the flow. Additional modules are included at the

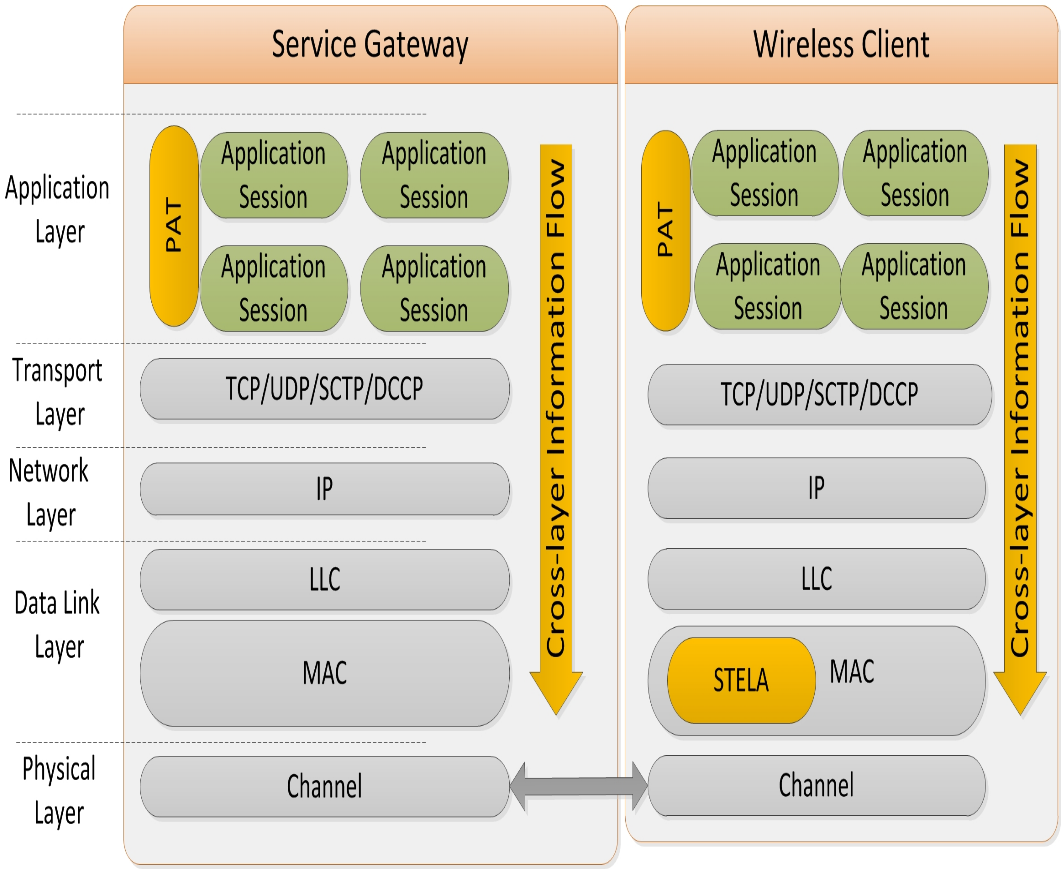


Figure 4.21 Q-PASTE architecture overview

application layer: *Traffic Shaper*, *Buffering Timer* and the *MAC Notifier*, as shown in Figure 4.22.

The *Traffic Shaper* is used to deliberately gather a group of packets into a burst before releasing them to the client. The input of this module, as shown in Figure 4.23 includes the packets received from the network, and the buffering timer which is restarted for every period of t_{bf} after the last data release. The output is the decision of whether to release the buffered data for transmission or not.

The *MAC Notifier* module, as illustrated in Figure 4.24, monitors the traffic shaping process and gathers information about the time elapsed from the beginning of data transfer. The output of the module is a control message sent cross-layer to the MAC layer. This message has set a new header field, FS_END. The default value of FS_END is 0. And the field is set to 1 for the first packet generated when the observed elapsed time exceeds

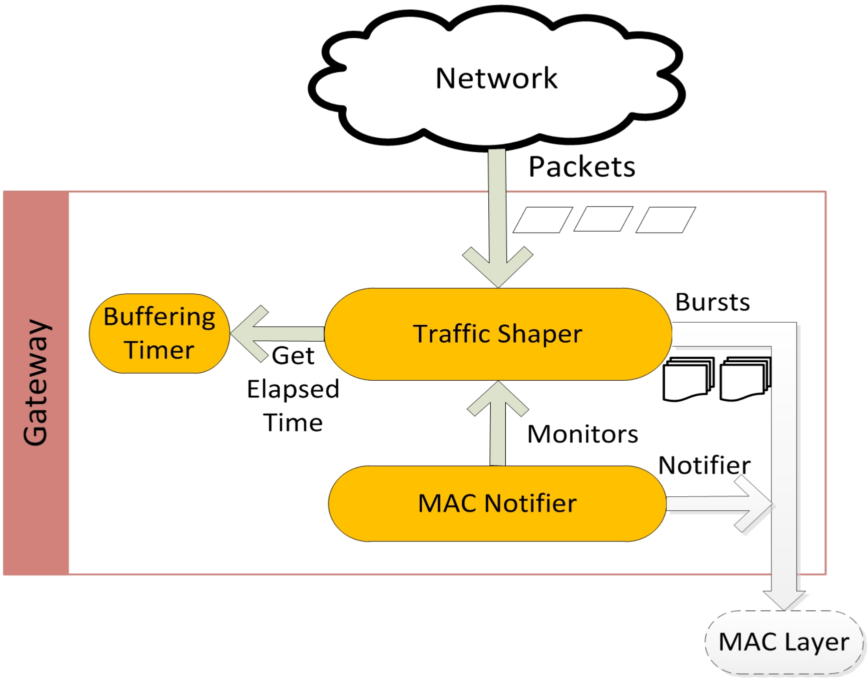


Figure 4.22 Gateway side architecture of Q-PASTE at application layer

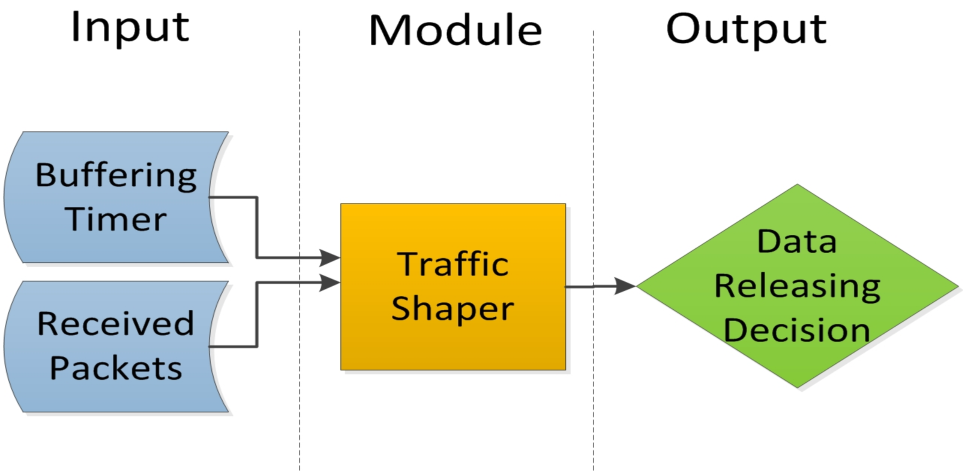


Figure 4.23 Traffic shaper in Q-PASTE

the fast start period T_{fs}. The value of FS_END is set back to 0 for the rest of the control packets. The information is passed down from the application layer to the MAC layer, so that the MAC layer module can take corresponding actions. The process of setting the FS-END value is repeated for every multimedia clip.

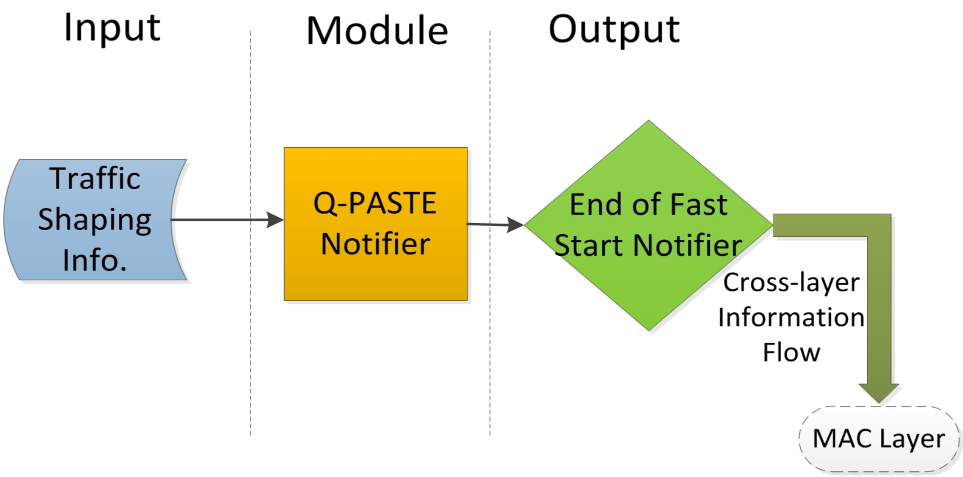


Figure 4.24 MAC notifier in Q-PASTE

4.5.1.2 Client Side Component Architecture

The Q-PASTE client side architecture, illustrated in Figure 4.21, includes classic comonents at all layers except application and MAC layers where two modules are added respectively: PAT and STELA. Additionally, a cross-layer information module is also employed. PAT and STELA were described in details before, where as the cross-layer information module provides information from application layer to MAC layer.

A particular component at the client side is the playout buffer, as shown in Figure 4.25. The buffer is used to alleviate the degradation caused by unwanted changes in the data rate. Packets are temporarily stored at the client buffer in order to smooth out any potential bandwidth variation, and are pulled out and then decoded and displayed by the multimedia player. The *fill rate* is the rate the data enters the client buffer and the *drain rate* is the rate the playout removes data from the client buffer. The fill rate varies during the playback process not only because the network condition fluctuates, but also due to the traffic shaping scheme employed by the service gateway. The fill rate reaches its peak during the fast streaming phase when packets are directly relayed to the client without buffering. The drain rate is related to the encoding rate of the corresponding multimedia content. By bufferring data, the receiver is able to smooth out any temporary variations in the received data rate.

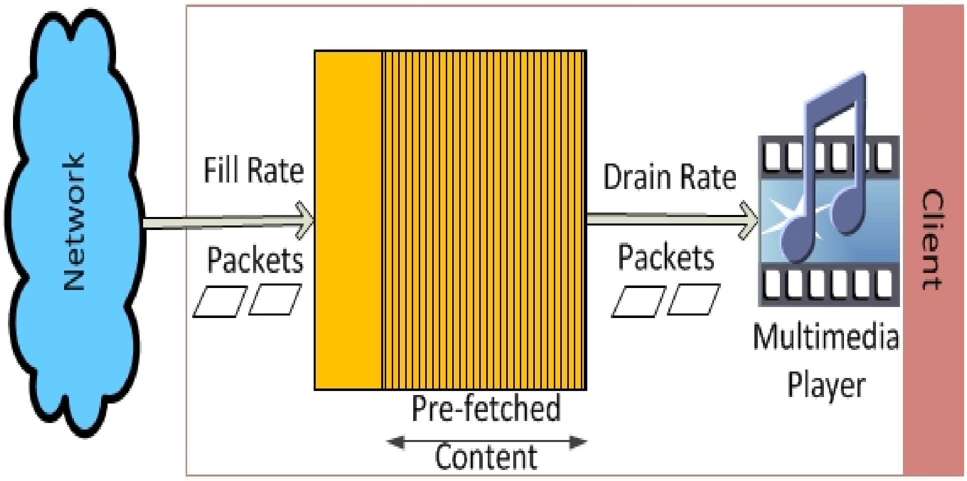


Figure 4.25 Playout buffer in Q-PASTE

4.5.2 Q-PASTE Algorithm

The Q-PASTE algorithm mainly includes the following components, which will be described in details next: traffic shaping and adaptive scheduling.

4.5.2.1 QoS-aware Traffic Shaping

Different from existing traffic shaping solutions, we propose a gateway-based Traffic Shaper that focuses on supporting smooth playback of audio/video streams, while also increasing the energy efficiency of the transmission process. The Traffic Shaper is located at the level of PAT, at the application layer of the gateway, as illustrated in Figure 4.21. Real time multimedia applications use typically a smoothing buffer (playout buffer) at the client to compensate for all variable delays. Multimedia delivery employs normally a fast start approach for a period, T_{fs}, at the beginning of each data transfer. Data is transmitted at a higher rate than the normal streaming rate to fill the client side playout buffer in order to provide smooth playback [87]. During this period, the traffic shaping function does not buffer any packet from the server and the playout buffer is filled normally. After the fast start period, data transmission rate drops down to the natural rate of the play-back. During this period, PAT hides data from the client at the gateway, groups packets into bursts of large size, and then for every buffering period t_{bf} forwards them to the client at a higher

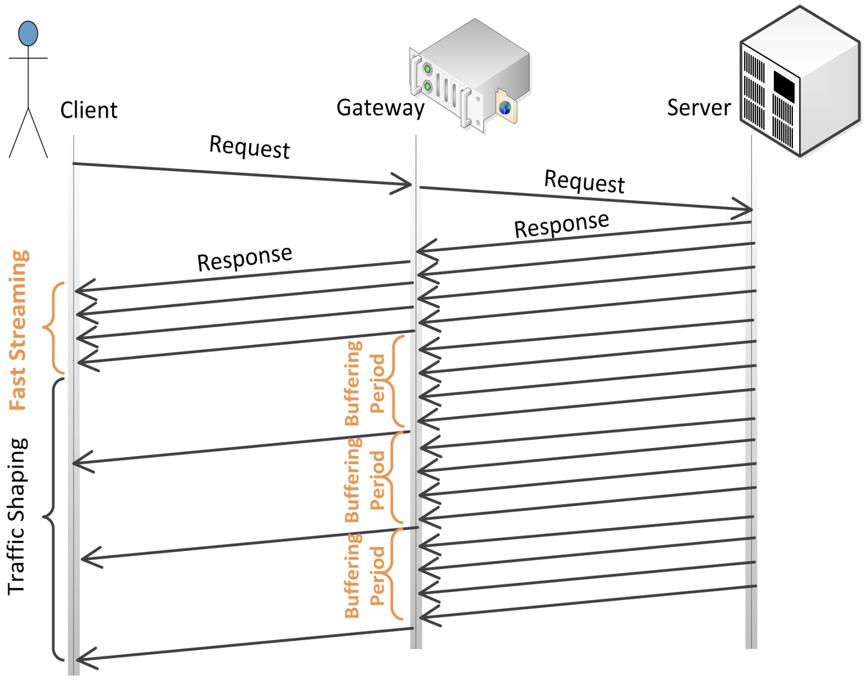


Figure 4.26 Illustration of fast start and data buffering

data rate, as shown in Figure 4.26.

At the end of the fast start period, the size of the playout buffer s_{po} is determined by eq. (4.30), where r_{fs} is the average data rate during fast start.

$$s_{po} = T_{fs} * r_{fs} \tag{4.30}$$

The maximum length of smooth playback provided by the playout buffer is calculated as in eq. (4.31), if no consecutive data packets are released by the gateway to fill the buffer after the fast start period. T_{bf} refers to the maximum buffering period before the playout buffer is emptied and r_{ec} denotes encoding rate at the client side.

$$T_{bf} = \frac{s_{po}}{r_{ec}} \tag{4.31}$$

In order to guarantee that the playout buffer is not emptying, the buffering period t_{bf} should be smaller than T_{bf}. While the server is filling the buffer during fast start or transmitting the rest of the data, the client can start the playback at any time. The buffer can be filled at the fastest pace, and T_{bf} could get its maximum value, if video playing starts after the fast start period. However, the gateway is not aware of when the client player starts draining the buffer, and t_{bf} should be short enough to guarantee the playout buffer does not become empty before the next burst is released to the client. To calculate the minimum value of T_{bf}, the worst situation of user starting playing the multimedia stream at the very beginning of the data transfer is considered. In this situation, the buffer is filled by the server and drained by the user at the same time, and therefore the buffering period should be kept small.

At any point of playout t_{po}, the size of the playout buffer is determined by eq. (4.32).

$$s_{po} = t_{po} * (r_{fs} - r_{ec}) \tag{4.32}$$

Considering eq. (4.32), the minimum value of T_{bf} is calculated as in eq. (4.33).

$$\begin{aligned} T_{bf_{min}} &= \frac{s_{po}}{r_{ec}} \\ &= \frac{t_{po} * (r_{fs} - r_{ec})}{r_{ec}} \end{aligned} \tag{4.33}$$

A Buffering Timer is used at the gateway to manage buffer releasing schedule. This timer is restarted after releasing reshaped bursts for every interval t_{bf}. In the case of static sleep/wakeup schedule, e.g. the WNIC wakes up to sample every beacon interval, the data scheduling time could be the same as t_{bf} as the client host is ready for data reception whenever the gateway beacons followed by data release. However, the sleeping window is dynamic, as defined by STELA and it is highly likely that the buffering timer expires during the sleeping interval of WNIC. This means the client host is not ready for data reception (i.e. it is in sleeping mode), resulting in the fact that the gateway-buffered data cannot be released until the next WNIC wake up. In this case, the scheduling time exceeds t_{bf} and maintaining high QoS level is not achieved due to playout buffer draining. Therefore, in order to make sure that enough data is accumulated at the playout buffer, the gateway is notified of the mobile host's sleep/wakeup schedule to assist energy-efficient data scheduling. At each wake up point, the next wake up schedule is calculated, and the data burst is released immediately if the buffering timer expires before the next wake up.

As PAT always releases data bursts before the playout buffer gets empty, it has to make sure that the buffer does not overflow due to early data scheduling. Before data is released, the size of the drained data is calculated and data is only streamed when buffer overflow will not take place, as expressed in eq. (4.34).

$$s_{ts} + s_{bf} < t_{ts} * r_{ec} \tag{4.34}$$

s_{ts} is the size of scheduled data after fast start period, s_{bf} is the size of data that is going to be released, and t_{ts} is the time elapsed after fast start. The burst scheduling algorithm is presented in Algorithm 3: data bursts are released when the Buffering Time is expiring or is going to expire before the next wakeup point, giving that the playout buffer will not overflow after receiving the bursts.

4.5.2.2 Adaptive Sleep/Wakeup Scheduling

Q-PASTE adopts dynamic STELA at the MAC layer of the mobile host device. Dynamic STELA instead of static STELA is adopted by Q-PASTE at the client side as both dynamic STELA and the traffic shaper as part of Q-PASTE are cross-layer solutions and require modification at the application layer and the MAC layer. Therefore the resulting overhead is smaller when adopting dynamic STELA as appose to employing static STELA which is a MAC layer solution.

The three-phase WNIC scheduling scheme works the same as described in Section 4.4.2, while the parameter tuning phase is only initiated after the fast start. This is indicated by FS_END set to 1 in the cross-layer information flow. However, it is worth mentioning that with the introduction of the traffic shaping at the gateway, dynamic STELA is able to achieve higher energy saving without compromising the user experience levels.

One of the assumptions made when introducing static and dynamic STELA is that

Algorithm 3: Burst Scheduling

```
Input: Buffering timer $t_{bf}$
Output: Burst scheduling time $t_{ds}$
Initialize the sleep/wakeup cycle counter N=1;
Initialize the total scheduled data size after fast start $s_{ts}$=0;
Initialize the elapsed time after fast start $t_{ts}$=0;
Initialize the size of buffer data at the gateway $s_{bf}$=0;
for every WNIC waking up point do
    get $t_{ts}$;
    get $s_{bf}$;
    if $\sum_{n=1}^{N} t_{sl\_n} = t_{bf}$ then
        if $s_{ts} + s_{bf} < t_{ts} * r_{ec}$ then
            release data;
            $t_{ds}$ = $\sum_{n=1}^{N} t_{sl\_n}$;
            reset N = 1;
            $s_{ts}$+=$s_{bf}$;
            return $t_{ds}$;
        else
            continue buffering;
            N++;
    else
        if $\sum_{n=1}^{N} t_{sl\_n} < t_{bf}$ and $\sum_{n=1}^{N+1} t_{sl\_n} > t_{bf}$ then
            if $s_{ts} + s_{bf} < t_{ts} * r_{ec}$ then
                release data;
                $t_{ds}$ = $\sum_{n=1}^{N} t_{sl\_n}$;
                reset N = 1;
                $s_{ts}$+=$s_{bf}$;
                return $t_{ds}$;
            else
                continue buffering;
                N++;
        else
            continue buffering;
            N++;
```

network traffic has a bursty nature and is of high regularity. In this sense, we can put WNIC into sleep mode during the interval between data bursts to reduce energy consumption. In Q-PASTE, the likelihood of traffic arriving in the form of bursts is further increased as data is deliberately grouped into large chunks at higher network layer and therefore the interval between bursts is increased. This further improves the energy efficiency.

During the fast start period, packets from the server arrive in a continuous way which means the WNIC should wake up frequently for data reception. After fast start, as the Traffic Shaper groups data into bursts, the traffic released to the client normally consists of consecutive packets transmitted within short periods of time. In this case, receiving a packet normally indicates an incoming burst and therefore STELA wakes up the WNIC frequently to ensure no extra delay is introduced due to WNIC inactivity and high QoS levels are maintained. During the second phase of STELA when the sleeping window grows fast, the binary exponential increase of W_s does not compromise user experience either, due to the fact that traffic is shaped by the gateway into bursts before being relayed to the clients (i.e. any small bursts from the server are grouped into larger bursts, slightly delaying data). This phase ends once the size of sleeping window W_s reaches the threshold value W_{thre}, value of which is adjusted dynamically to guarantee that WNIC is switched on in time for data reception.

4.5.3 Q-PASTE Performance Analysis

Q-PASTE adopts dynamic STELA at the client side and traffic shaping at the service gateway, which work together to prolong the sleeping interval of the WNIC. It is already seen in Section 4.4.3 that dynamic STELA is able to achieve energy efficiency under all circumstances without compromising delivery quality significantly. This result also applies to Q-PASTE. Moreover, the traffic shaping scheme employed by the service gateway in Q-PASTE further helps form the data traffic into bursts, the feature of which is the key to save energy by switching off the wireless interface. Therefore Q-PASTE provides higher energy efficiency than using STELA alone.

The data shaping scheme employed by Q-PASTE utilizes the fast streaming mechanism which fills the playout buffer at the client side as quickly as possible at the beginning of multimedia streaming, and always releases data to the multimedia player before the buffer starves. The two actions guarantee that the size of data stored in the playout buffer can always satisfy smooth playback, hence high QoS levels are maintained.

4.6 Summary

This chapter presents the detailed description of the three proposed solutions: static STELA, dynamic STELA and Q-PASTE. For each of the solutions, the overall architecture in terms of the protocol stack components is presented, followed by description of each new module. The algorithm is described in details and its pseudo-code provided. Finally, the performance of the contribution is analyzed considering different traffic arriving scenarios in terms of energy consumption and QoS.

The first solution presented is static STELA which provides energy efficient sleep/wakeup scheduling of the WNIC at MAC layer. The solution consists of three phases: *slow start* phase which provides quick response to packets arriving with short intervals, the *binary exponential* increase phase which enables quick growth of the sleeping window size and therefore reduces energy consumption wasted otherwise on idle listening and for frequent state switches, and the *linear increase* phase during which the sleeping window size grows at moderate pace to make sure that energy saving is achieved without degrading QoS levels significantly. The threshold value which terminates the second phase and activates the third phase can be configured to either achieve maximum energy saving or provide good QoS levels.

The second solution presented is dynamic STELA which employs the three phases as in static STELA, and adds a parameter tuning phase which aims at balancing the energy saving and QoS. The parameter tuning phase monitors and studies the historical traffic arrival pattern in order to predict the arriving time of future traffic so that the WNIC is waken up in time when the next burst of packets arrive. The information collected is used to set the threshold value such as it self-adapts to the real time traffic pattern.

Q-PASTE, the third solution presented, is a quality-oriented energy efficient mechanism for multimedia streaming applications. It is deployed at the last hop to the mobile host, i.e. service gateway, and shapes data into bursts at application layer to group packets into bursts and increase the intervals between bursts so that the WNIC can spend more time in sleeping mode. In order to not compromise QoS levels, Q-PASTE utilizes the fast start duration to fill the playout buffer at the client side, so that smooth playback is provided.

Packet delay as an important metric in measuring QoS is analyzed in different scenarios, with packets arriving in different phases. The analysis results have demonstrated that the solutions provided in this work are able to provide high QoS levels while large energy saving is achieved.

Chapter 5

Testing: Environment and Scenarios

This chapter presents the simulation-based testing environment and describes the scenarios used in the simulations. The simulation tool is briefly described, followed by the simulation settings. The metrics used for assessment during the testing are then presented along with the generic simulation network topology employed. For each contribution, the specific network topology tested, the schemes used for performance comparison and the tested scenarios are introduced respectively.

5.1 Testing Environment

5.1.1 Simulation Tool

Simulation-based testing is performed for performance assessment using Network Simulator version 3 (NS3) [1] NS3 is an open-source software which is widely used in network simulation. It consists of a discrete event simulator popularly used in the simulation of MAC, network, transport and application layer protocols. Support for most components of real life networks are provided in this simulator, including duplex links, queues, nodes, etc. These components can be easily deployed and configured for simulation use, intercon-

[1] Network simulator version 3-http://www.nsnam.org/.

necting to each other. Besides that, many popular protocols have been incorporated into this project, and more features and protocols are constantly being added into it. NS3 is a discrete-event simulator, the core of which is implemented in C++. It is basically a C++ library which can be linked and executed by a C++ main program with a network topology defined.

Compared with other simulation tools such as OMNeT++ [2] and OPNET [3], NS3 provides a better organised code hierarchy with plenty of new features contributed to every release. For example, neither of them support power saving schemes and it is more difficult to add modules to both as they are not open source. Moreover, NS3 adopts modern design, pure C++ implementation and many handy features, for example, smart pointer and callback mechanism. Therefore, it is relatively easy to develop bug free implementation of networking protocols. The code structure is easier to understand, which makes maintenance and further development much easier. Therefore NS3 is used as the testing platform for the proposed solutions.

A typical simulation life cycle in NS3 involves several steps. The server/client connections are established through the user assigned IP addresses at the beginning of simulation. Packets are transmitted from the specified start time for the specified duration, according to the configured parameters such as transmitting schedule (i.e. on/off pattern), data rate etc. Trace files are updated during the simulation to record the packet-related events, which is made possible by the built-in callback mechanism.

5.1.2 Simulation Settings

5.1.2.1 Protocol Support

The default MAC and physical layer configurations for IEEE 802.11 and IEEE 802.16 were not changed in NS3. The tested IEEE 802.11 version is IEEE 802.11b, which provides basic WiFi functionality with PSM support, as is needed in the performance comparison.

[2] Objective Modular Network Testbed in C++ (OMNeT++)-http://www.omnetpp.org/

[3] Network Planning and Simulation (OPNET)-http://www.opnet.com/

However, the power saving feature of IEEE 802.16 is not supported in NS3. Therefore, in order to test the performance of the energy efficient mechanism supported by IEEE 802.16, the PSM algorithm in WLAN was modified and adapted to IEEE 802.16.

Static STELA and Dynamic STELA are compared against IEEE 802.11 and IEEE 802.16 standards. These standards are widely used for wireless communications and have built-in power saving schemes. Therefore they are used as the benchmark for testing of most energy efficient MAC layer solutions. Other MAC layer energy efficient solutions are not implemented for testing purpose as low layer protocol modification is very time consuming. However, in order to fully test Q-PASTE, the Buffering Streaming solution was implemented as it mainly requires application modification. Moreover, STELA is part of Q-PASTE which is tested against IEEE 802.11 and the Buffering Streaming solution.

5.1.2.2 Power Saving Scheme

As already mentioned, NS3 has IEEE 802.11 model support for both ad-hoc and infrastructure based networking. However, it fails to implement properly the power saving mechanism (PSM) which is a part of the infrastructure-based IEEE 802.11. The real PSM is implemented on both access point, which is aware of the buffer conditions at all associated mobile stations, and attached wireless hosts which use a sleep/wakeup schedule based on the underlying power saving scheme. However, the NS3 implementation is simplified as depicted in Figure 5.1.

In the NS3 simulation-based testing environment, power saving schedule is built on the access point only to simplify the implementation. The access point beacons regularly with a beacon period of 102.4ms. Several components are added to the AP, as follows:

- **Power saving Buffer**. An extra buffer is used at the AP to keep all the packets that are not delivered to mobile stations whose wireless interfaces are switched off. The buffer has a mapping between receiver address and data packets. Once a host is switched on, all data associated with that host's address is transmitted. The buffer

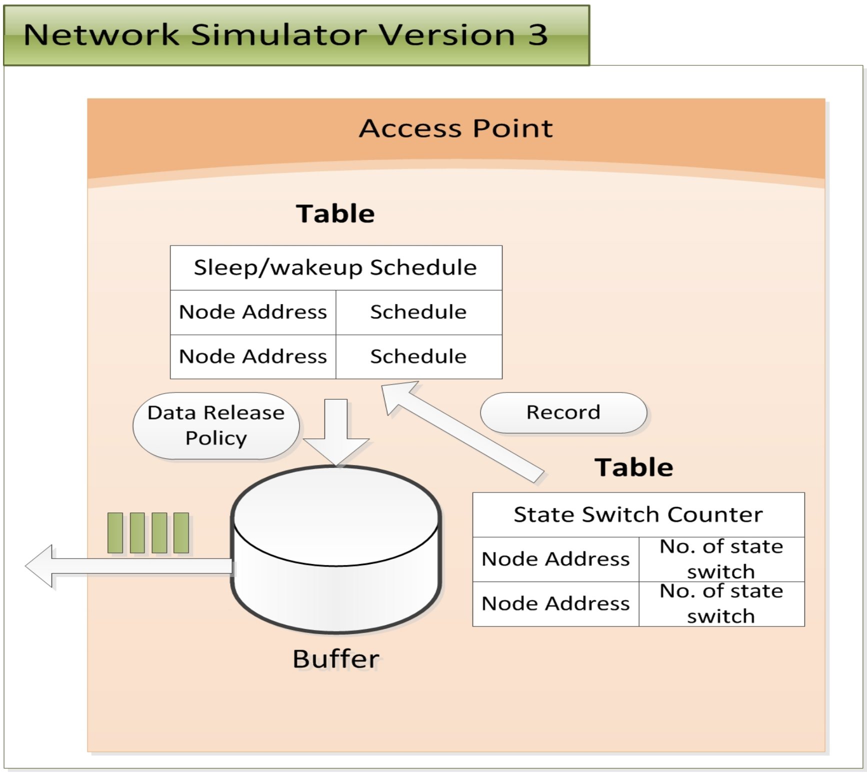


Figure 5.1 Implementation of power saving mode

has a limited size therefore overflow might happen if an inappropriate power saving scheme is employed.

- **Sleep/wakeup Table**. The sleep/wakeup table is maintained by AP and is used for recording the next waking up schedule. Each entry in the table corresponds to the schedule associated with one node address. A typical entry includes intervals left until next waking up, sleeping window, threshold of sleeping window etc.
- **State Switching Recorder Vector**. This is a variable vector which stores the number of state switching associated with each host and it is located at the AP. This information is used to calculate the energy consumption at the end of simulation.

5.1.2.3 Energy Consumption Assessment

NS3 has a built-in energy model which records the time spent in each state and based on the consumption rate in each state it calculates the total energy consumption. Four energy states are considered in the energy measurement: receiving, transmitting, sleeping and state switching. However, as previously stated, power saving mode is not supported in NS3, which means a new energy model with support for the sleeping mode and state switching needed to be implemented. In the extended energy model, the total time spent transmitting and receiving is recorded in the same way as in the built-in model, while the time spent performing state switching is calculated as the number of mode transitions multiplied by the average duration of a transition. The total number of state switches is then recorded by the AP in its sleep/wakeup table for all attached mobile stations. The total sleeping interval is calculated as the total simulation time minus the time spent in other states. Finally, at the end of the simulation, the total energy consumption is calculated.

Energy consumption as stated earlier is decomposed into four energy consumption values, based on different states. According to studies made on Proxim RangeLAN2 2.4 GHz 1.6 Mbps PCMCIA wireless interface card [232], for each second, 1.5 W of energy is spent in transmitting mode, 0.75 W in receiving mode, and 0.01W in sleeping mode. State switching lasts on average 2 ms and has an energy consumption of 0.75 W, as observed by [231]. These values are representative of other wireless network cards as they demonstrate the fact that the energy spend on sleeping mode is far less than for other modes. The basic simulation parameters used in the proposed solutions, IEEE 802.11 and IEEE 802.16 are listed in 5.1.

The other parameters chosen for the testing of the proposed schemes are used to show the trends of their impact on the performance of the solutions compared. For example, three data rates are used in the simulation to show the impact of the data rate on the performance of the solutions compared.

5.1.2.4 Quality Level Assessment

There are two major methods commonly used for delivery quality measurement when performing NS3 simulations. The first one uses NS3 trace files which document every event happened during the simulation process. Important information such as the size of

Table 5.1 Common Simulation Parameters

Parameter	Value
Queue	DropTail
DIFS	60 sec
SIFS	16 sec
Slot time	9 sec
Beaconing period	102.4 ms
Transmitting energy	1.5 W
Receiving energy	0.75 W
Sleeping energy	0.01W
Switching energy	0.75 W
Switching duration	2 ms

each packet, the time a packet is transmitted and received at for each intermediate nodes are post-processed using scripts (e.g. AWK etc) to compute quality metrics such as packet delay, jitter, loss, throughput, etc. The other method is more straightforward. It utilizes the flow monitor which is a built-in component of NS3. The flow monitor generates an XML file at the end of the simulation, which already includes the total packet delay, the number of received and lost packets for each data flow from source to destination, etc. This tool is extremely useful when assessing packet delay and throughput. However, it does not provide enough detail such as time stamps in some cases, for example when analyzing packet jitter or when the playback window size is needed. Next quality measurement metrics used in this thesis are briefly defined:

- **Average packet delay**. Delay in this thesis refers to one way delay defined as the time between the moment when a bit is sent from the source and the time when the bit reaches its destination. As the proposed power saving solutions do not affect the time it takes to send packet from client to server, average packet delay in the simulation only measures one way delay from the server to the client.

- **Packet loss rate**. Loss rate is defined as the rate of dropped packets divided by the total number of packets transmitted by source. This metric is measured as packets may get lost if inappropriate sleeping schedule is adopted leading to long sleep intervals and buffer overflow at the access point.

- **Packet jitter**. Packet jitter is the average of the deviations from the network mean latency. It is an important QoS parameter especially for multimedia streaming applications as it reflects the smoothness and continuity in playback.

- **Playback buffer size**. This metric is used for multimedia streaming-based applications only, and it shows the changes of the playback buffer size while the multimedia clip is being played by the user. Non-empty playback buffer is a crucial component if smooth playback is required.

- **Peak signal-to-noise ratio**. Peak signal-to-noise ratio (PSNR) is a metric used to approximate human perception of reconstruction quality. PSNR value is calculated based on the maximum bit rate of the transmitted content, the expected throughput and the actual throughput. Scaling from 0 to 100, higher PSNR values generally indicate better perceived quality.

- **R-factor**. R-factor is a metric used for quantitative assessment of the quality of VoIP communications. Scaling from 0 to 100, R-Factor is a more precise tool for measuring voice quality than some other existing tools, such as Mean Opinion Score (MOS) [66].

A subset of these quality metrics are chosen to be measured during static STELA, dynamic STELA and Q-PASTE testing respectively, depending on which of them are more interesting to the solution assessed. For example, Q-PASTE is more interested in the playback buffer size than the packet delay, as smooth playback can be provided as long as the buffer is not drained empty by the multimedia player.

5.1.2.5 Simulation Topology

A simple simulation topology generally used for broadband wireless testing is shown in Figure 5.2. It represents the basic network topology built to validate the benefits achieved by a proposed solution, using NS3 as modelling platform. However, there exist some minor differences with the different scenarios tested for the three proposed solutions, respectively.

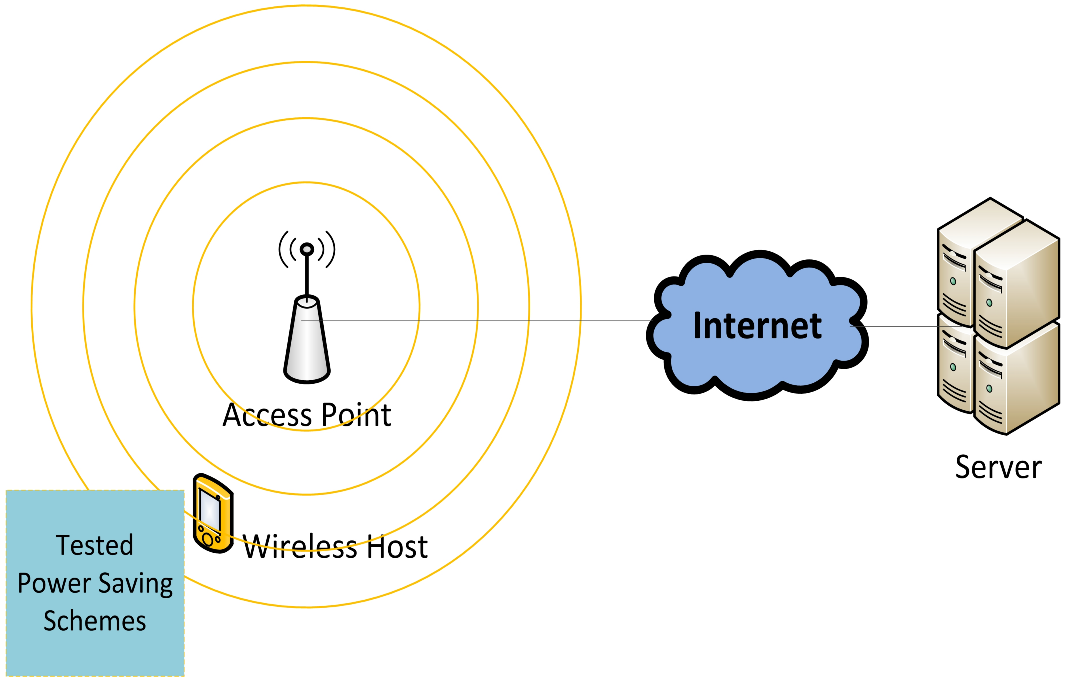


Figure 5.2 Basic simulation topology

The simulation scenario involves one server and one client wirelessly connecting to the access point. The source pushes data periodically via the network to the sink. It can be seen as a pair of server and client or two wireless hosts connecting with each other. STELA is deployed at the MAC layer. The MAC operates at 11Mbps with MIN-CW 15, MAX-CW 1023, SIFS 16 sec and 9 sec slot time.

Client-server network is applied with the coordination of access point in the simulation, which can be seen in Figure 5.3. The client side is a wireless node which requests content from the server, and the server responds with traffic with different intervals. Both energy consumption and quality of service levels are measured at the client side. Next the simulation settings for each of the proposed solutions are presented in details.

5.2 Static STELA Testing

5.2.1 Simulation Test-bed Setup

The principle of the proposed slow start, exponential and linear increase algorithm (STELA) is to allow the wireless network interface adopt a wise sleep/wakeup schedule so that the sleeping period is increased, in order to save energy. The algorithm is employed by the MAC layer at the client side, which sends out requests and receives response packets from the server.

In Figure 5.3, two mobile stations are wirelessly connected to the access point. The source pushes data periodically via the network to the sink. It can be seen as a server/client pair communicating with each other. At the source side, we change the traffic pattern with different inter-burst intervals, replicating scenarios associated with most applications such as multimedia streaming with on/off traffic patterns. The AP beacons periodically to mobile stations with an interval of 102.4 milliseconds. AP buffers data if the receiver is in sleeping mode and forwards all buffered data once the receiver switches on. At the sink side, i.e. the wireless client, static STELA is employed at the MAC side for wise WNIC scheduling. Details about static STELA are available in the previous chapter.

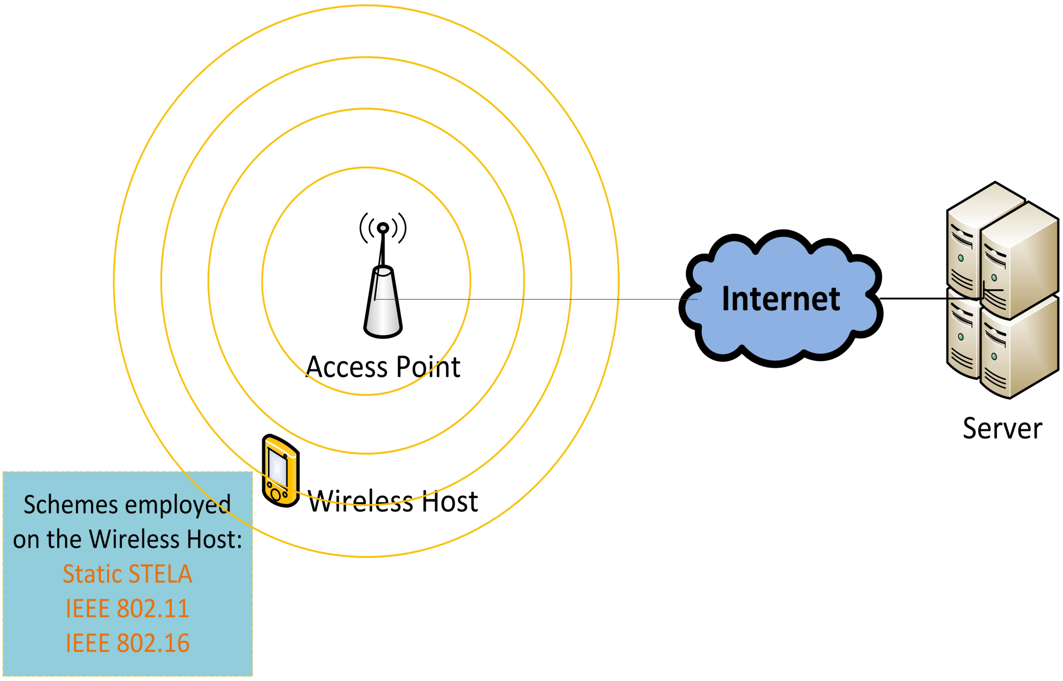


Figure 5.3 Static STELA simulation topology

As seen in Figure 5.3, the performance of static STELA is compared with the Power Saving Mechanism (PSM) scheme used by the IEEE 802.11 and IEEE 802.16. Unlike STELA, the sleeping strategies used by the IEEE 802.11 and IEEE 802.16 standards do not consider the type and particularities of data traffic. IEEE 802.11 adopts a fixed sleep/wakeup schedule where the wireless interface wakes up regularly to sample the wireless channel. IEEE 802.16 employs the binary exponential increase algorithm where the size of the sleeping window is doubled if no packets are detected during radio channel sampling until it reaches the maximum value of W_{thre}. In these tests both energy consumption and quality metrics, which include QoS (i.e. packet delay and delay jitter) and QoE (i.e. PSNR), are assessed.

The solutions compared in the testing are listed in Table 5.2. W_{thre} is a variable that determines the maximum sleeping window size of the power saving scheme used in IEEE 802.16. It is also required in the configuration of static STELA as it defines the boundary between exponential increase and linear increase phases.

Table 5.2 Schemes Compared in Validation of Static STELA

Scheme	Variables	Description
IEEE 802.11	NA	WNIC wakes up regularly after every beacon interval
IEEE 802.16	W_{thre}	Sleeping window size of WNIC grows exponentially
Static STELA	W_{thre}	Sleeping window size adapts to real time traffic, WNIC could be in one of the three phases: slow start, exponential increase or linear increase

5.2.2 Test Cases

Several test cases are considered in order to assess the benefits of STELA under different networking configurations, as shown in Table 5.3.

The simulation consists of a total of 240 sets of test cases which differ from each other by varying the following parameters:

- **Traffic** refers to types of data traffic used at the application layer. The simulation-based testing is performed using both Constant Bit Rate (CBR) and Variable Bit Rate (VBR) traffic shapes which are widely used by applications involving multimedia content delivery. The proposed solutions are tested over UDP instead of TCP, as speed is more important than reliability for most multimedia applications.

- **Type** refers to traffic patterns. For both CBR and VBR traffic, three different traffic patterns are compared as outlined in Figure 5.4 and Figure 5.5, respectively. These patterns have different on/off intervals and traffic rates to validate the performance of STELA under different application configurations. The on/off periods of the CBR traffic considered are 20s on/20s off, 10s on/20s off, and 20s on/10s off, respectively. Traffic with 0.01s on/0.01s off, 0.02s on/0.01s off and 0.01s on/0.02s off patterns are set for the VBR traffic.

- **Data rate** refers to the data rate generated by the server. The traffic data rate is set to

Table 5.3 Testing Parameters– Static STELA

Parameter	Value
Simulation Duration	200s
Traffic	CBR VBR
Rate (CBR)	20s on/20s off 10s on/20s off 20s on/10s off
Rate (VBR)	0.01s on/0.01s off 0.02s on/0.01s off 0.01s on/0.02s off
Data rate	0.5 Mbps 1.0 Mbps 1.5 Mbps
W_{thre}	2 4 8 16

0.5 Mbps, 1.0 Mbps and 1.5 Mbps respectively in distinct test cases. Two staircase patterns are additionally considered for both CBR and VBR traffic with a step rate of 0.5 Mbps and 200s duration for each step in order to test the impact of fluctuation in data rate on the delivery performance.

- **Threshold** refers to the threshold values used by STELA. As one of the most important factors in tuning STELA's behaviour, the threshold value dictates when the exponential increase phase terminates and the linear increase phase starts. The threshold values assigned for each of the test cases are 2, 4, 8, and 16, respectively. These values represent multiples of beacon time intervals.

These four test scenario parameters form a vector <traffic, type, rate, threshold>, and only one of the vector's values is changed in every individual test case.

Figure 5.4 Illustration of traffic patterns: Type 1, 2 represent on/off CBR traffic, (20s on, 20s off) and (10s on, 20s off) respectively. Type 4, 5 represent on/off VBR traffic, (0.01s on, 0.01s off) and (0.02s on, 0.01s off) respectively.

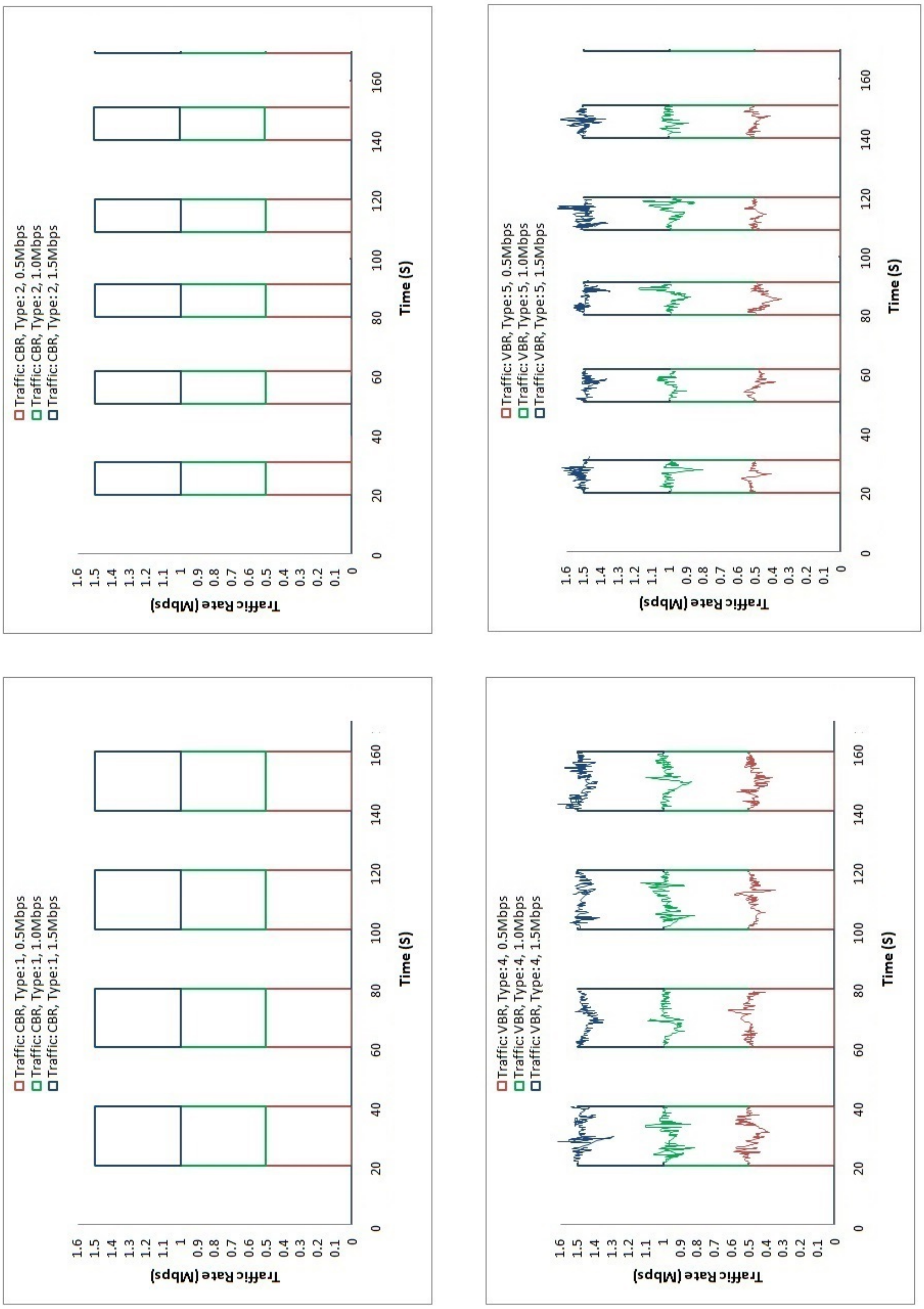
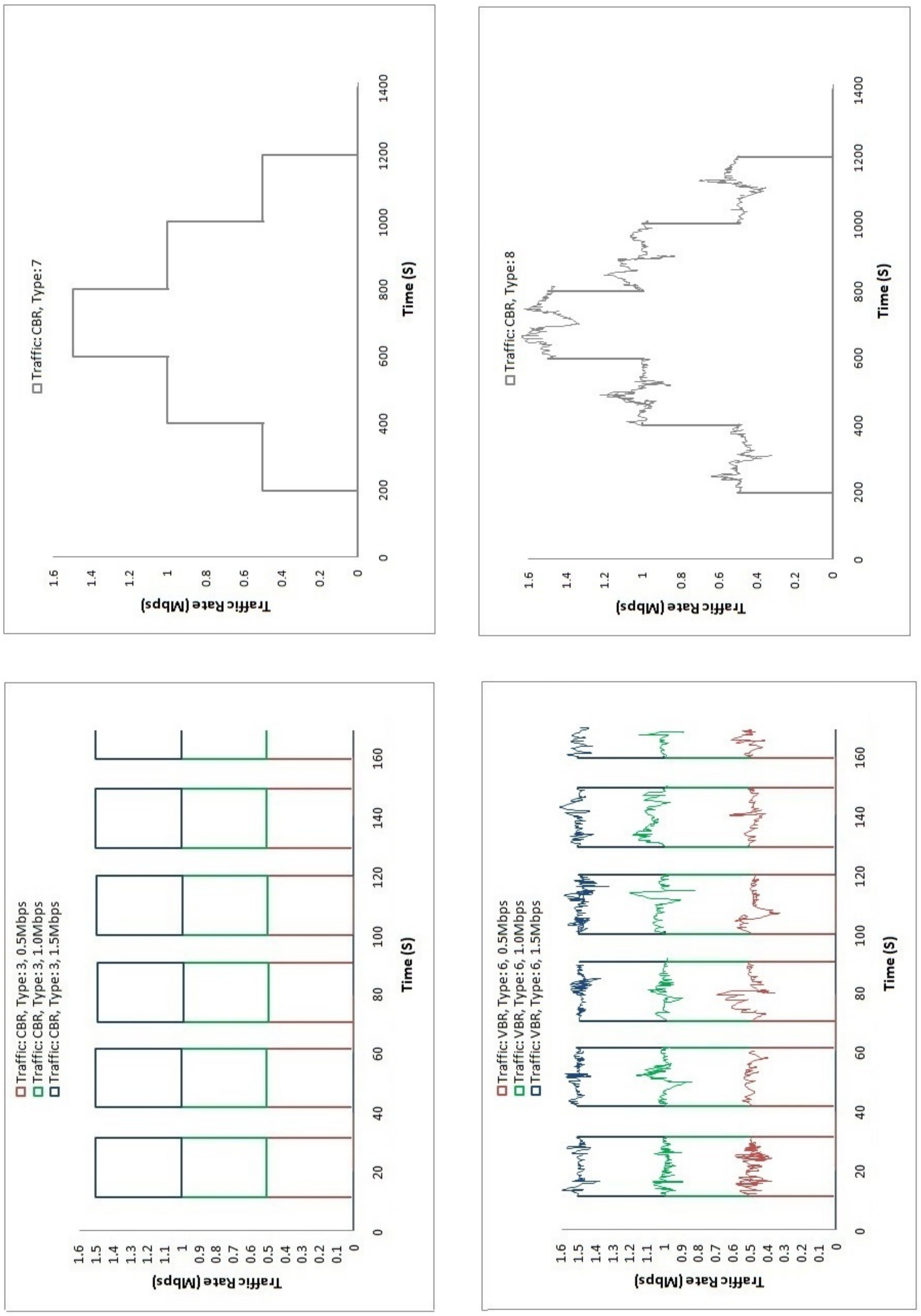


Figure 5.5 Illustration of traffic patterns: Type 3 represents on/off CBR traffic (20s on, 10s off), and Type 7 is staircase type CBR traffic with 200s duration and a rate variation of 0.5Mb for each stair. Type 6 represents on/off VBR traffic (0.01s on, 0.02s off), and Type 8 is a staircase type VBR traffic with 200s duration and a rate variation of 0.5Mb for each stair.



5.3 Dynamic STELA Testing

5.3.1 Simulation Test-bed Setup

The goal of the dynamic slow start, exponential and linear algorithm , i.e. dynamic STELA, is to provide a balanced performance between energy efficiency and QoS through wise WNIC scheduling and traffic prediction. The proposed solution mainly involves two components: an application monitor at application layer, and STELA at MAC layer, both of which require client-side modification only.

Similar with static STELA, the performance of dynamic STELA is tested and compared with those of IEEE 802.11 and IEEE 802.16. The three algorithms are individually deployed for each traffic pattern. IEEE 802.11 refers to the fixed sleeping interval scheme where the radio transceiver is powered on for each beacon interval. IEEE 802.16 refers to the binary exponential increase algorithm where the size of the sleeping window is doubled if no packets are detected during radio channel sampling until it reaches the maximum value of W_{thre}. Energy consumption, QoS levels including packet delay and delay jitter, and QoE levels in terms of PSNR are analyzed based on the trace files generated by the simulator.

The solutions tested and compared are listed in Table 5.4. Different from static STELA, W_{thre} does not require configuration as it is self adaptive to real time traffic based on the parameter tuning process.

5.3.2 Test Cases

The test cases used for assessing the performance of dynamic STELA is divided into two major scenarios: single server which involves one server/client pair and multiple servers which represents the secenario in which one client communicates with two servers at the same time. Next the two scenarios will be discussed with more details.

Table 5.4 Schemes Compared in Validation of Dynamic STELA

Scheme	Variables	Description
IEEE 802.11	NA	WNIC wakes up regularly after every beacon interval
IEEE 802.16	W_{thre}	Sleeping window size of WNIC grows exponentially
Dynamic STELA	NA	Sleeping window size adapts to real time traffic, WNIC could be in one of the four phases: slow start, exponential increase, linear increase or parameter tuning. Parameter tuning phase is used for data arrival prediction

5.3.2.1 Scenario One – Single Server

Figure 5.6 illustrates the simulation network topology used in scenario one. At the source side, we change the traffic pattern with different intervals between each burst, which is the scenario for most applications such as multimedia streaming with on/off traffic patterns. The AP beacons periodically to mobile stations with an interval of 102.4 milliseconds. It buffers data if the receiver is in sleeping mode and forwards all buffered data once the receiver is switched on. At the client side, the session monitor is deployed at the application layer to monitor all session-related activities and notify MAC layer about all session changes. The four-phase dynamic STELA is employed at the MAC layer which functions similar to the static STELA with an extra parameter tuning phase triggered by the application layer information. The detail description of dynamic STELA is presented in the previous chapter.

Table 5.5 Testing Parameters– Dynamic STELA, Scenario One

Parameter	Value
Simulation Duration	200s
On/Off interval	1s on/ 2s off 2s on/1s off 1s on/1s off
Data rate	0.5 Mbps 1.0 Mbps 1.5 Mbps
W_{thre}	2 4 8 16

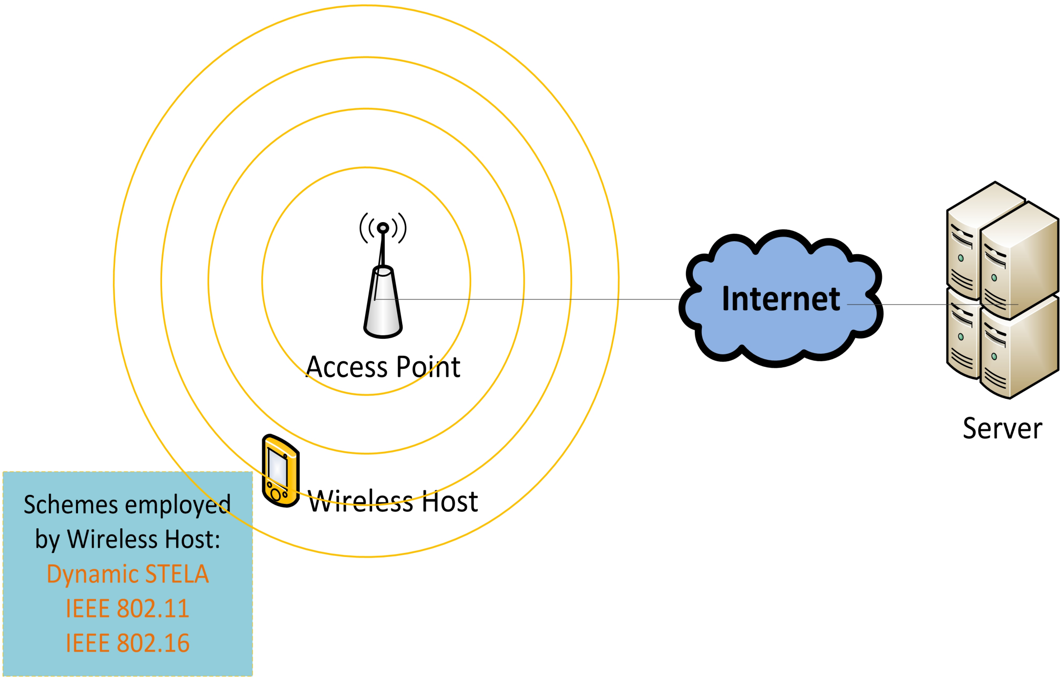


Figure 5.6 Dynamic STELA simulation scenario one: single server

The simulation-based testing consists of testing of dynamic STELA, IEEE 802.11 and IEEE 802.16 with different network parameter settings including:

- **Traffic** refers to the different data arrival patterns. It is essential in the testing of dynamic STELA as the parameter tuning phase is dependent on the inter-burst intervals. Three traffic patterns are individually tested: 1s on/ 2s off, 2s on/1s off, 1s on/1s off.

- **Data rate** refers to the data rate generated by the server. Three values are used in the testing: 0.5 Mbps, 1.0 Mbps and 1.5 Mbps.
- **Threshold** refers to the maximum sleeping window size of WNIC allowed. It is applicable to IEEE 802.16 only. Four values for W_{thre} are respectively tested, 2, 4, 8, 16. For example, 802.16-2 denotes maximum sleeping window of 2 beacon intervals and 802.16-16 represents a maximum window of 16 beacon intervals.

The three parameters above are changed in different testing scenarios and one of them is changed in each single test case only. The simulation parameters are listed in Table 5.5.

5.3.2.2 Scenario Two– Multiple Servers

The second scenario involves the situation where the client is communicating with more than one server at a time, as shown in Figure 5.7. Server 1 is transmitting data to the client during the whole process of simulation, while server 2 starts data transmission in the middle of the simulation and terminates before the simulation is complete.

Traffic arrival pattern changes when the client starts communicating with a new server, which indicates the old threshold value could not provide accurate prediction of the arrival time of the next data burst. However, as the threshold adjusting phase is triggered whenever an application session is established or is terminated, dynamic STELA is capable of predicting the future traffic arrival pattern even if the pattern changes. More specifically, the parameter adjusting phase is initiated both when the data flow between server 2 and the client starts and when that flow ends.

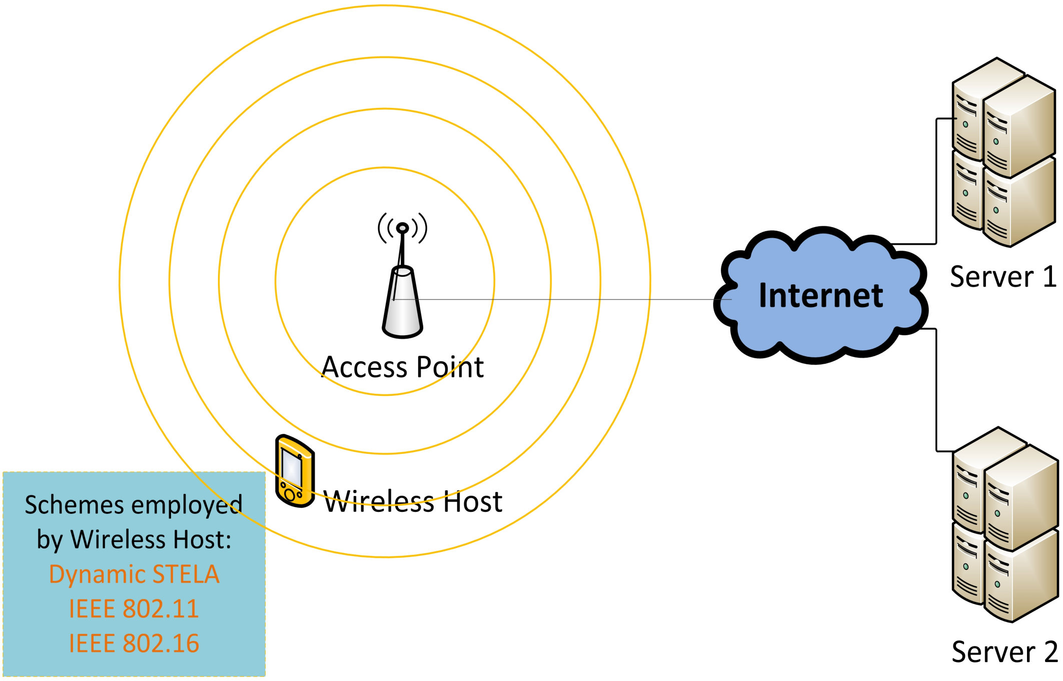


Figure 5.7 Dynamic STELA simulation scenario two: multiple servers

The simulation consists of testing of dynamic STELA, IEEE 802.11 and IEEE 802.16 with different network parameter settings. The parameters varied during the simulations are:

- **Traffic** refers to the different data arrival patterns. For both data flow between the client and the servers, three traffic patterns are individually tested: 1s on/ 2s off, 2s on/1s off, 1s on/1s off.
- **Data rate** refers to the data rate generated by the server. The impact of data rate on the performance of the compared schemes are illustrated in scenario one, and therefore only one data rate, i.e. 0.5Mbps is evaluated in scenario two.
- **Threshold** refers to the maximum sleeping window size of WNIC allowed. It is only applicable to IEEE 802.16. The largest and smallest values for W_{thre} used in scenario one are tested only, e.g. 802.16-2 denotes maximum sleeping window of 2 beacon intervals and 802.16-16 represents a maximum window of 16 beacon intervals.

Table 5.6 Testing Parameters– Dynamic STELA, Scenario Two

Parameter	Value
Case 1	On/Off interval: 1s on/ 1s off Server 2: Starting time: 10s, Ending time: 50s
Case 2	On/Off interval: 2s on/ 1s off Server 2: Starting time: 50s, Ending time: 90s
Case 3	On/Off interval: 1s on/ 2s off Server 2: Starting time: 100s, Ending time: 150s
Data rate	0.5 Mbps
W_{thre}	2 16
Server one	Starting time: 0s Ending time: 200s

- **Transmitting time** refers to the starting point and ending point of the data flow between the client and each server. Server one works the same way as in scenario one, which runs for 200 seconds, while server two keeps sending data for a shorter period, starting from different time during the simulation. Server 2 joins the communication from 10s to 50s, from 50s to 90s and from 100s to 150s respectively, as indicated in case 1, 2 and 3. The fluctuation of data arrival pattern is used to illustrate its impact on the threshold adjusting phase.

Testing parameters used for scenario two are listed in Table 5.6.

5.4 Q-PASTE Testing

5.4.1 Simulation Test-bed Setup

In order to further prolong sleeping duration of the WNIC without compromising user quality of experience levels, the cross-layer solution Q-PASTE was proposed to allow application layer data shaping to work along with adaptive MAC layer sleep/wakeup scheduling. In order to validate the performance of Q-PASTE, both the client host and the service gateway were modified, as described in the previous chapter.

The network topology used for Q-PASTE testing is shown in 5.8. The gateway beacons regularly to the connected wireless host, the same as any standard AP. However, the application-layer data shaping buffers the received packets for a while until they are released to the station. Also, in order to provide smooth playback at the client side, fast start is also enabled at the service gateway. At the client side, similar application-layer session monitor and MAC-layer STELA is deployed as was used for dynamic STELA.

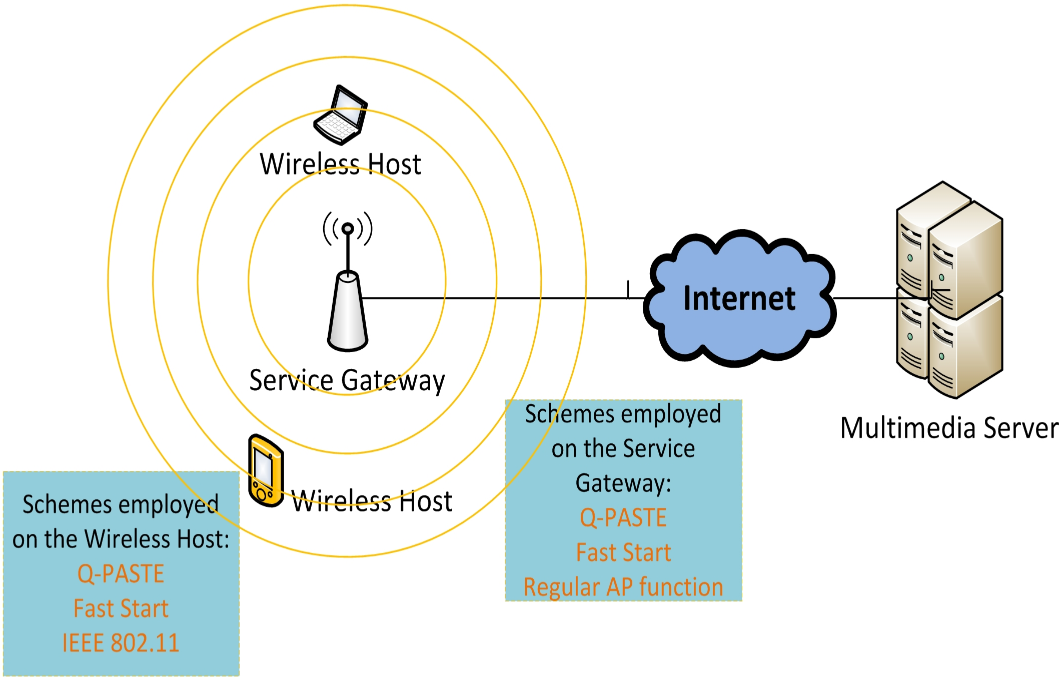


Figure 5.8 Q-PASTE simulation topology

Q-PASTE is evaluated and compared with the IEEE 802.11 PSM and a cross-layer energy saving algorithm proposed in [204], denoted as Buffered Streaming algorithm in this paper. The solution proposed in [204] employs at the proxy a traffic shaping algorithm which adopts a similar fast streaming technique as described in our solution, and at the client side PSM, as proposed in IEEE 802.11. The solution proposed in [204] was chosen as the baseline for comparison as it is a cross-layer solution which employs application

Table 5.7 Schemes Compared in Validation of Q-PASTE

Scheme	Description
IEEE 802.11	WNIC wakes up regularly after every beacon interval
Buffered Streaming	The playout buffer is filled quickly at the beginning during the fast start period. It will be always filled before being drained empty later on.
Q-PASTE	Enables fast start, and also employs smart scheduling of the WNIC through data prediction

layer data buffering and MAC layer power saving scheme, similar to the proposed solution Q-PASTE. Although there are other cross-layer energy efficient solutions, they are not necessarily application-MAC solutions. Both energy consumption and client-side quality levels are evaluated. The tested solutions are listed in Table 5.7.

5.4.2 Test Cases

The experiments mainly vary two parameters at the gateway:

- **Encoding rate** refers to the encoding rate of the multimedia clip. We conduct experiments on three streams with various encoding rates representing three different Internet radio stations [204] all using the ITU-T R. G.711 codec. All streams are 400s long. The encoding rates of these streams are 8 kbps, 16 kbps, 24 kbps for stream 1, 2 and 3, respectively, as shown in Table 5.8.

- **Buffering period** refers to the maximum duration that the data is buffered at the gateway before being relayed to the client. The buffering period is set from 1 s to the maximum buffering period $T_{bf_{min}}$, value of which is calculated for each stream individually.

There are two reasons why only audio is tested in the simulation. First, the major difference between audio and video is that video content playing normally indicates higher data rate than audio. As several audio streams with different data rates have been tested to demonstrate the impact of increasing data rate on the testing results, the results for video content should follow the same trend. The main difference would be that video content would consume more energy as more packets are received by the client. Second, the tested audio content is the same data as used in [204]. The authors of [204] did not include any video sample, and therefore it is hard to obtain the statistics needed for video testing.

In the experiments, we study the impact of the buffering period on energy efficiency, playout buffer size. Also we study the impact of playback start time on playout buffer size. Finally, we evaluate the estimated user quality experience by comparing R-factor values. The testing parameters are shown in 5.8.

Table 5.8 Testing Parameters– Q-PASTE

Parameter	Value
Stream length	400s
Encoding rate (stream 1)	16 kbps
Encoding rate (stream 2)	8 kbps
Encoding rate (stream 3)	24 kbps
Buffering period (stream 1)	2s 12s 25s
Buffering period (stream 1)	2s 25s 50s
Buffering period (stream 2)	2s 8s 16s

5.4.2.1 Scenario One – Immediate Playback

Different scenarios affect the playout buffer size significantly, hence two major scenarios are considered in the test cases. The first one, as illustrated in Figure 5.9, refers to the situation that the user starts the playback immediately after the connection between server and client is established, without waiting for the playout buffer to be fully filled.

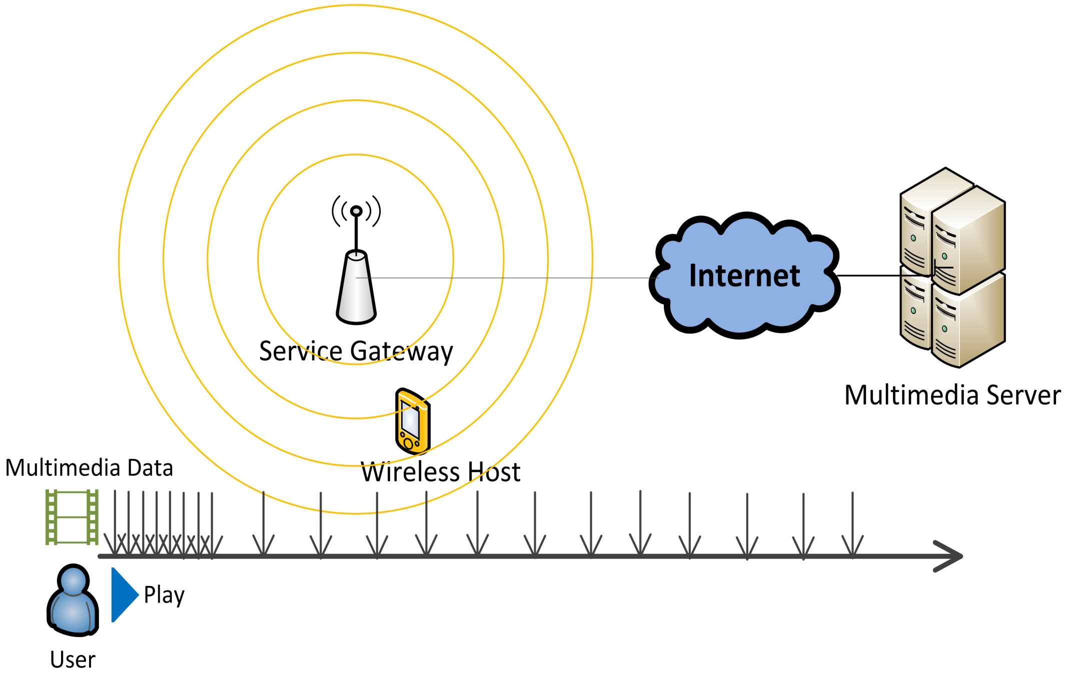


Figure 5.9 Q-PASTE simulation scenario one: immediate playback

In this scenario, from the beginning of the streaming process, the playout buffer is being drained by the multimedia player while is also being filled by the packets from the server simultaneously. In this situation, the number of packets stored in the playout buffer is relatively lower than in the other cases.

5.4.2.2 Scenario Two – Delayed Playback

The second scenario considered the situation when the user starts the playback after the fast streaming period is over, as shown in Figure 5.10. In this case, the playout buffer size is filled quickly during the fast streaming period, and the average buffer size remains higher than in the previous scenario.

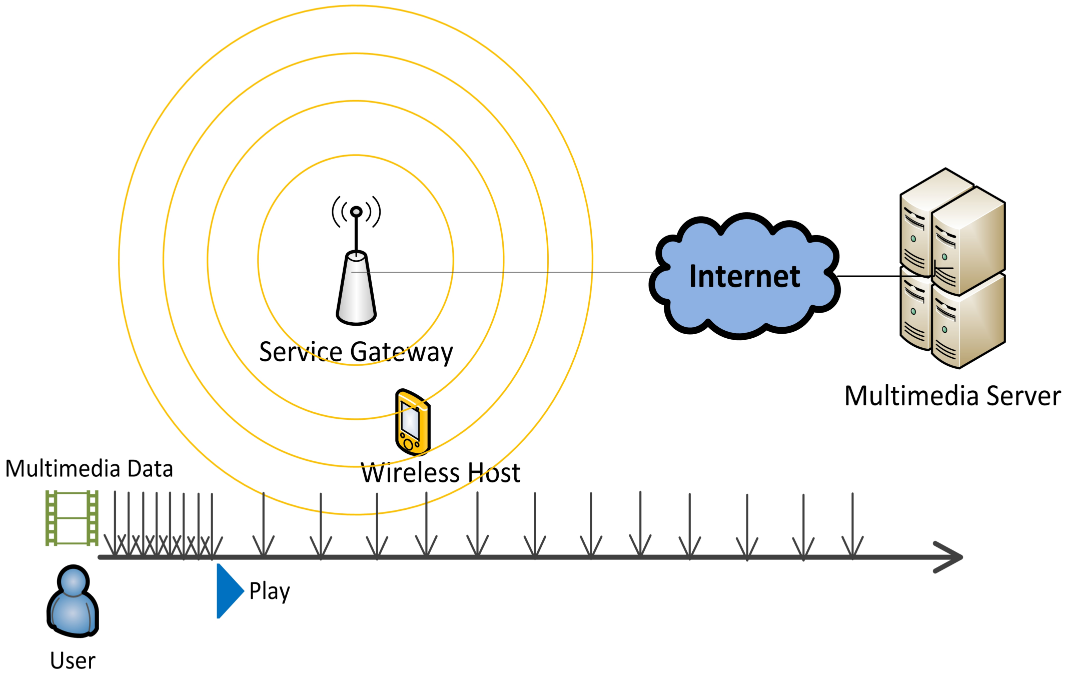


Figure 5.10 Q-PASTE simulation scenario two: delayed playback

5.5 Summary

This chapter has introduced Network Simulator version 3 (NS3) as modelling and simulation tool used in this thesis. Next the additional modules which were implemented for the purpose of enabling power saving scheme, due to the limitation of NS3 were described . The metrics used for performance measurement are then presented, including both energy efficiency and quality of service metrics. A common network topology is illustrated.

For each of the proposed solutions, i.e. static STELA, dynamic STELA and Q-PASTE, the simulation test-bed setup and testing scenarios are described in details, respectively. As different simulation parameters and system configurations are required for each individual scheme, the parameters for each scheme are explained, respectively. The existing solutions chosen to be compared with the proposed solutions are also presented followed by the testing scenarios used during testing of each solution.

Chapter 6

Testing Results and Analysis

This chapter presents the results obtained from the simulation-based testing of the proposed solutions. The experimental results for static STELA, dynamic STELA and Q-PASTE are presented, and the performance of the proposed solutions are compared and analyzed with existing schemes. Energy consumption is measured for all the proposed solutions, and different metrics are used in the evaluation of quality. For both static STELA and dynamic STELA, these metrics include packet delay, jitter and PSNR, and for Q-PASTE quality of experience and R-factor as an indicator of user experience level. Testing results show how important benefits are achieved by the proposed solutions, in comparison with existing state-of-the-art mechanisms.

6.1 Static STELA

The performance of static STELA is compared with the that of Power Saving Mechanism (PSM) scheme used by the IEEE 802.11 and the binary exponential increase function employed by the IEEE 802.16. Unlike STELA, the sleeping strategies used by the IEEE 802.11 and IEEE 802.16 standards do not consider the type and particularities of data traffic. In these tests both energy consumption and network quality parameters (i.e. packet delay, delay jitter and PSNR) are assessed in different scenarios.

The testing scenarios consider four parameters: traffic pattern, traffic type, data rate and threshold, and the testing results have shown the effect of each parameter variation on the performance of the three compared algorithms. Testing results obtained under different parameter configurations show great savings in terms of energy consumption and also good levels of QoS achieved by using static STELA, when compared with the other two power saving schemes employed by IEEE 802.11, IEEE 802.16.

The energy consumption of the three schemes evaluated in all test cases are graphically presented in Figure 6.1 and Figure 6.2, except the staircase traffic patterns which are proved to show similar results with the other traffic types. It can be observed that STELA saves up to 55% energy for CBR traffic when compared with IEEE 802.11 (fixed window) and up to 36% when compared to IEEE 802.16 (exponentially increasing window). In the context of VBR traffic, it can be observed that STELA consumes up to 50% less energy than the constant window algorithm (IEEE 802.11), and up to 18% less when compared with the exponential increasing window scheme (IEEE 802.16).

On one hand, although both average packet delay and jitter slightly increase when using STELA, their values are less than 25 ms, which is considered acceptable for multimedia content delivery [233], thus not compromising the delivery performance from the user perspective.

Moreover, in order to assess the user quality of experience levels, PSNR metric is used to estimate the delivery quality. It is calculated as in eq. (6.1).

$$PSNR \quad = \quad 20 \cdot \log_{10}\left(\frac{255}{\sqrt{MSE}}\right) \tag{6.1}$$

where MSE represents the Mean Square Error and is defined as the cumulative squared error between the original and the processed data. There are various different approaches in defining PSNR in literature. In this thesis, the PSNR value is calculated according to eq. (6.2) [234], where *MAX_BitRate* is the maximum bit rate, *Exp_Thru* is the expected throughput, and *Thru* is the actual throughput. Higher PSNR values indicates better quality of service. In order to measure the actual throughput, the data packets that arrive at the client with delay longer than 25ms are considered lost [233].

$$PSNR \;=\; 20 \cdot \log_{10}\left(\frac{MAX\_BitRate}{\sqrt{(Exp\_Thru - Thru)^2}}\right) \tag{6.2}$$

On the other hand, this algorithm can also be deployed in application contexts which are less sensitive to delay and jitter, e.g. web applications.

Next the influence of the four parameters on the compared schemes are analyzed in turn.

6.1.1 Impact of Traffic Pattern and Type on the Performance of Static STELA

Traffic pattern refers to the manner in which the traffic varies in time where type variation is limited to either CBR or VBR traffic. It can be observed from the testing results that when the traffic pattern varies, energy consumption and performance results differ.

For both CBR and VBR traffic types, the longer the burst period and the shorter the idle period, the more energy is consumed. For example, with the same data rate, traffic type 1 refers to the situation where the data arrival pattern is split into two parts of the same length , i.e. 20s on/20s off, which leads to moderate energy efficiency. Traffic type 2 refers to the type of longer off period and shorter on period, resulting in increased energy savings, while in traffic type 3 the on period is longer than the off period leading to lower energy conservation levels.

When we consider an individual scheme, it can be seen that with the same threshold value, traffic type 2 consumes the least energy as the WNIC spends the least time on data transmission. Moreover, with regards to comparison among the three schemes, the longer the off period, the more energy is saved by static STELA. Table 6.1 presents the

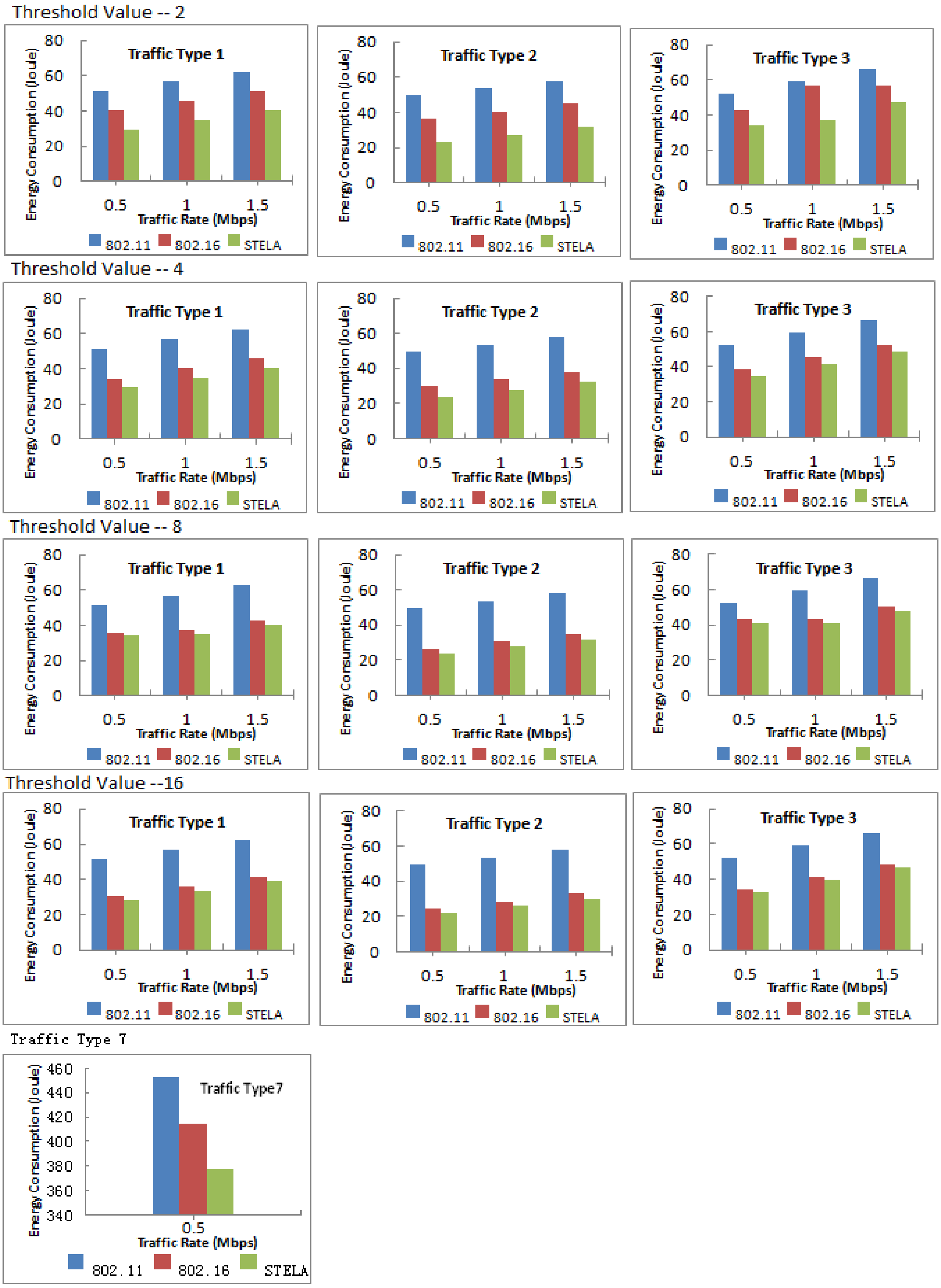


Figure 6.1 Energy consumption when CBR traffic is generated.

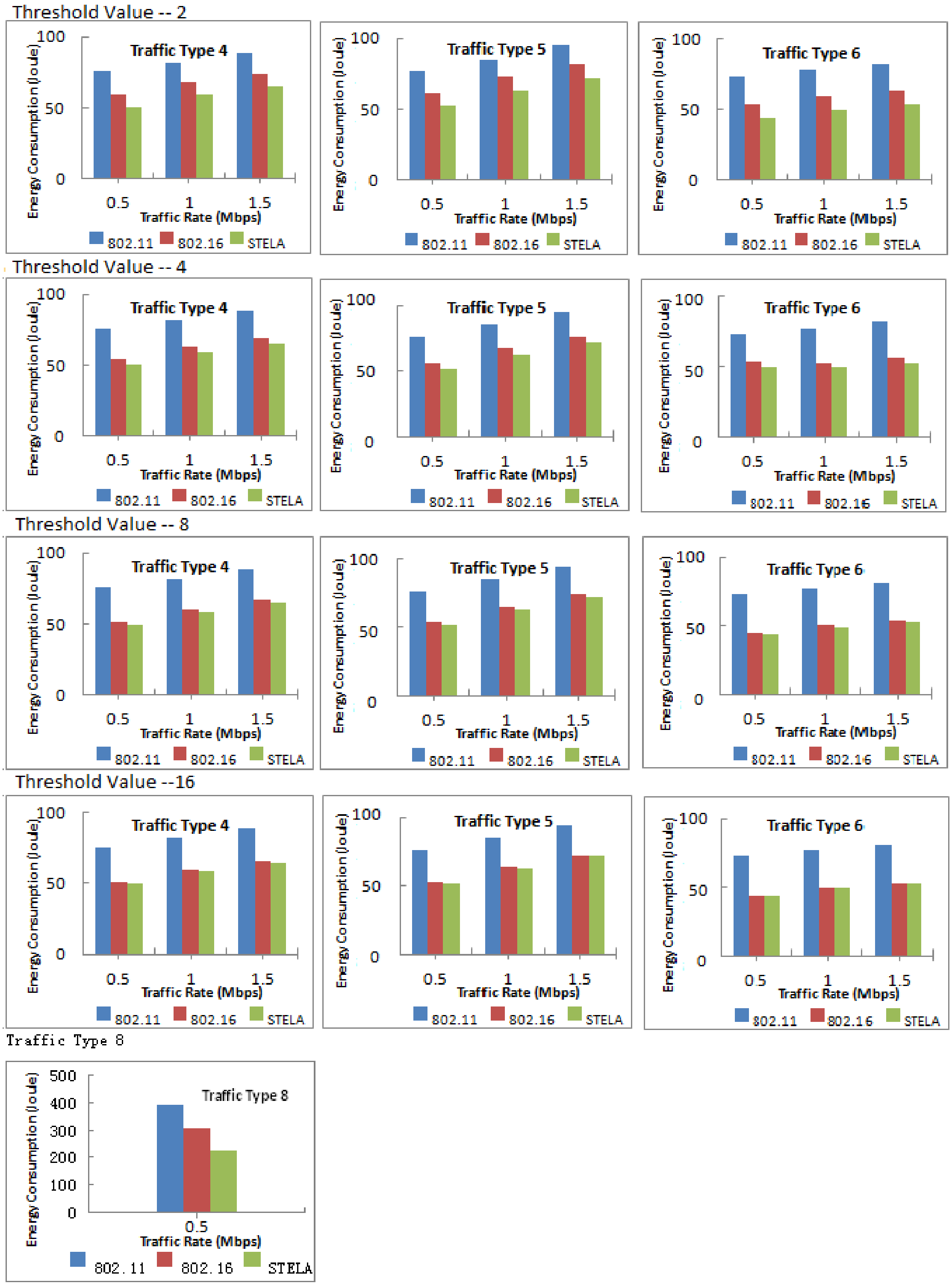


Figure 6.2 Energy consumption when VBR traffic is generated.

Table 6.1 Impact of Traffic Pattern on Static STELA–Data Rate: 1.5Mbps, Threshold:16 Beacon Intervals

Traffic Pattern	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)	PSNR (dB)
Type 1	Fixed	62.58	14.61	5.47	47.96
	Exponential	41.34	14.74	5.50	31.48
	Static STELA	39.12	15.68	5.68	29.54
Type 2	Fixed	57.81	14.88	5.47	37.50
	Exponential	32.98	14.87	5.53	31.06
	Static STELA	30.35	16.43	5.84	27.96
Type 3	Fixed	66.35	14.33	5.44	39.44
	Exponential	48.43	14.83	5.52	31.48
	Static STELA	46.40	15.57	5.67	31.27

performance of the compared schemes with data rate of 1.5Mbps and threshold value of 16 beacon intervals. It can be seen that the energy saving achieved by static STELA is 37.49% for type 1, 47.50% for type 2, and 30.07% for type 3, compared with IEEE 802.11, and 5.38%, 7.98% and 4.2% for type 1, 2, 3 respectively, compared with IEEE 802.16. This is mainly due to the fact static STELA adapts its sleeping window dynamically and therefore longer interval between bursts allows longer sleeping period and less state switching of WNIC. On the other hand, longer delays and lower PSNR values are introduced as the size of the WNIC's sleeping window is getting larger with the increase of the off period.

6.1.2 Impact of Data Rate on the Performance of Static STELA

Energy consumption increases with the increase in data rate, for all the schemes compared, which can be seen from the testing results as shown in Figure 6.1 and Figure 6.2. An example is given in Table 6.2 to demonstrate the impact of data rate on the performance of the compared schemes witH traffic type 1 generated from the server and threshold value set to 16 beacon intervals. The energy consumption increases by 11 Joule for IEEE 802.11, IEEE 802.16 and static STELA, when the data rate increases from 0.5 Mbps to 1.5 Mbps.

Among all the modes of WNIC, transmitting and receiving modes consume the most

Table 6.2 Impact of Data Rate on Static STELA–Type: 1, Threshold:16 Beacon Intervals

Data Rate	**Scheme**	**Energy (Joule)**	**Delay (ms)**	**Jitter (ms)**	**PSNR (dB)**
0.5Mbps	Fixed	51.38	18.02	5.76	31.70
	Exponential	30.22	18.37	5.87	31.06
	Static STELA	28.48	21.18	6.39	27.96
1.0Mbps	Fixed	56.98	14.98	7.61	36.48
	Exponential	35.78	15.9	7.72	33.98
	Static STELA	33.80	16.61	7.98	29.12
1.5Mbps	Fixed	62.58	14.61	5.47	47.96
	Exponential	41.34	14.74	5.50	31.48
	Static STELA	39.12	15.68	5.68	29.54

amount of energy, compared with sleeping and idle modes. In this case, high data rate from the server indicates more data to receive at the client side which requires more time spent in the most energy consuming mode. Therefore the energy consumption is increased when the data rate grows. For example, the energy consumption of IEEE 802.11 increases from 51.38 Joule for a data rate of 0.5Mbps to 62.58 Joule for a data rate of 1.5Mbps during a 200s duration of simulation. The same phenomenon can be observed for IEEE 802.16 and static STELA as well.

On the other hand, with the increase of data rate, the associated packet delay and jitter decreases and therefore the PSNR scores increase. The reason behind this is twofold. First, the WNIC spends more time in active mode and therefore less time in sleeping mode, which means the response time of WNIC is quicker if the data rate is higher. Second, the jitter, indicating the interval between received packets, decreases as more packets are generated and received by the client within a single burst. In this case, the user experience of quality levels are increased.

6.1.3 Impact of Threshold Value on the Performance of Static STELA

The experiments show the significant impact of the threshold values, measured in terms of access point beacon periods, on the performance of static STELA. The smaller the threshold value, the more power is saved by static STELA, when compared with IEEE 802.16. The reason behind this is that static STELA activates the linear increase phase and the sleeping window keeps growing afterwards, while IEEE 802.16 does not allow further increase of W_{thre} once the sleeping window size achieves W_{thre}. However, the longer sleeping interval enabled by static STELA the longer the increase in packet delay and jitter and the lower PSNR scores are experienced.

Here an example which shows the results for traffic type 1, 1.5 Mbps is given in table 6.3. For example, the PSNR score for IEEE 802.11 remain 47.96dB for all the tested threshold values, while it decreases from 43.52dB to 29.54dB when the threshold value increases from 2 beacon intervals to 16 beacon intervals for static STELA. However, the degradation of the QoS metric values is acceptable due to the slow growth of the sleeping window allowed by the linear increase phase, as shown in Table 2.6. Similar results should apply to other traffic patterns and types.

On the other hand, the larger the threshold value, i.e. W_{thre}=16, the more likely that static STELA and IEEE 802.16 perform similarly in terms of both energy efficiency and QoS, as the likelihood for a packet to arrive before the binary exponential increase phase terminates is higher. It can be seen that even under the worst case scenario (i.e. threshold value set to 16), STELA performs similarly to the mechanism utilized in IEEE 802.16, while still outperforming IEEE 802.11 in terms of energy efficiency.

In comparison with the IEEE 802.16, threshold value of 2 yields the best performance in terms of energy saving and reduced negative impact on QoS, while a value of 16 reduces the benefit of static STELA. For example, using IEEE 802.11 as the benchmark, STELA saves 18.10% , 8.07% , 3.68% and 2.91% more energy than IEEE 802.16 with threshold set to 2, 4, 8 and 16 beacon intervals, respectively.

Table 6.3 Impact of Threshold Value on Static STELA–Type: 1, Data Rate:1.5Mbps

Threshold Value	**Scheme**	**Energy (Joule)**	**Delay (ms)**	**Jitter (ms)**	**PSNR (dB)**
2	Fixed	62.58	14.61	5.47	47.96
	Exponential	51.33	14.61	5.48	45.46
	Static STELA	40.00	15.27	5.60	43.52
4	Fixed	62.58	14.61	5.47	47.96
	Exponential	45.66	14.66	5.48	40.94
	Static STELA	40.61	14.80	5.51	38.92
8	Fixed	62.58	14.61	5.47	47.96
	Exponential	42.74	14.74	5.50	35.22
	Static STELA	40.45	14.85	5.52	34.58
16	Fixed	62.58	14.61	5.47	47.96
	Exponential	41.34	14.74	5.50	31.48
	Static STELA	39.12	15.68	5.68	29.54

Table 6.4 Testing Results for Static STELA with Threshold Value Set to 2 Beacon Intervals

Traffic CBR	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)	Traffic VBR	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)
0.5Mb Type 1	Fixed	51.38	18.02	5.76	0.5Mb Type 4	Fixed	75.76	16.97	5.76
	Exponential	40.13	18.03	5.76		Exponential	58.97	21.68	12.55
	Static STELA	29.14	19.96	6.17		Static STELA	50.00	21.89	12.55
0.5Mb Type 2	Fixed	49.41	20.39	5.76	0.5Mb Type 5	Fixed	76.82	16.72	5.85
	Exponential	36.28	19.02	5.81		Exponential	61.13	20.60	10.81
	Static STELA	23.44	21.18	6.48		Static STELA	51.94	20.97	11.03
0.5Mb Type 3	Fixed	52.35	16.33	5.70	0.5Mb Type 6	Fixed	73.20	17.18	6.23
	Exponential	42.98	18.00	5.66		Exponential	53.72	23.32	14.28
	Static STELA	33.83	19.50	6.11		Static STELA	43.82	24.94	15.81
1.0Mb Type 1	Fixed	56.98	14.98	7.61	1.0Mb Type 4	Fixed	82.08	16.36	7.24
	Exponential	45.73	15.0	7.68		Exponential	68.25	19.19	9.21
	Static STELA	34.56	15.98	7.87		Static STELA	58.80	19.52	9.29
1.0Mb Type 2	Fixed	53.61	15.03	7.63	1.0Mb Type 5	Fixed	85.94	16.04	7.40
	Exponential	40.48	15.01	7.68		Exponential	72.49	18.17	8.49
	Static STELA	27.43	16.61	8.0		Static STELA	63.28	18.44	8.59
1.0Mb Type 3	Fixed	59.35	14.98	7.68	1.0Mb Type 6	Fixed	77.33	16.62	7.21
	Exponential	56.98	15.0	7.68		Exponential	59.09	21.24	10.64
	Static STELA	37.54	15.89	7.86		Static STELA	49.11	22.51	11.39
1.5Mb Type 1	Fixed	62.58	14.61	5.47	1.5Mb Type 4	Fixed	88.87	15.83	6.06
	Exponential	51.33	14.61	5.48		Exponential	74.09	18.52	7.34
	Static STELA	40.00	15.27	5.60		Static STELA	64.80	18.70	7.30
1.5Mb Type 2	Fixed	57.81	14.88	5.47	1.5Mb Type 5	Fixed	95.03	15.36	5.71
	Exponential	44.68	14.61	5.47		Exponential	81.39	17.09	6.34
	Static STELA	31.43	15.69	5.69		Static STELA	72.18	17.23	6.38
1.5Mb Type 3	Fixed	66.35	14.33	5.44	1.5Mb Type 6	Fixed	81.73	16.20	6.03
	Exponential	56.97	14.62	5.48		Exponential	63.35	20.66	8.38
	Static STELA	47.50	15.12	5.58		Static STELA	53.30	21.64	8.69
0.5Mb Type 7	Fixed	452.00	15.53	6.52	0.5Mb Type 8	Fixed	387.06	15.37	6.55
	Exponential	414.50	15.53	6.52		Exponential	300.81	15.72	6.61
	Static STELA	378.01	15.54	6.52		Static STELA	222.89	19.55	7.70

Table 6.5 Testing Results for Static STELA with Threshold Value Set to 4 Beacon Intervals

Traffic CBR	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)	Traffic VBR	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)
0.5Mb Type 1	Fixed	51.38	18.02	5.76	0.5Mb Type 4	Fixed	75.76	16.97	5.76
	Exponential	34.49	18.11	5.81		Exponential	54.17	22.12	13.05
	Static STELA	29.53	18.55	5.89		Static STELA	50.07	21.99	12.74
0.5Mb Type 2	Fixed	49.41	20.39	5.76	0.5Mb Type 5	Fixed	76.82	16.72	5.85
	Exponential	29.70	18.19	5.87		Exponential	56.35	21.08	11.03
	Static STELA	23.93	18.78	5.96		Static STELA	52.08	20.70	10.93
0.5Mb Type 3	Fixed	52.35	16.33	5.70	0.5Mb Type 6	Fixed	73.20	17.18	6.23
	Exponential	38.26	18.13	5.82		Exponential	53.91	22.27	12.75
	Static STELA	34.31	18.13	5.82		Static STELA	49.98	21.85	12.48
1.0Mb Type 1	Fixed	56.98	14.98	7.61	1.0Mb Type 4	Fixed	82.08	16.36	7.24
	Exponential	40.07	15.06	7.69		Exponential	62.72	19.61	9.43
	Static STELA	35.07	15.28	7.73		Static STELA	58.76	19.38	9.30
1.0Mb Type 2	Fixed	53.61	15.03	7.63	1.0Mb Type 5	Fixed	85.94	16.04	7.40
	Exponential	33.88	15.12	7.70		Exponential	67.57	18.43	8.57
	Static STELA	28.06	15.42	7.76		Static STELA	63.17	18.40	8.60
1.0Mb Type 3	Fixed	59.35	14.98	7.68	1.0Mb Type 6	Fixed	77.33	16.62	7.21
	Exponential	45.25	15.10	7.75		Exponential	53.04	23.53	11.93
	Static STELA	41.29	15.45	7.97		Static STELA	49.30	22.46	11.21
1.5Mb Type 1	Fixed	62.58	14.61	5.47	1.5Mb Type 4	Fixed	88.87	15.83	6.06
	Exponential	45.66	14.66	5.48		Exponential	69.09	18.68	7.34
	Static STELA	40.61	14.80	5.51		Static STELA	64.90	18.59	7.33
1.5Mb Type 2	Fixed	57.81	14.88	5.47	1.5Mb Type 5	Fixed	95.03	15.36	5.71
	Exponential	38.10	14.69	5.49		Exponential	76.21	17.30	6.42
	Static STELA	32.20	14.90	5.53		Static STELA	71.79	17.31	6.46
1.5Mb Type 3	Fixed	66.35	14.33	5.44	1.5Mb Type 6	Fixed	81.73	16.20	6.03
	Exponential	52.23	14.66	5.49		Exponential	56.93	22.51	9.15
	Static STELA	48.27	14.66	5.49		Static STELA	52.99	21.81	8.76
0.5Mb Type 7	Fixed	452.00	15.53	6.52	0.5Mb Type 8	Fixed	387.06	15.37	6.55
	Exponential	395.75	15.53	6.52		Exponential	250.85	16.99	7.05
	Static STELA	377.36	15.62	6.52		Static STELA	217.65	20.92	8.02

Table 6.6 Testing Results for Static STELA with Threshold Value Set to 8 Beacon Intervals

Traffic CBR	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)	Traffic VBR	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)
0.5Mb Type 1	Fixed	51.38	18.02	5.76	0.5Mb Type 4	Fixed	75.76	16.97	5.76
	Exponential	35.93	18.13	5.82		Exponential	51.48	22.35	13.16
	Static STELA	33.93	18.83	5.96		Static STELA	49.84	21.93	12.67
0.5Mb Type 2	Fixed	49.41	20.39	5.76	0.5Mb Type 5	Fixed	76.82	16.72	5.85
	Exponential	26.36	18.54	5.94		Exponential	53.66	21.24	11.28
	Static STELA	23.71	19.21	6.08		Static STELA	51.95	20.85	10.92
0.5Mb Type 3	Fixed	52.35	16.33	5.70	0.5Mb Type 6	Fixed	73.20	17.18	6.23
	Exponential	42.91	15.06	7.69		Exponential	45.04	26.21	16.73
	Static STELA	40.83	15.42	7.78		Static STELA	43.78	24.98	15.56
1.0Mb Type 1	Fixed	56.98	14.98	7.61	1.0Mb Type 4	Fixed	82.08	16.36	7.24
	Exponential	37.17	15.19	7.72		Exponential	60.64	19.60	9.33
	Static STELA	34.91	15.35	7.75		Static STELA	58.62	19.43	9.38
1.0Mb Type 2	Fixed	53.61	15.03	7.63	1.0Mb Type 5	Fixed	85.94	16.04	7.40
	Exponential	30.51	15.29	7.74		Exponential	65.26	18.41	8.54
	Static STELA	27.82	15.63	7.81		Static STELA	63.35	18.32	8.50
1.0Mb Type 3	Fixed	59.35	14.98	7.68	1.0Mb Type 6	Fixed	77.33	16.62	7.21
	Exponential	42.91	15.0	7.70		Exponential	50.71	23.67	11.90
	Static STELA	40.83	15.0	7.71		Static STELA	49.07	22.70	11.41
1.5Mb Type 1	Fixed	62.58	14.61	5.47	1.5Mb Type 4	Fixed	88.87	15.83	6.06
	Exponential	42.74	14.74	5.50		Exponential	66.69	18.74	7.34
	Static STELA	40.45	14.85	5.52		Static STELA	65.02	18.56	7.34
1.5Mb Type 2	Fixed	57.81	14.88	5.47	1.5Mb Type 5	Fixed	95.03	15.36	5.71
	Exponential	34.67	14.83	5.52		Exponential	73.97	17.30	6.42
	Static STELA	31.94	15.03	5.56		Static STELA	71.84	17.27	6.41
1.5Mb Type 3	Fixed	66.35	14.33	5.44	1.5Mb Type 6	Fixed	81.73	16.20	6.03
	Exponential	49.89	14.66	5.49		Exponential	53.94	22.96	9.44
	Static STELA	47.73	14.90	5.53		Static STELA	52.99	21.59	8.86
0.5Mb Type 7	Fixed	452.00	15.53	6.52	0.5Mb Type 8	Fixed	387.06	15.37	6.55
	Exponential	386.36	15.54	6.52		Exponential	230.18	19.23	7.65
	Static STELA	377.79	15.59	6.52		Static STELA	209.56	23.18	8.61

Table 6.7 Testing Results for Static STELA with Threshold Value Set to 16 Beacon Intervals

Traffic CBR	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)	Traffic VBR	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)
0.5Mb Type 1	Fixed	51.38	18.02	5.76	0.5Mb Type 4	Fixed	75.76	16.97	5.76
	Exponential	30.22	18.37	5.87		Exponential	50.45	22.43	13.02
	Static STELA	28.48	21.18	6.39		Static STELA	49.71	22.07	12.84
0.5Mb Type 2	Fixed	49.41	20.39	5.76	0.5Mb Type 5	Fixed	76.82	16.72	5.85
	Exponential	24.69	18.72	5.99		Exponential	52.78	20.92	11.06
	Static STELA	22.62	23.38	6.90		Static STELA	52.11	20.70	10.89
0.5Mb Type 3	Fixed	52.35	16.33	5.70	0.5Mb Type 6	Fixed	73.20	17.18	6.23
	Exponential	34.58	18.64	5.93		Exponential	43.84	26.72	16.85
	Static STELA	33.01	20.87	6.9		Static STELA	43.69	24.59	15.72
1.0Mb Type 1	Fixed	56.98	14.98	7.61	1.0Mb Type 4	Fixed	82.08	16.36	7.24
	Exponential	35.78	15.9	7.72		Exponential	59.49	19.57	9.29
	Static STELA	33.80	16.61	7.98		Static STELA	58.67	19.57	9.31
1.0Mb Type 2	Fixed	53.61	15.03	7.63	1.0Mb Type 5	Fixed	85.94	16.04	7.40
	Exponential	28.84	15.38	7.76		Exponential	63.82	18.45	8.63
	Static STELA	26.48	17.73	8.22		Static STELA	63.20	18.33	8.51
1.0Mb Type 3	Fixed	59.35	14.98	7.68	1.0Mb Type 6	Fixed	77.33	16.62	7.21
	Exponential	41.50	15.32	7.75		Exponential	49.48	23.62	11.91
	Static STELA	39.70	16.43	7.97		Static STELA	49.48	22.16	11.21
1.5Mb Type 1	Fixed	62.58	14.61	5.47	1.5Mb Type 4	Fixed	88.87	15.83	6.06
	Exponential	41.34	14.74	5.50		Exponential	65.90	18.71	7.34
	Static STELA	39.12	15.68	5.68		Static STELA	64.53	18.81	7.41
1.5Mb Type 2	Fixed	57.81	14.88	5.47	1.5Mb Type 5	Fixed	95.03	15.36	5.71
	Exponential	32.98	14.87	5.53		Exponential	72.58	17.39	6.44
	Static STELA	30.35	16.43	5.84		Static STELA	71.75	17.31	6.45
1.5Mb Type 3	Fixed	66.35	14.33	5.44	1.5Mb Type 6	Fixed	81.73	16.20	6.03
	Exponential	48.43	14.83	5.52		Exponential	53.42	22.88	9.21
	Static STELA	46.40	15.57	5.67		Static STELA	52.97	21.73	8.79
0.5Mb Type 7	Fixed	452.00	15.53	6.52	0.5Mb Type 8	Fixed	387.06	15.37	6.55
	Exponential	381.63	15.54	6.52		Exponential	209.28	23.02	8.52
	Static STELA	377.39	15.59	6.52		Static STELA	197.88	28.07	9.72

6.2 Dynamic STELA

The test scenarios used for dynamic STELA assessment vary three variables: traffic pattern, data rate and threshold value. The first two parameters apply to all the compared schemes while the threshold value is only used by IEEE 802.16 to determine the maximum value of the WNIC's sleeping window, as discussed in the previous chapter. Both energy consumption and quality of service metrics such as average end-to-end delay, average jitter and PSNR are analyzed in order to estimate the delivery performance achieved by dynamic STELA. Testing results for 0.5Mbps, 1.0Mbps and 1.5Mbps traffic rates are shown in Table 6.8 , Table 6.9 and Table 6.10, respectively.

Table 6.8 Testing Results for Dynamic STELA with Data Rate set to 0.5Mbps

Traffic Pattern	**Scheme**	**Energy (Joule)**	**Delay (ms)**	**Jitter (ms)**	**PSNR (dB)**
1s on/2s off	Fixed	18.75	24.64	3.77	43.31
	Exponential-2	17.07	32.73	4.93	36.66
	Exponential-4	16.51	33.34	3.65	36.03
	Exponential-8	16.32	36.56	3.45	32.94
	Exponential-16	12.62	38.9	9.75	28.42
	Dynamic STELA	12.62	30.91	9.75	40.20
2s on/1s off	Fixed	34.59	32.5	3.88	33.06
	Exponential-2	34.12	37.8	3.88	29.83
	Exponential-4	33.9	38.2	3.88	29.58
	Exponential-8	31.45	42.46	3.36	26.12
	Exponential-16	31.45	44.65	3.36	26.60
	Dynamic STELA	33.99	39.5	3.88	30.53
1s on/1s off	Fixed	26.77	31.69	3.77	35.77
	Exponential-2	26.07	32.9	3.77	34.53
	Exponential-4	25.74	34.09	3.78	32.98
	Exponential-8	22.06	36.57	2.74	30.71
	Exponential-16	22.06	40.65	2.74	20.66
	Dynamic STELA	25.89	31.74	3.77	34.60

Table 6.9 Testing Results for Dynamic STELA with Data Rate set to 1.0Mbps

Traffic Pattern	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)	PSNR (dB)
1s on/2s off	Fixed	34.38	25.48	3.81	32.12
	Exponential-2	33.19	30.72	4.04	27.50
	Exponential-4	31.12	32.88	4.44	26.82
	Exponential-8	28.72	35.19	4.9	23.67
	Exponential-16	22.44	38.39	5.11	19.20
	Dynamic STELA	25.78	26.9	3.52	29.08
2s on/1s off	Fixed	66.04	34.31	3.91	32.76
	Exponential-2	65.11	41.07	5.57	27.44
	Exponential-4	64.58	47.57	6.2	22.07
	Exponential-8	57.99	49.07	5.26	21.32
	Exponential-16	50.99	51.18	5.26	19.19
	Dynamic STELA	55.44	35.52	3.93	31.99
1s on/1s off	Fixed	50.41	25.5	3.81	33.93
	Exponential-2	49.29	27.68	3.88	30.24
	Exponential-4	48.6	28.62	5.76	29.15
	Exponential-8	40.72	37.37	3.66	21.67
	Exponential-16	38.72	46.09	3.66	15.48
	Dynamic STELA	39.52	29.76	3.81	30.03

Table 6.10 Testing Results for Dynamic STELA with Data Rate set to 1.5Mbps

Traffic Pattern	Scheme	Energy (Joule)	Delay (ms)	Jitter (ms)	PSNR (dB)
1s on/2s off	Fixed	36.63	30.79	4.04	41.29
	Exponential-2	35.31	40.63	4.01	37.64
	Exponential-4	34.04	48.5	3.65	30.58
	Exponential-8	32.38	53.63	3.11	28.91
	Exponential-16	28.74	62.91	3.07	21.20
	Dynamic STELA	30.9	33.79	2.28	38.65
2s on/1s off	Fixed	68.03	50.82	4.35	31.36
	Exponential-2	66.99	50.85	4.4	31.41
	Exponential-4	66.53	64.48	4.42	21.37
	Exponential-8	63.10	65.3	3.47	21.46
	Exponential-16	63.10	65.58	3.47	21.58
	Dynamic STELA	65.58	60.80	4.17	30.64
1s on/1s off	Fixed	53.81	42.87	4.37	37.51
	Exponential-2	52.23	45.2	4.37	36.55
	Exponential-4	51.55	50.12	4.34	33.84
	Exponential-8	46.38	51.17	2.45	33.72
	Exponential-16	46.38	58.61	2.46	24.18
	Dynamic STELA	47.18	46.33	3.84	36.38

Simulation results demonstrate the balanced performance between energy conservation and quality of service levels, achieved by dynamic STELA. Significant savings on energy is achieved without quality of service degrading severely. On the other hand, it can be noted that IEEE 802.11 is a high energy consuming solution and IEEE 802.16 provides a mechanism biased towards energy efficiency.

Dynamic STELA saves similar amounts of energy with IEEE 802.16 with maximum sleeping window set to 16, but significantly reduces the delay and improves PSNR levels. When the sleeping window is set to 2, IEEE 802.16 generates similar or small amounts of delay, but with much higher energy consumption. IEEE 802.11 has the shortest delay due to frequent waking up, but at the same time wastes the most energy. Average jitter does not vary significantly in the three schemes studied.

Next the influence of traffic patterns, data rates and threshold values on dynamic STELA performance are analyzed, respectively. The case of multiple servers is also discussed in details.

6.2.1 Impact of Traffic Pattern on the Performance of Dynamic STELA

Three sets of traffic patterns are individually tested with different on/off intervals including 1s on/ 1s off, 2s on/ 1s off, 1s on/ 2s off. Testing results show that longer on period and shorter off intervals lead to higher energy consumption for all the compared power saving schemes. This is due to the high energy consumption of the transmitting and receiving modes of the WNIC. Figure 6.3 and Figure 6.4 demonstrate an example of the impact of traffic pattern on the performance of the compared schemes when the traffic rate is set to 1.0Mbps and threshold value is set to 8 beacon intervals. The threshold value 8 is used in the testing of 802.16 as this section included only to show the impact of the traffic pattern on the performance of the three compared schemes. The impact of the threshold value is illustrated in section 6.2.3. For example, IEEE 802.11 consumes 66.04 Joule of energy with the longest on interval traffic pattern, i.e. 2s on/1s off, and 34.38 Joule of energy with the shortest on interval traffic pattern, i.e. 1s on/2s off, while the energy consumption with the 1s on/1s off traffic pattern lies in between, i.e. 50.41 Joule. The same phenomenon is observed for the other two schemes as well.

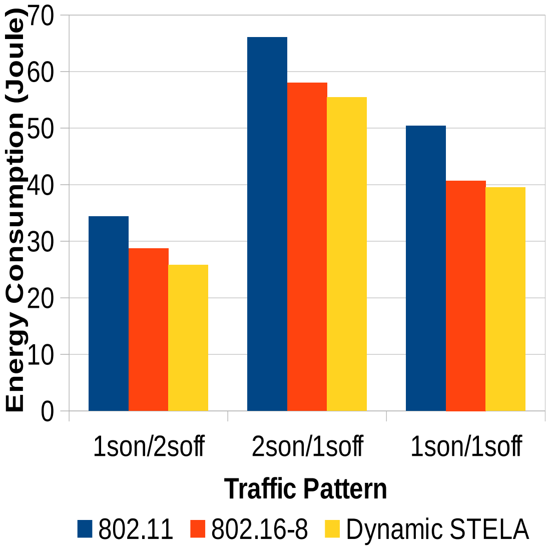


Figure 6.3 Impact of traffic pattern on energy consumption: Data Rate:1.0Mbps, Threshold Value: 8

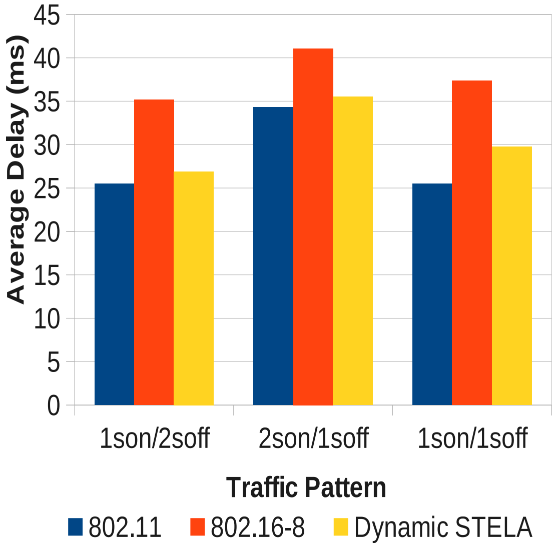


Figure 6.4 Impact of traffic pattern on packet delay: Data Rate:1.0Mbps, Threshold Value: 8

The traffic pattern also affects the performance of dynamic STELA compared with other solutions. STELA saves more energy when the on period is short. Short on periods indicate less packets need to be received by the mobile host and more time can be spent in the sleeping mode, enabled by the dynamic WNIC scheduling. For instance, with 1.0 Mbps of data traffic, the energy saving achieved by dynamic STELA is 25.01%, 16.05%, 21.60% higher, with traffic pattern 1s on/2s off, 2s on/1s off and 1s on/1s off, respectively, compared with IEEE 802.11.

The impact of traffic pattern on average packet delay and PSNR is dependent on two variables: queuing time and sleeping schedule of the WNIC. On the one hand, the longer the on periods, the more packets are transmitted by the server and thus longer queuing is expected at the service gateway and longer delays and lower PSNR levels are experienced. On the other hand, the longer the on intervals, the slower the sleeping window grows, which might compensate for the increased delays caused by queuing.

6.2.2 Impact of Data Rate on the Performance of Dynamic STELA

It can be seen that with the increase of the data rate, energy consumption increases for all the compared schemes and IEEE 802.16 exhibits worse delays when the value of sleeping window is large. Figure 6.5 gives an example of the impact of the data rate with the 1s on/2s off traffic and W_{thre} set to 8.

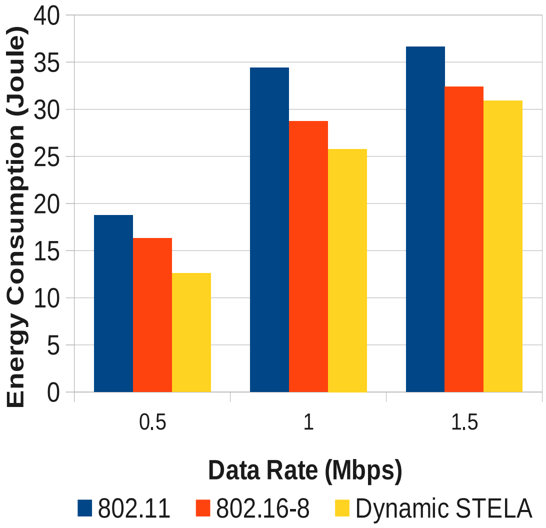


Figure 6.5 Impact of data rate on energy consumption: Traffic Pattern: 1s on/2s off, Threshold Value: 8

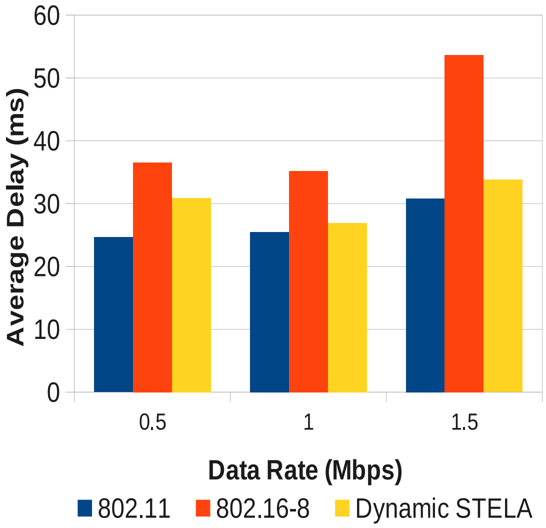


Figure 6.6 Impact of data rate on packet delay: Traffic Pattern: 1s on/2s off, Threshold Value: 8

Higher data rates indicate that more packets are generated by the server and therefore more time is spent in data receiving, which is the most energy consuming mode of the WNIC. Therefore the energy consumption associated with the wireless interface grows no matter which power saving solution is considered. Moreover, the energy saving achieved by dynamic STELA decreases with the increase in the data rate, due to shorter sleeping period caused by more packets received by the client host. For example, using IEEE 802.11 as a benchmark, under traffic pattern of 1s on/2s off, the energy saving achieved by dynamic STELA decreased from 32.69% to 15.64% with the data rate increasing from 0.5 Mbps to 1.5 Mbps.

The impact of data rate on average packet delay, as shown in Figure 6.6 varies according to two main factors. First, the higher data rate, the longer time is spent in queuing at the gateway and therefore delays are increased. On the other hand, the WNIC is kept busy for longer and less time is spent in sleeping mode which might compensate for the increased delays and decreased PSNR scores to some extent.

6.2.3 Impact of Threshold on the Performance of Dynamic STELA

This parameter only applies to IEEE 802.16 and indicates the maximum value of sleeping window size. It is configured beforehand as there is no adaptive scheme in the I802.16 binary exponential increase algorithm. Four values, increasing from 2 to 16 beacon intervals are configured to demonstrates the impact of W_{thre} on the tested solutions, respectively: 2, 4, 8, 16. The impact of the threshold on the energy consumption and packet delay is shown in Figure 6.7 and Figure 6.8. The traffic pattern is set to 1s on/2s off with 1.0 Mbps data rate.

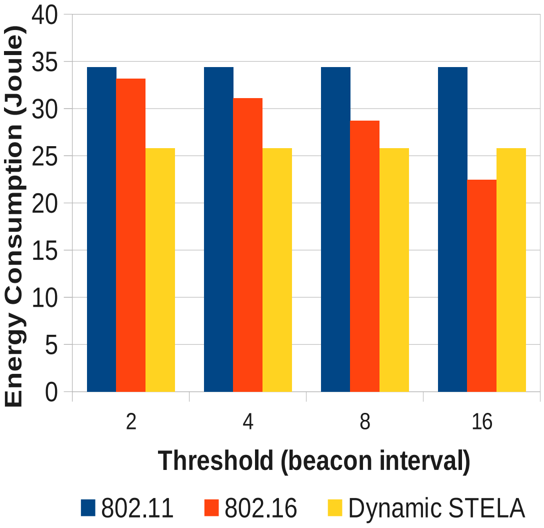


Figure 6.7 Impact of threshold value on energy consumption: Traffic Pattern: 1s on/2s off, Data Rate:1.0 Mbps

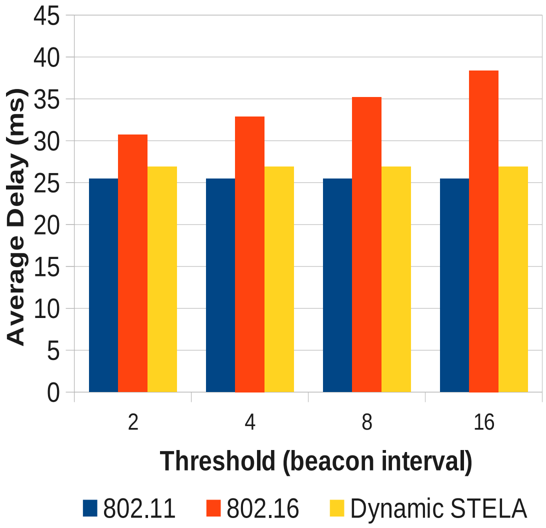


Figure 6.8 Impact of threshold value on packet delay: Traffic Pattern: 1s on/2s off, Data Rate:1.0 Mbps

It can be seen that the performance of IEEE 802.11 and dynamic STELA is not dependent on the W_{thre}, while IEEE 802.16 is significantly affected by W_{thre}. Figure 6.7 shows that small W_{thre} has very limited ability to conserve energy although it does not increase the delays and jitter too much, while large W_{thre} saves energy at a the expense of quality of service alteration. Energy saving is achieved by switching the WNIC to low power consumption mode without considering the traffic arrival patterns, and therefore QoS is affected.

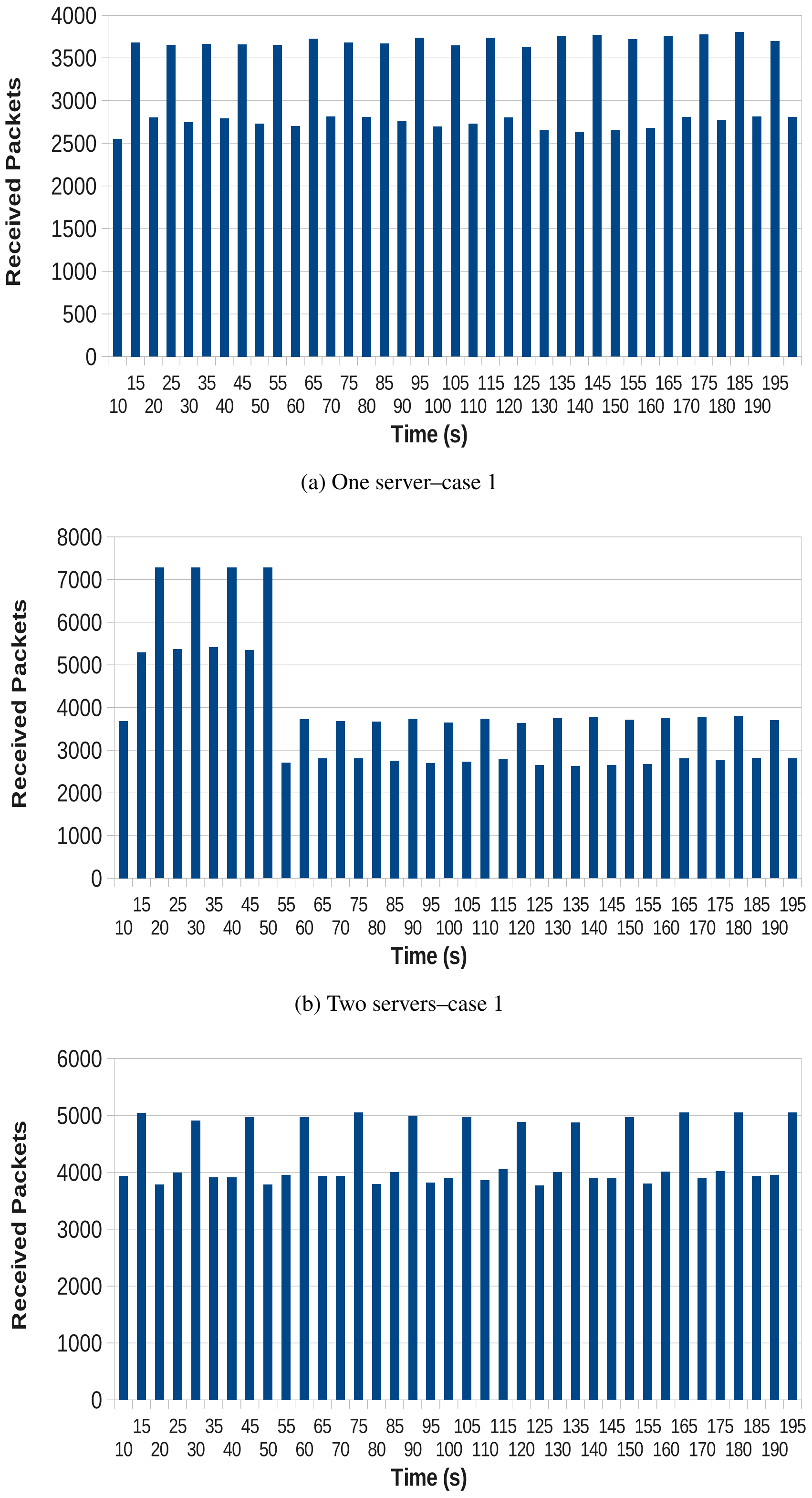
On the other hand, compared with IEEE 802.16 with the W_{thre} set to 16, dynamic STELA saves almost the same amount of energy without increasing packet delay or decreasing PSNR scores significantly, using IEEE 802.11 as benchmark. This is mainly due to aggressive growth of the sleeping window which is offset by the linear increase phase and adaptive configuration of the threshold value used by dynamic STELA. For example, with a data rate set to 1.0 Mbps and a traffic pattern of 1s on/2s off, IEEE 802.16 consumes 28.72 Joule of energy when the threshold value is set to 8 while the energy consumed by dynamic STELA is 25.78 Joule. Therefore dynamic STELA is 10% more energy efficient than 802.16. Although IEEE 802.16 saves 9% more energy when the threshold value is increased to 16 beacon intervals, it is obvious that dynamic STELA is capable of maintaining similar average delay and PSNR levels as IEEE 802.11, unlike IEEE 802.16 which introduces 50% larger delays with PSNR drops under 15dB. This demonstrates that dynamic STELA is an energy efficient solution with good QoS levels.

6.2.4 Impact of Multiple Servers on the Performance of Dynamic STELA

Both initiation and termination of a session results in significant fluctuation in the data arrival pattern and hence the sleep/wakeup schedule of the WNIC. Figure 6.8(a) demonstrates the fluctuation in data arrival pattern with data rate of 0.5 Mpbs and various data patterns generated by the servers. In Figure 6.8, there is a single server which generates CBR traffic for an interval of 200 seconds. It can be seen that the traffic pattern remains relatively static during the whole process, with a regular on/off pattern. For example, Figure 6.8(b) demonstrates the changes in data arrival pattern when an additional server joins and communicates with the client for a period of 40s during the 200s long simulation in case 1. At time=10s when server 2 starts data transmission, data arrival pattern changes significantly and forms a new pattern, i.e. increased number of packets received per second, which remains similar for the next 40 seconds until server 2 stops transmitting data, resulting in another pattern, i.e. decreased number of packets received per second, in the figure.

The threshold adjusting phase is triggered when there are any application session changes made at the application layer in order to capture any large fluctuation in the data arrival pattern. To be more specific, the process is initiated at 0 second, 10 seconds and 50 seconds after the first server joins the communication, respectively, which is the time when the second sever joins. The process is also triggered when the second server quits the communication, .

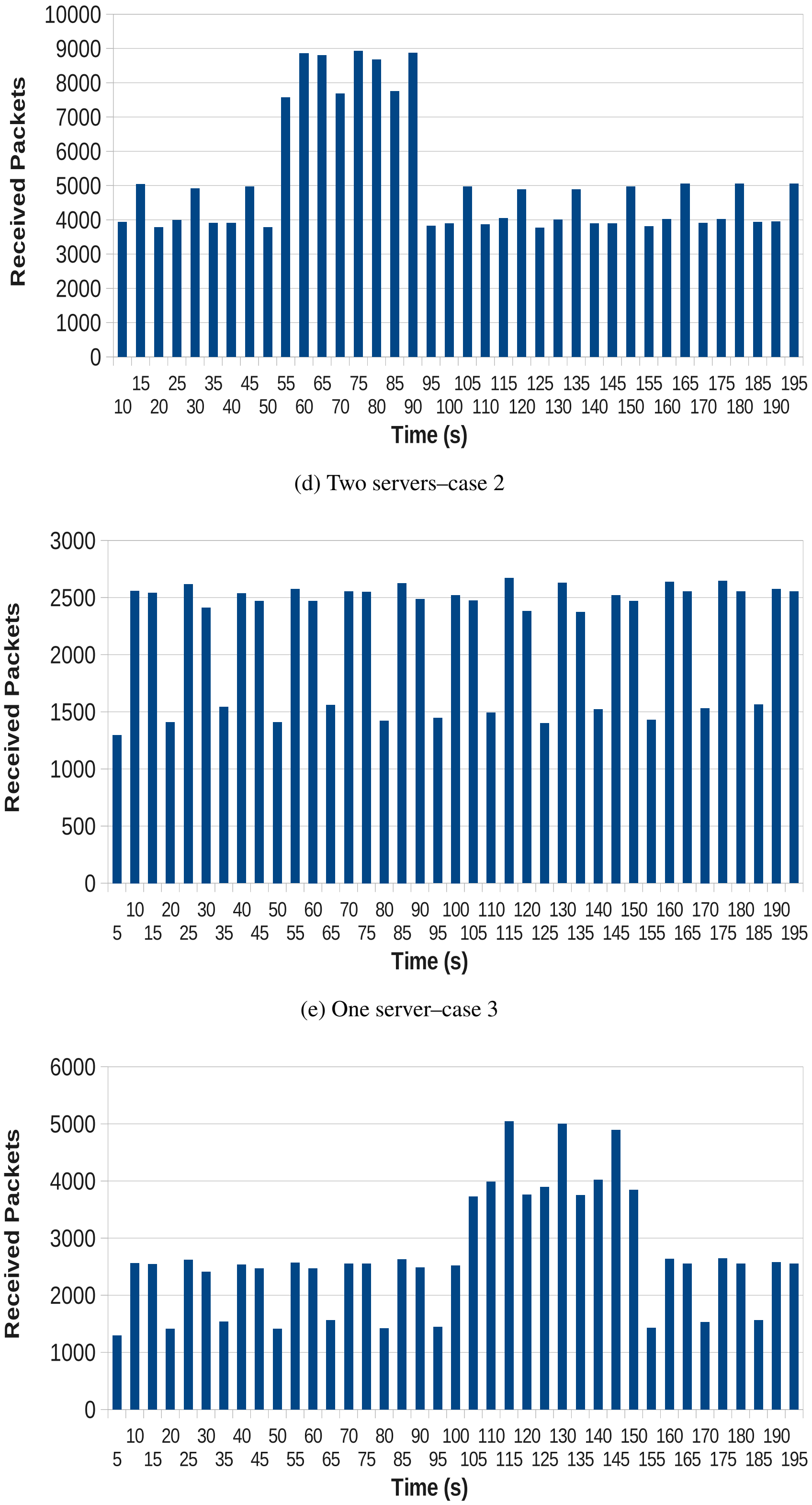
Testing results of the second scenario discussed in the previous chapter which includes two servers, are listed in Table 6.11. It can be seen that due to the multiple callings of the threshold adjusting phase, dynamic STELA is still capable of providing high quality of



(a) One server–case 1

(b) Two servers–case 1

(c) One server–case 2



(d) Two servers–case 2

(e) One server–case 3

(f) Two servers–case 3

Figure 6.8 Impact of session changes on data arrival pattern

service with good results in terms of energy efficiency. For example, with a traffic pattern of 1s on/2s off, dynamic STELA consumes 26.60 Joule of energy, similar to 25.35 Joule consumed by IEEE 802.16's exponential-16, which results in a saving of approximately 15.53% compared to IEEE 802.11. However, dynamic STELA only introduces 1s extra delay and similar PSNR score while an extra 16 seconds delay and a drop of 15dB PSNR score is introduced by IEEE 802.16.

It can be seen that dynamic STELA does not decrease the delivery quality levels in order to achieve energy saving, unlike most energy saving solutions such as IEEE 802.16. For example, in the case of the traffic pattern set to 2s on/1s off with two servers deployed, which indicates high traffic intensity, 802.16 with threshold 16 saves a total of 2 Joule of energy but introduces 14 ms of extra delay, and the PSNR score drops from 37dB to 26dB. In the context of high data intensity, dynamic STELA although does not conserve too much power, makes sure that the quality is not compromised, as well.

Table 6.11 Testing Results for Dynamic STELA with Two Servers, Data Rate set to 0.5Mbps

Traffic Pattern	**Scheme**	**Energy (Joule)**	**Delay (ms)**	**Jitter (ms)**	**PSNR (dB)**
1s on/2s off	Fixed	31.49	21.83	2.38	35.95
	Exponential-2	30.71	25.83	2.68	23.65
	Exponential-16	25.35	37.83	3.88	20.06
	Dynamic STELA	26.60	22.83	3.58	35.29
2s on/1s off	Fixed	40.74	42.89	4.27	27.18
	Exponential-2	40.36	43.70	4.27	27.02
	Exponential-16	38.23	57.87	3.86	16.34
	Dynamic STELA	40.27	42.94	4.27	27.02
1s on/1s off	Fixed	36.83	42.68	4.54	26.19
	Exponential-2	34.83	47.06	4.61	20.20
	Exponential-16	27.83	68.71	3.18	15.34
	Dynamic STELA	28.83	43.94	4.54	25.49

6.3 Q-PASTE

Besides adaptive scheduling of the WNIC at MAC layer, deliberate data shaping is employed at application layer by the service gateway when Q-PASTE is enabled. Therefore Q-PASTE is not only compared with MAC layer energy efficient solution IEEE 802.11, but is also tested against a cross-layer scheme Buffered Streaming [204]. Buffered Streaming is an energy efficient solution designed for multimedia content delivery. At the service gateway, data is transmitted at a higher rate at the beginning and buffered for a certain period of time before is released for the rest of the playback time. IEEE 802.11 is adopted at the client side in order to improve energy efficiency.

Multimedia streams with various encoding rate evaluated respectively. The value of the maximum buffering period is calculated according to the encoding rate and playout buffer size at the client side. This information is then used as a boundary of the parameter buffering period which has direct impact on the performance of Q-PASTE.

On the other hand, the user behavior of playback start time is an arbitrary value which affects the playout buffer size significantly. Therefore two extreme scenarios are considered: playback starting immediately when the data transmission is triggered and playback starting after the fast streaming phase is complete.

Energy consumption and QoS are individually measured as usual. Besides that, quality of experience is also tested in the evaluation of Q-PASTE, as higher quality is required by the users in multimedia delivery applications. Testing results have demonstrated that compared with existing solutions, Q-PASTE provides high energy efficiency and maintains high quality levels for multimedia delivery, which is attributed to the cooperation of application layer data shaping scheme and MAC layer dynamic WNIC scheduling solution.

Next Q-PASTE is assessed in terms of the effect different parameters have on energy consumption and delivery quality.

6.3.1 Impact of Buffering Period on Energy Consumption

When the fast start streaming timer expires, data from the server is buffered at the gateway for an interval of t_{ds} (i.e. the burst scheduling time) before is released to the client in the form of bursts. t_{ds} should be smaller than $T_{bf_{min}}$ which is the low bound of the maximum time that data can be buffered without compromising user quality of experience levels. For stream 1, the playout buffer size was set to 410 kb, value calculated as the fast streaming interval multiplied by the data rate without the traffic shaping solution activated. $T_{bf_{min}}$, illustrated in eq. (4.33), is then calculated as the buffer size divided by the encoding rate of 16 kbps, which gives 25 s. $T_{bf_{min}}$ for the other two streams are calculated as 50 s and 16 s, respectively.

The average energy consumption when the proposed solution Q-PASTE and the other two schemes are employed in turn are shown in Table 6.12 when delivering the three streams. For each stream, an average of the energy consumed with buffering period varying from 1 s to $T_{bf_{min}}$ is measured. Additionally, Table 6.12 shows the comparison results of the energy consumption of Buffered Streaming and Q-PASTE against that of IEEE 802.11 PSM expressed in percentages. With the data shaping algorithm, both Q-PASTE and Fast Streaming achieve significant higher energy savings compared with the standard PSM proposed in IEEE 802.11. Even a short buffering period has a huge saving on the energy compared with the standard PSM. For instance, for stream 1, Buffered Streaming consumes an average of 7.57 Joule of energy with varying buffering periods, which is about 20% of the total energy consumption when using PSM. Significantly Q-PASTE consumes less than 10% of the energy consumed when employing PSM (3.41 Joule). Figure 6.9, Figure 6.10 and Figure 6.11 illustrate the energy consumptions of the three schemes with different buffering periods when delivering the three streams, respectively. It can be clearly seen that the longer the data is buffered, the less frequently user WNIC wakes up and the less power is consumed for both Q-PASTE and Fast Streaming. For example, when Q-PASTE is employed, the energy consumed with 5 s buffering period is 9.11 Joule, while the energy consumption decreases to 2.62 Joule with a buffering period of 10 s, and the energy consumption further decreases to 1.62 Joule for a 20 s buffering period. The same rule applies to the Fast Streaming algorithm. Buffering period has no effect on the performance of IEEE 802.11 PSM, as it does not include any traffic shaping.

Table 6.12, Figure 6.9, Figure 6.10 and Figure 6.11 clearly demonstrate that Q-PASTE is much more energy efficient than Fast Streaming when supporting similar delivery quality, as demonstrated in the next section and also suggested in Section 6.3.5. This is an important

Table 6.12 Average Energy Consumption when Delivering Stream 1, 2 and 3, respectively

Stream No.	**Max. Buffering Period(s)**	**Average Energy Consumption**					
		802.11		**Buffered Streaming**		**Q-PASTE**	
		Joule	%	Joule	%	Joule	%
1	25	36.46	100	7.57	20.76	3.41	9.35
2	50	22.67	100	7.47	32.95	2.26	9.97
3	16	36.46	100	9.71	26.63	4.55	12.48

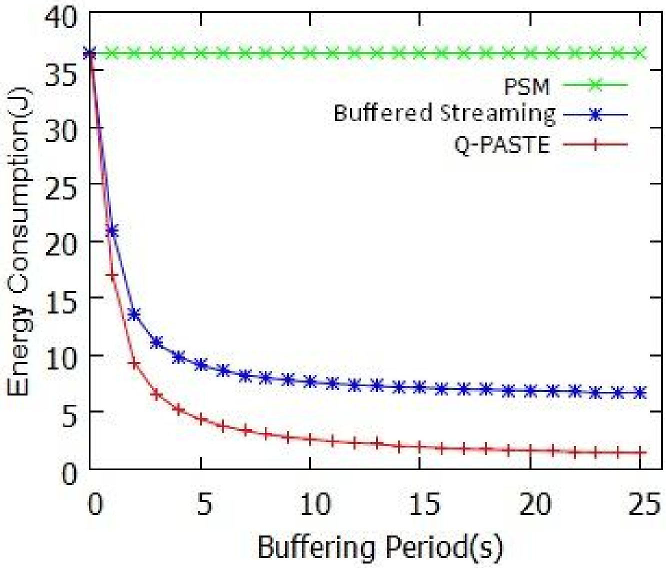


Figure 6.9 Energy consumption when delivering Stream 1 with different buffering periods

benefit of the proposed adaptive sleep/wakeup scheduling of WNIC, employed by STELA. The energy saving increases as the buffering period grows. This is due to the fact that STELA allows longer sleeping periods, i.e. sleeping window size grows fast during the exponential increase phase if no packet arrives, while it is able to wake up the WNIC before the expected data arrival time. On the other hand, Fast Streaming adopts the standard PSM as its sleep/wakeup scheduler, which leads to unnecessary transceiver wake ups and wastes energy.

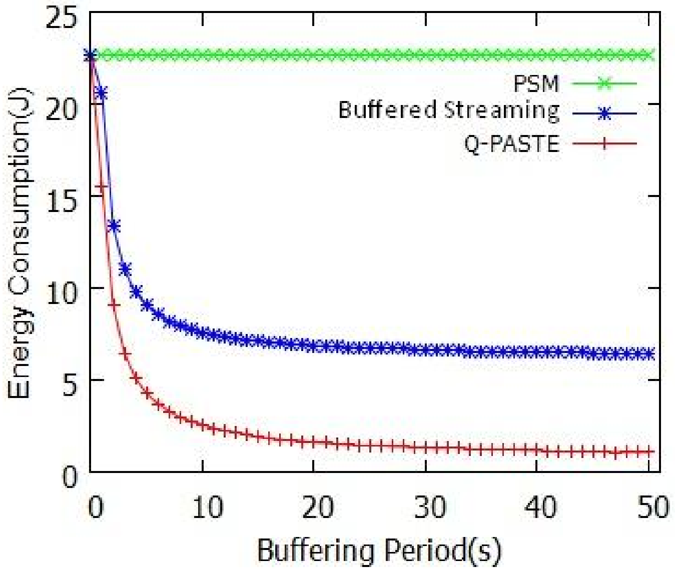


Figure 6.10 Energy consumption when delivering Stream 2 with different buffering periods

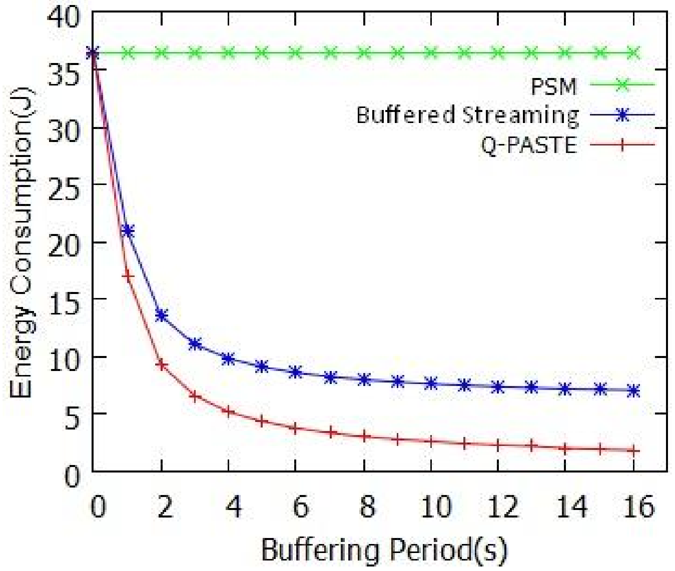


Figure 6.11 Energy consumption when delivering Stream 3 with different buffering periods

6.3.2 Impact of Buffering Period on Playout Buffer Size

This section studies the impact of buffering period on playout buffer size when the playback starts immediately after the connection between the client and the server is established. Buffering period is directly related to the size of the client-side playout buffer as it controls the duration of data being buffered at the service gateway. The larger the buffer

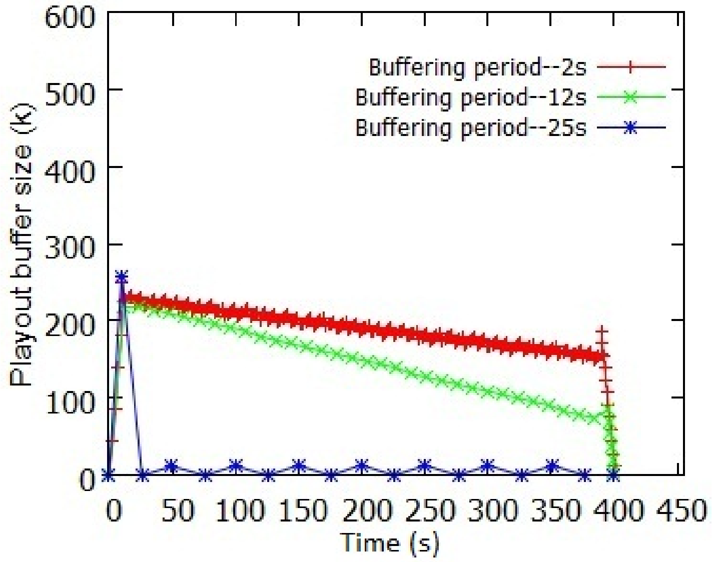


Figure 6.12 Playout buffer size with different buffering periods when delivering Stream 1 (Playback: immediate, Scheme: Q-PASTE)

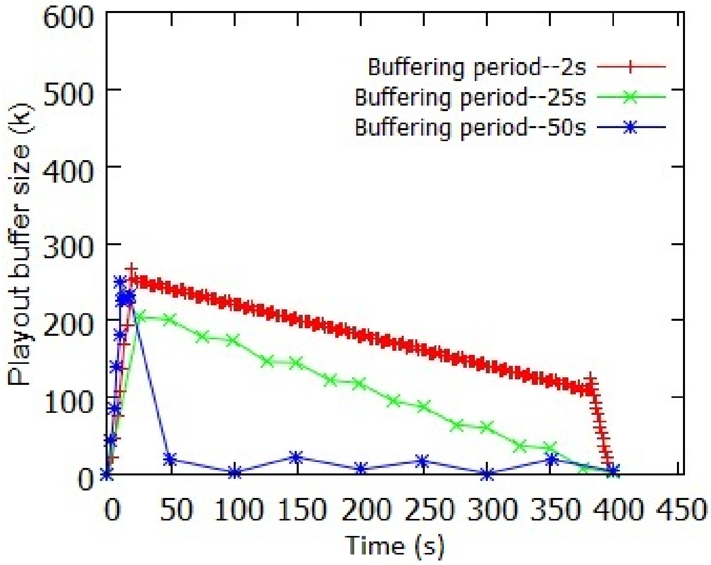


Figure 6.13 Playout buffer size with different buffering periods when delivering Stream 2 (Playback: immediate, Scheme: Q-PASTE)

size is, the longer the playout buffer is drained without any data input and the more likely s_{po} (i.e. the size of data stored in the playout buffer) drops severely before the next data bursts arrive. Figure 6.12, Figure 6.13 and Figure 6.14 show the impact of the buffering period on s_{po} for the three streams, respectively.

From Figure 6.12, it can be seen that at the beginning, s_{po} suffers from severe fluctu-

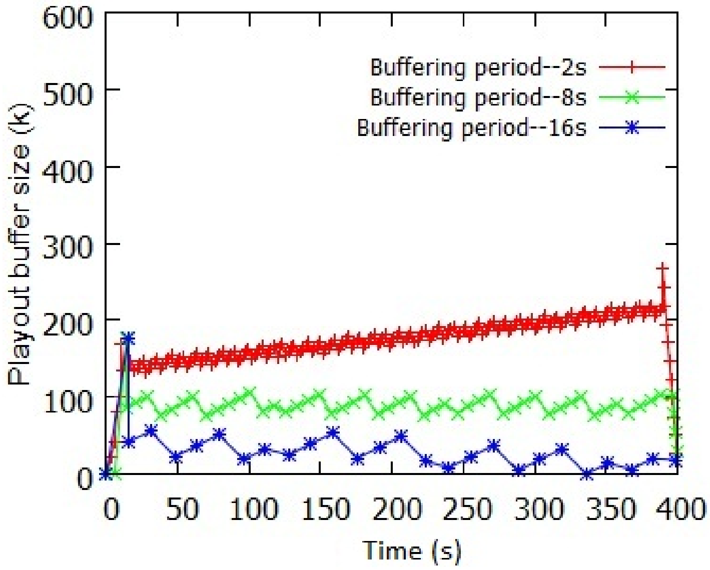


Figure 6.14 Playout buffer size with different buffering periods when delivering Stream 3 (Playback: immediate, Scheme: Q-PASTE)

ations when buffering period is set to 25 s, which is the maximum value that can be set without compromising user quality of experience levels. During fast streaming period of 10 s, data is accumulated without being reshaped at the gateway and s_{po} grows fast. After the fast start period, data from the server is stored in the gateway for 25 s, during which period the playout buffer is drained by the player. s_{po} sees linear drop during this period. Once the buffering timer expires, a burst of data of size $t_{ds} * r_{ec}$ is released before the buffer is almost drained empty, and the playout buffer is filled again, where t_{ds} and r_{ec} are the buffer releasing interval and the encoding rate, respectively. However, s_{po} does not drop to the lowest value again as we guarantee the reshaped data is released earlier than the buffering period which means more data is accumulated than consumed in the playout buffer. The process repeats until all data is received by the client host. With shorter buffering periods, the s_{po} is kept higher at all times as new packet bursts arrive at the playout buffer sooner and less fluctuations in s_{po} are observed. Between 350s and 400s depending on the buffering period, all the data has been delivered to the client and the buffer window decreases in a linear manner until the end of the playback.

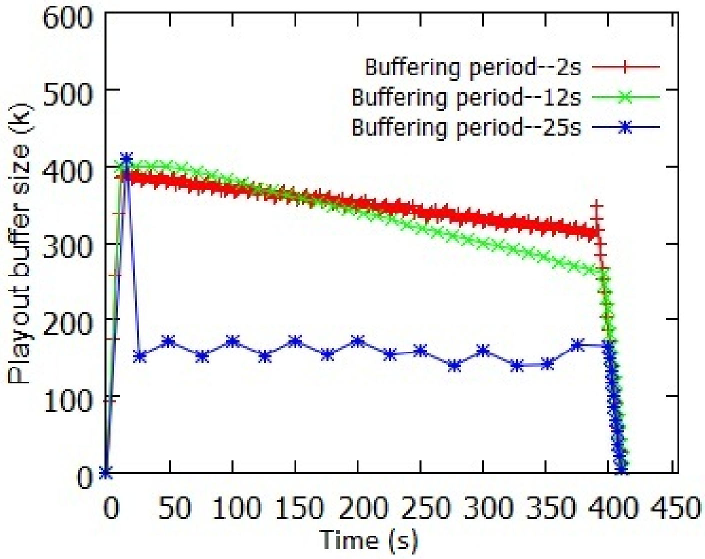


Figure 6.15 Playout buffer size with different buffering periods when delivering Stream 1 (Playback: after fast start, Scheme: Q-PASTE)

6.3.3 Impact of Playback Time on Playout Buffer

In Section 6.3.2, we have conducted the experiments as the user started streaming once the first packet arrives. This is the situation where playout buffer drains quickly as data is pulled during the fast start period. In this section, we study the impact of the playback start time on the size of the playout buffer by letting the data draining process start after the fast start period, as shown from Figure 6.15 to Figure 6.17.

For each stream, late startup time achieves smoother fluctuations in the playout buffer size. For example, Figure 6.12 shows the situation of the user starting playback at the beginning of streaming connection being established. It can be seen that with a buffering period of 25 s, the buffer size reaches its peak of 250 k after the fast start period and nearly drops to 0 after every 25 s hereafter. However, if the user starts playback after the fast start period, as shown in Figure 6.15 with the same buffering period, the playout buffer size reaches its peak of 410 k after the fast start period and then fluctuates around 150 k.

Although the playout buffer size drops seriously with fast startup time at the client side, PAT working along with STELA provides fast delivery of data which guarantees the playout buffer is never drained empty before the next data bursts arrive. Therefore smooth

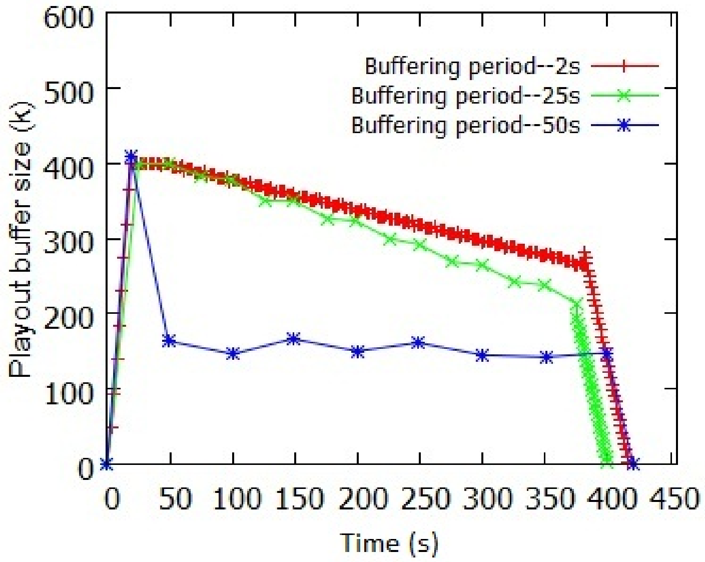


Figure 6.16 Playout buffer size with different buffering periods when delivering Stream 2 (Playback: after fast start, Scheme: Q-PASTE)

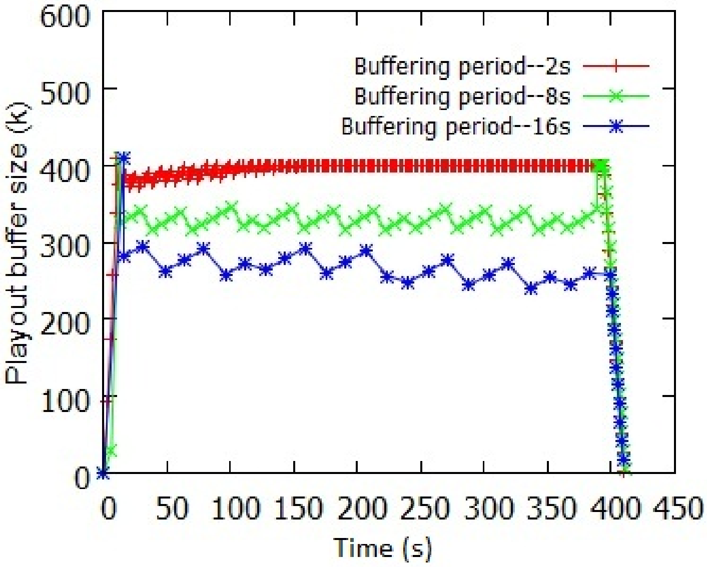


Figure 6.17 Playout buffer size with different buffering periods when delivering Stream 3 (Playback: after fast start, Scheme: Q-PASTE)

playback is guaranteed.

6.3.4 Impact of Buffering Period on End-to-End Delay

Deliberate buffering at application layer will introduce extra end-to-end packet delay in Q-PASTE, which is similar with the approach employed by all packet buffering schemes such as Buffered Streaming [204]. The impact of the buffering period on average packet delay is shown in Table 6.13. It can be seen that the buffering period does not affect the IEEE 802.11 MAC layer, while highly impacts on both cross-layer solutions: Q-PASTE and Buffered Streaming. With a buffering period t_{bf} , the longest time interval that any packet can be delayed at the application layer, the average packet delay is smaller than t_{bf}, as validated in Table 6.13. It is evident that the larger t_{bf} is, the higher the energy saving is, but also the larger the packet delay introduced. Although both Q-PASTE and Buffered Streaming introduce long delays, if the buffering period is set long, they do not impact user quality of experience levels, as the playback buffer window is always filled with new packets before being drained empty [87], as shown in Section 6.3.2.

Besides upper layer packet buffering, Q-PASTE employs STELA at the MAC layer, which takes advantage of the long WNIC sleeping periods generated by PAT in order to achieve energy efficiency. Although long sleeping duration might cause delayed response to arriving packets, it can be seen from Table 6.13 that Q-PASTE does not introduce extra delay in comparison with Buffered Streaming, while providing much higher energy efficiency, as demonstrated in Section 6.3.1. The main reason behind this high energy efficiency and low delay cost is that STELA at MAC layer adapts very well the WNIC sleeping behavior to the traffic pattern during the self-adjusting phase. STELA studies the data arrival pattern and the sleep/wakeup schedule is adjusted correspondingly every time when an application session is established or terminated in order to allow effective wakeup of WNIC for the predicted future traffic. Therefore the sleeping interval is increased and the WNIC can still wake up in time for data receiving.

Table 6.13 End-to-End delay for 802.11, Buffered Streaming and Q-PASTE

Stream No.	**Buffering Periods:** t_{bf} **(s)**	**Average Packet Delay(s)**		
		802.11	**Buffered Streaming**	**Q-PASTE**
1	2	0.59	1.67	1.66
	12	0.59	6.48	5.5
	25	0.59	18.59	18.22
2	2	0.58	1.60	1.58
	25	0.58	21.54	20.20
	50	0.58	37.66	35.73
3	2	0.62	1.80	1.79
	8	0.62	7.80	7.79
	16	0.62	12.84	12.28

6.3.5 Quality Evaluation

In this section, we study the effect of Q-PASTE and that of the other schemes on stream quality levels. ITU-T E-Model [65] provides a tool for voice quality evaluation in heterogeneous networks. R-factor, as a result of the E-Model, takes into account both the physical equipment impairments, including network delivery, and the related perceptual effects. R-factor is simplified by [235] and calculated as eq. (6.3).

$$R = 94.2 - I_d - I_{ef} \tag{6.3}$$

I_d refers to the impairment caused by mouth-to-ear delay including network delay, playout delay and codec delay, and I_{ef} represents losses due to codecs and network. The values of I_d and I_{ef} are hard to obtain and therefore are simplified for G.711 codec by [235], as in eq. (6.4).

$$\begin{aligned} R &= 94.2 - 0.24 * d \\ &- 0.11 * (d - 177.3)H(d - 177.3) \\ &- 30log(1 + 15e) \end{aligned} \tag{6.4}$$

where $H(x)$ is 1 if $x \geq 0$ or is 0 otherwise and d refers to the one way delay from the source to the sink, as in eq. (6.5).

$$d = d_{network} + d_{codec} + d_{playout} \tag{6.5}$$

and e refers to the loss probability ranging from 0 to 1, as in eq. (6.6).

$$e = e_{network} + (1 - e_{network}) * e_{playout} \tag{6.6}$$

For the G. 711 codec, d_{codec} is 20 ms, and $d_{playout}$ is set to its default value of 60 ms. $e_{playout}$ is the loss probability caused by overflow at the decoder-side buffer and is set to 0.005 by default. $d_{network}$ and $e_{network}$ are the real time delay and loss probability measured in the experiments.

R-factor is calculated for each of the three streams with various buffering periods, as shown in Table 6.14. It can be seen from Figure 6.18 to Figure 6.20 that with the increase in the buffering period, the value of R-factor decreases for both Buffered Streaming and Q-PASTE, due to the increased delay caused by the traffic shaper. However, with short buffering periods, significant power saving is achieved without compromising in any way the value of the R-factor. Moreover, Q-PASTE is much more energy efficient than Buffered Streaming, while achieving similar R-factor by employing the newly proposed adaptive sleep/wakeup scheduling of WNIC. For instance, Q-PASTE consumes 2.3 Joule of energy with a 12 s buffering period when delivering stream 1 and results in an R-factor of 43.44, while Buffered Streaming has three times higher energy consumption (7.39 Joule) and an R-factor of 42.38 with the same parameter configuration. This demonstrates how Q-PASTE achieves energy efficiency at high quality levels.

Table 6.14 R Factor for 802.11, Buffered Streaming and Q-PASTE

Stream No.	**Buffering Periods: t_{bf} (s)**	**R-Factor**		
		802.11	**Buffered Streaming**	**Q-PASTE**
1	2	71.34	62.36	62.34
	12	71.34	42.38	43.44
	25	71.34	33.69	33.81
2	2	72.83	72.66	68.69
	25	72.83	35.33	40.13
	50	72.83	33.17	39.41
3	2	71.38	71.38	71.38
	8	71.38	61.57	61.45
	16	71.38	55.79	56.09

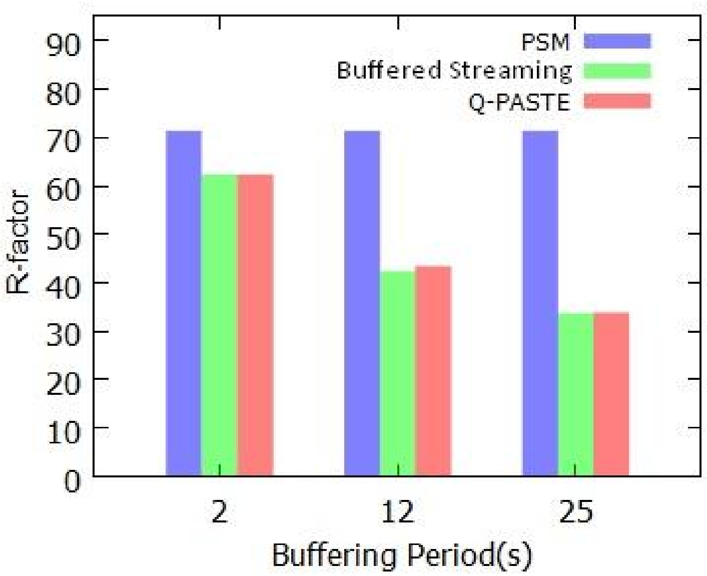


Figure 6.18 R-factor for stream 1

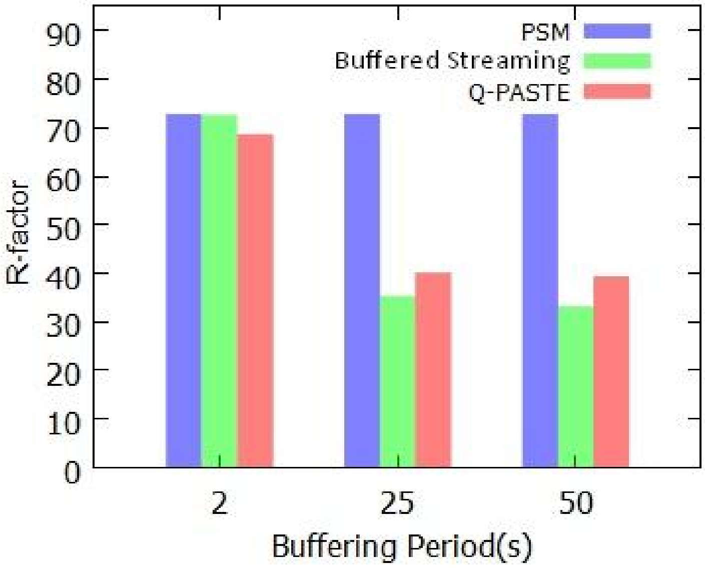


Figure 6.19 R-factor for stream 2

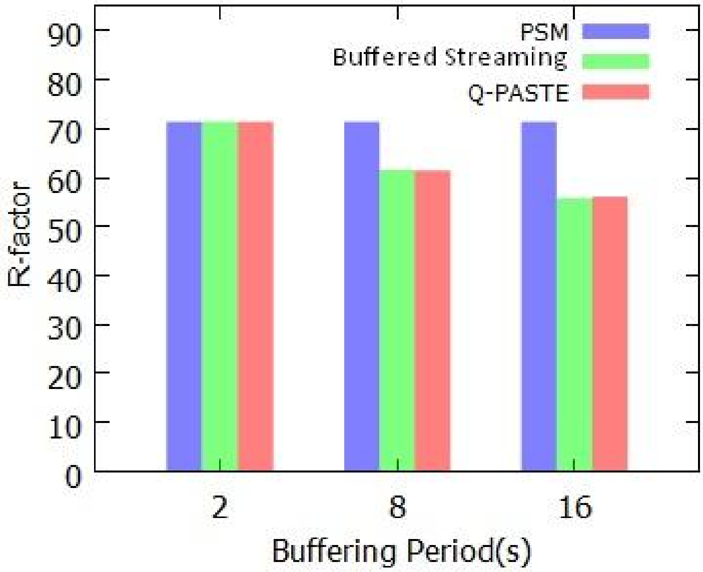


Figure 6.20 R-factor for stream 3

6.4 Summary

This chapter presents the simulation-based testing results for the proposed solutions: static STELA, dynamic STELA, and Q-PASTE. Various network configurations are used to validate the benefits achieved through adopting the proposed solutions. Energy consumption and data delivering quality, in terms of delays and jitter etc, are evaluated respectively.

The performance of static STELA is compared with the results of two widely deployed standards, IEEE 802.11 and IEEE 802.16, respectively, and testing results show how static STELA outperforms the other two solutions. The impact of different parameters such as traffic pattern, data rate and threshold value is analyzed. Traffic patterns and data rates have significant impact on the performance of all the compared schemes as the time spent in active and sleeping modes is directly affected. The threshold value affects static STELA and IEEE 802.16 as it determines the maximum sleeping window size of the WNIC during exponential increase phase. It has been demonstrated how static STELA can be used as either an energy-efficiency-oriented or a QoS-based solution through proper parameter configuration.

Dynamic STELA is tested and compared with the same solutions: IEEE 802.11 and IEEE 802.16, respectively, which it outperforms in terms of energy efficiency and quality levels. Simulation results show how dynamic STELA balances the energy efficiency and quality levels including QoS in terms of packet delay and jitter and QoE in terms of PSNR, demonstrating how dynamic STELA achieves high energy efficiency without compromising quality.

Testing of the cross-layer solution, Q-PASTE, was also conducted. Q-PASTE as a quality-oriented solution designed for multimedia applications incorporates application layer data buffering and MAC layer WNIC scheduling and was compared with IEEE 802.11 and Buffered Streaming. The impact of buffering period at the service gateway, playback time at the client side on its performance are individually analyzed. Energy consumption, playout buffer size, packet delay and R-factor as an indicator of quality of experience are tested, and the results demonstrate that Q-PASTE is capable of delivering multimedia content efficiently in terms of energy consumption while the quality is maintained at high level, outperforming the other solutions to which it was compared.

Chapter 7

Q-PASTE – Prototype-based Testing

This chapter presents the prototype-based testing performed, in order to assess the quality delivered by employing Q-PASTE. First, the prototype software that implements Q-PASTE is introduced. Next the architecture of the software, the library used, implementation of buffering and burst shaping mechanisms of Q-PASTE, and multimedia delivery protocols employed for video delivery testing are presented. The metrics used to assess Q-PASTE performance in the real testing are then described with the relevant testing settings. The results are discussed at the end.

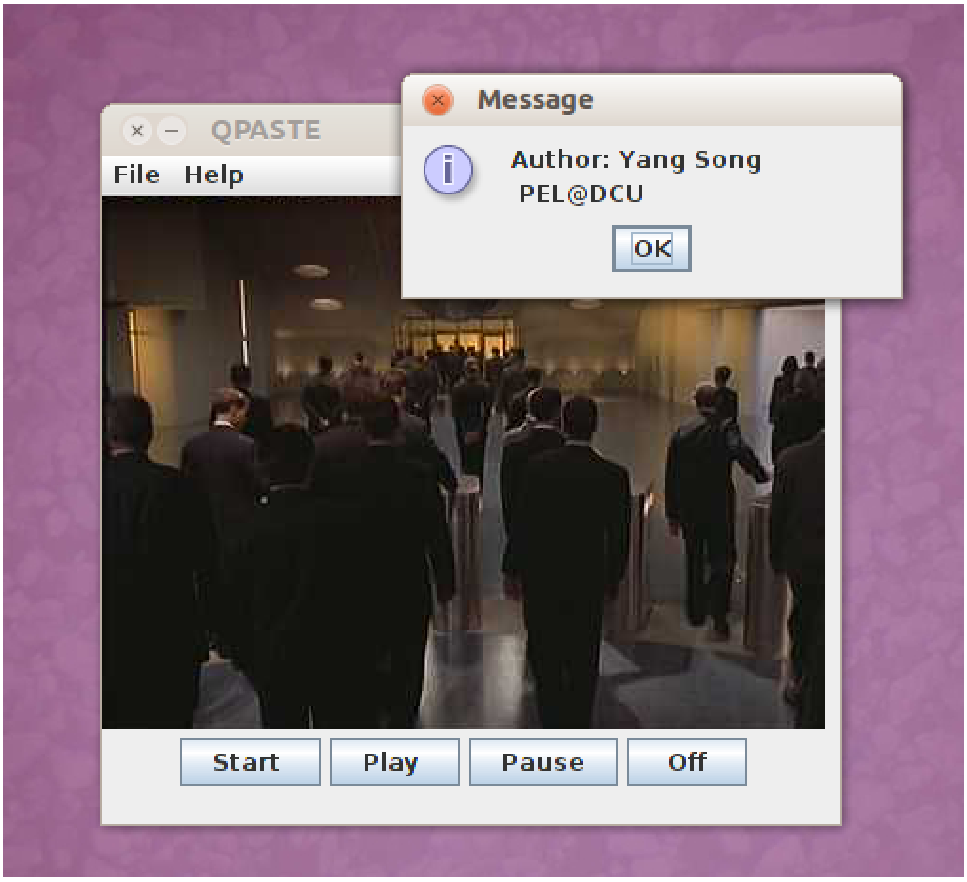
7.1 Q-PASTE Prototype Software System

7.1.1 Client-Server Video Delivery in Java

In order to test Q-PASTE in a real-life situation, a client-sever prototype software system that delivers video content using Q-PASTE-enabled AP was built. The video delivery is made possible by employing RTP protocol on top of UDP. RTSP is used on top of TCP to implement the playback ”remote” control mechanism (e.g. play, pause, tear down). The architecture of this software is inspired by an existing solution [1], which also provided a

[1] ”Streaming Video with RTSP and RTP”-http://www.csee.umbc.edu/ pmundur/courses/CMSC691C/lab5-kurose-ross.html

sample Motion JPEG (MJPEG) video for testing.



Message

Author: Yang Song
PEL@DCU

Figure 7.1 Screen shot: Q-PASTE Java implementation

This prototype is written in Java 7.0 for two reasons. First, Java provides a comprehensive library to support video playback, user interface design and computer networking. Secondly, Java programs can be executed on various platforms, including Linux machines, Windows PCs and mobile devices.

The main purpose of the prototype software is to observe the real burst behaviour demonstrated by Q-PASTE-enabled AP, and the performance gain or loss at the client side. The format of the video transmitted does not affect the testing results in this aspect. Consequently, MJPEG video is used for its simplicity, despite the fact that it lacks support for the modern compression techniques with no temporal encoding support across frames. Specifically, video content of this type consists of consecutive frames independently encoded which can be easily implemented and tested. In contrast, many contemporary video formats feature far more complex packetization schemes and encoding complexity, yet for the purpose of our testing, MJPEG is enough. Due to the flexible design of the software, multi-format videos can be easily supported in the code.

The prototype software shows a user-friendly graphic user interface as in Figure 7.1. Using the *File* menu button, a user can select the remote video source for video playback. The four buttons beneath the main screen allow support for video player control: *Start* for power on, *Play* for play, *Pause* for pause, and *Off* for power off. In fact they trigger the corresponding behaviour on the server side by sending commands to the remote video server. Clicking the four buttons determine sending ”start”, ”play”, ”pause” and ”tear down” RTSP commands respectively. The video delivery and control using RTP and RTSP is illustrated in Figure 7.2.

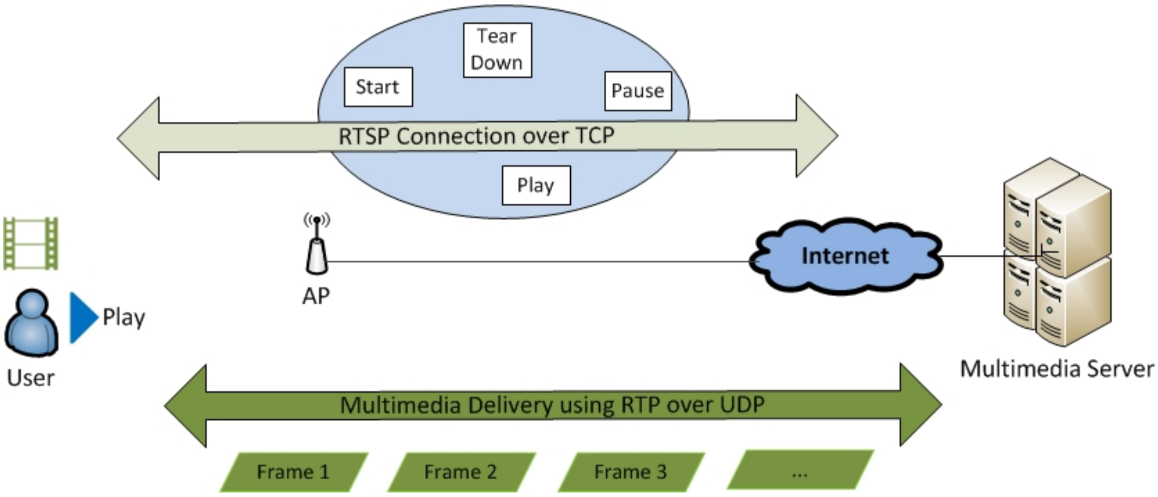


Figure 7.2 Using RTP with RTSP for video delivery

The Q-PASTE-enabled AP and video delivery server have no UI. This is because they are developed as daemon services on the server and AP, respectively. The server transmits data to the AP at varied speeds: faster during the fast start period and slower afterwards. The AP buffers data packets from the server and sends a burst periodically to the client. The AP deploys Q-PASTE, and is described in details next.

7.1.2 Q-PASTE- Testing Server

The server is a piece of software implemented to support RTP and RTSP. The client will establish two socket-based connections to the server via the AP: one RTSP control channel using TCP socket for sending RTSP control message and one RTP data channel using UDP socket for transmitting video data traffic using RTP. The server stores video content and transmits the data to the AP when the playback request is received.

The server listens to client's requests from the RTSP channel and reacts correspondingly. For example, it starts, pauses and terminates video delivery. Once the server receives a request for video content, it starts to transmit video content via the RTP data channel. It obtains the video as an input stream and fetches one frame at a time from the stream. It then constructs a RTP packet with the frame data and sends the RTP packet to the AP via the established data channel. The rate of packet construction and delivery is controlled by a dedicated timer.

7.1.3 Q-PASTE-enhanced AP

Q-PASTE is not deployed at the testing server, as Q-PASTE has AP and client components only, as described in Section 4.5. The Q-PASTE-enabled AP buffers packets transmitted from the server and shapes them into bursts before relaying them periodically to the client. This section describes the implementation of the Q-PASTE-enhanced AP. The architecture of the prototype software is illustrated in Figure 7.3. The yellow AP block pre-fetches content and caches it before sending it to the client. The drain rate reflects the speed in which the AP sends the buffer to the multimedia player, and is the encoding rate for the video content.

The major components of the Q-PASTE implementation are the *Frame Cache* at AP, and the *Queue Timer* at AP as illustrated in Figure 7.4. The figure also shows the *Display Timer* component deployed at the client.

The Frame Cache is implemented as a First In First Out (FIFO) queue on the AP.

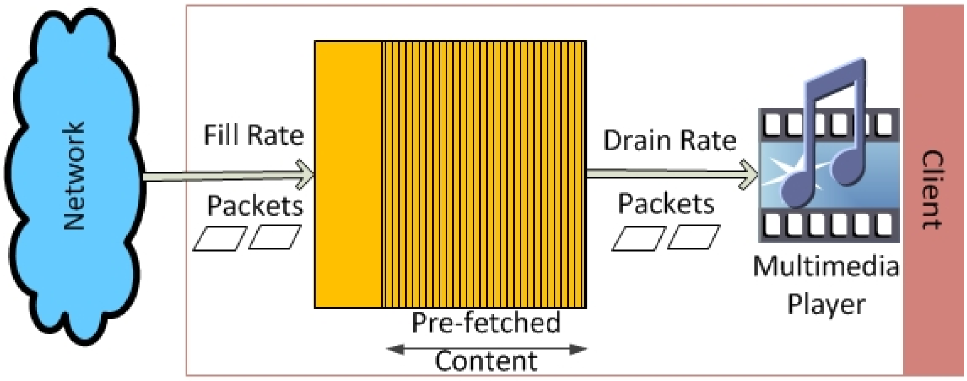


Figure 7.3 Video delivery using the prototype

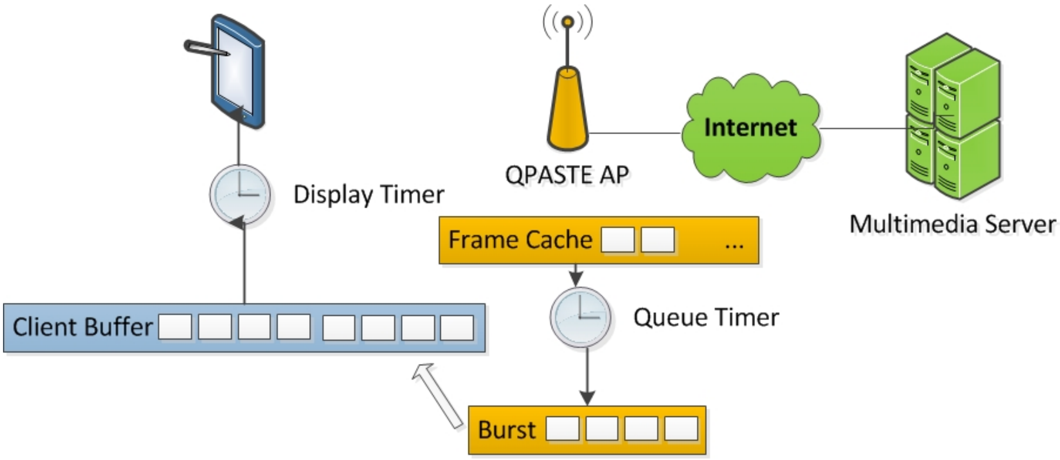


Figure 7.4 Key elements of Q-PASTE

A dedicated Queue Timer controls the behaviour of this cache. This timer-governed cache collects the arriving video packets and according to the Q-PASTE policy, the AP will buffer packets received from the server and shape them into bursts before forwarding them to the client. The cache size is set to 250 frames, which is the total number of packets transmitted during the fast start period.

The Queue Timer controls the scheduling for data transmission to the client from AP's buffer. During the fast start period, all data are transmitted from the AP to the client at the same rate as the rate data is received from the server. Once the fast start phase ends, the server sends data at the same rate while the AP does not relay data until the queue timer expires. The maximum timeout interval set for the Queue Timer is calculated according to the client buffer size, encoding rate etc using the Q-PASTE algorithm, as illustrated in eq. (4.33). In order to show the effects of various buffering periods on the playback at the client side, various Queue Timer settings are configured in the test.

7.1.4 Q-PASTE Client

Q-PASTE client is composed of a regular video player to which a *Client Buffer* is attached and a *Display Timer* is deployed.

The Client Buffer, which is used in many contemporary multimedia players to compensate for network condition variations, stores the received frames from the AP. At the same time when the playback takes place, the multimedia player fetches frames from the buffer at the encoding rate. In order to provide smooth playback, the Client Buffer should not be empty when the data fetching is performed by the player.

The Display Timer controls the frame rate of video playback. The Client Buffer receives the data burst from the AP. When the buffer is empty, the video will freeze at the last frame received by the client. When the buffer overflows, the video will lose frames. When the buffer is neither empty nor overflows, the Display Timer enables fetching one frame at a time from the buffer and sends it for display on the screen.

7.2 Real Tests on the Q-PASTE Prototype Software

7.2.1 Testing Settings

The multimedia server runs on a Thinkpad T400 laptop with Intel Core2 P8700 @2.53 GHz CPU and 4G RAM. It connects to the AP through IP address and port number. The IP address of the multimedia server and the dedicated port number is 192.168.1.1 and 4567 respectively. Both the AP and client software pieces are implemented on the same machine, which is another Thinkpad T400 laptop equipped with 14 inch screen, Intel Core2 P8700 @2.53 GHz CPU and 4G RAM. The AP software receives data from the server using the dedicated IP address (i.e. 192.168.1.2) and port number (i.e. 5678), and then shapes and forwards the bursts to the client using localhost address and another port number (i.e. 5679).

The MJPEG video [2] is of the length of 1800 frames. The sample video uses 60 fps to enforce smooth playback. The fast start period is set to 5s [3]. The Linux machine running on Ubuntu 13.04 has the video playback client deployed. Both the video server and the Q-PASTE-enabled AP are deployed on the Windows machine running on Windows 7. The lab setting is shown in Figure 7.5.

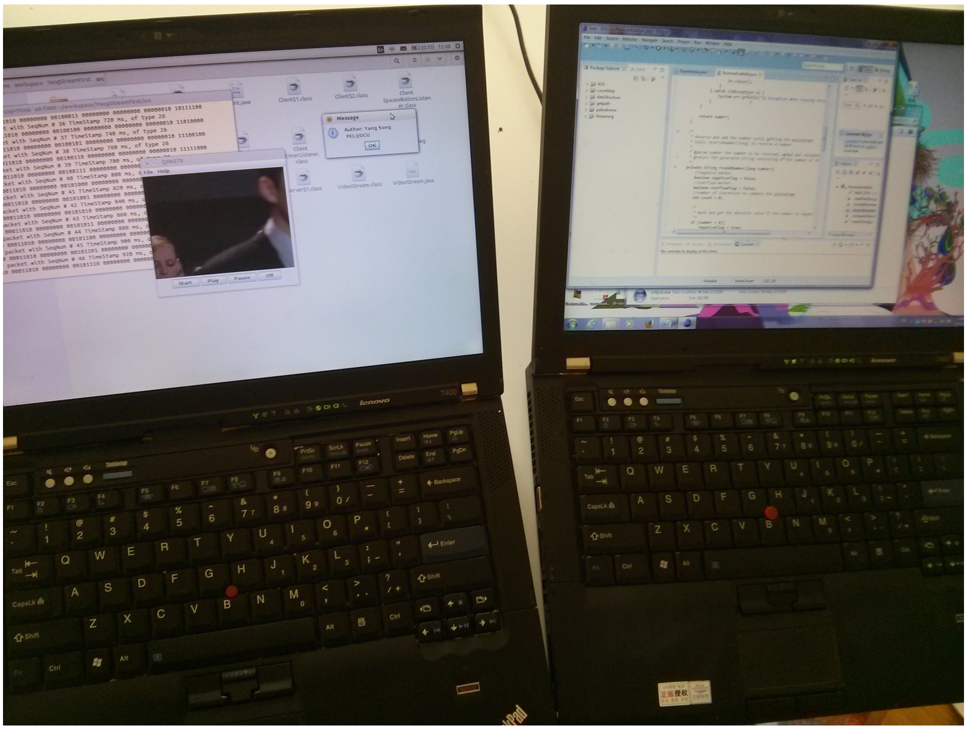


Figure 7.5 Real test lab setting: client and server laptops

In order to test the effect of data shaping at the access point on the client side playback, two parameters are tested respectively:

- **Buffering period** is the time between two consecutive bursts released from the AP. The maximum buffering period is calculated based on the encoding rate and playout buffer size at the client side. The value is used as an upper limit for the buffering timer, and various buffering periods are configured at the AP to demonstrate the

[2] http://www.csee.umbc.edu/ pmundur/courses/CMSC691C/movie.Mjpeg

[3] http://www.microsoft.com/windows/windowsmedia/howto/articles/BroadcastDelay.aspx

effect on delivery performance.

- **Playback starting time**. The staring point of playback directly impacts the playout buffer size mainly due to two reasons. First, the playout buffer size is determined by the size of data that the AP has transmitted to the client, and the size of the data that has been fetched by the player, at any time point. Second, the data receiving rate at the client size varies along the playback, i.e. faster during the fast start period, and slower afterwards. In this case, playback at the end of fast start period leads to a fuller buffer in terms of data size while playback from the beginning of data transmission causes emptier buffer. Therefore the two extreme cases are tested respectively. For the tested video clip, data is transmitted at 120 fps during the fast start period which lasts 2.5s, and then at the encoding rate of 60 fps from the server to the AP.

The testing parameters are listed in Table 7.1.

Table 7.1 Prototype-based Testing Parameters

Parameter	Value
Stream length	30s
Encoding rate	60 fps
Fast start duration	2.5s
Buffering period	1s 2s 5s
Playback time	Immediate After fast start

To ensure high delivery quality and smooth playback, the size of the smoothing buffer at the client side should be kept above zero all the time. Therefore the real time buffer is monitored and the buffer size is measured at the client side to assess delivery quality.

Q-PASTE is tested against the Buffered Streaming solution [204], which was implemented at the AP and the client. The original results obtained by the authors in [204] were

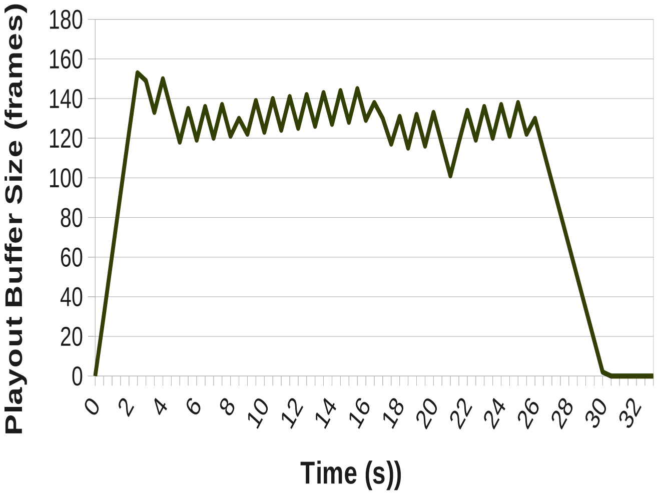


Figure 7.6 Playout buffer size with 1s buffering periods (Playback: immediate, Scheme: Q-PASTE)

not used as the results do not meet the testing requirements of the work reported in this thesis. STELA needs to be tested in terms of delay, jitter, R-factor and the playout buffer size whilst in [204] the authors only focused on the buffer size. Therefore this approach was re-implemented and the required functionality was added to the implementation. The results obtained by my implementation are not identical to the original authors due to different network condition configurations. However, the conclusions are the samełthe fast start scheme is able to smooth out the delay introduced by deliberate data buffering.

7.2.2 Impact of Buffering Period on Playout Buffer Size

The impact of buffering period on the performance of Q-PASTE in terms of playout buffer size is shown in Figure 7.6, Figure 7.7 and Figure 7.8. According to eq. (4.33), the maximum value of buffering period that can be adopted by the AP without affecting smooth playback is calculated according to the playout buffer size and the encoding rate of the video being played. In the case tested, the maximum buffering period is 5s. Three values of buffering periods are individually tested: 1s, 2s, and 5s.

The average buffer size and the average deviation is listed in Table 7.2. It can be

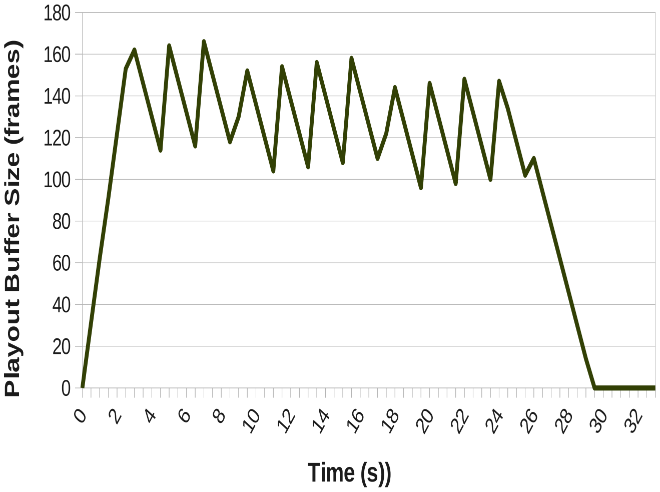


Figure 7.7 Playout buffer size with 2s buffering periods (Playback: immediate, Scheme: Q-PASTE)

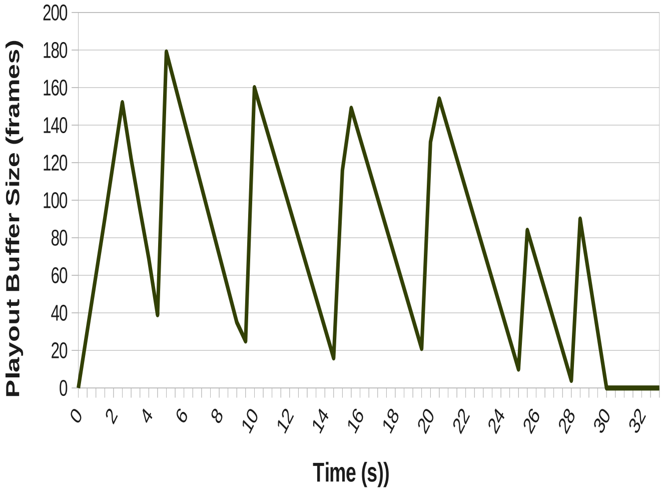


Figure 7.8 Playout buffer size with 5s buffering periods (Playback: immediate, Scheme: Q-PASTE)

seen that longer buffering period leads to more significant fluctuations, e.g. higher average deviations, due to larger bursts generated by Q-PASTE. The observation is similar to the analysis made in Section 6.3.2. For example, with a buffering period of 1s, the higher value of playout buffer size stays at approximately 180 frames, while the lower value of the buffer size increases to approximately 300 frames when a longer buffering period of 5s is adopted by the AP.

Table 7.2 Buffer Size for Q-PASTE in Prototype Test

Buffering Periods: t_{bf} (s)	**Average Buffer Size (frames)**	**Average Variation (frames)**
1	125	23
2	122	28
5	80	40

7.2.3 Impact of Buffering Period on End-to-End Delay

The impact of buffering period on average packet delay is shown in Table 7.3. As end-to-end delay is the duration from the time a packet is transmitted from the server to the time it arrives at the client, this value is the sum of the processing time, the propagation time etc, and the duration that the packet has been buffered for at the AP. Therefore the longer the buffering period, the longer average packet delay. For example, in Table 7.3, the packet delay for Q-PASTE is 1.33s when buffering period is 1s , and the delay increases to 5.36s when the buffering period is set to 5s.

Table 7.3 End-to-End delay for Buffered Streaming and Q-PASTE in prototype test

Buffering Periods: t_{bf} (s)	**Average Packet Delay(s)**	
	Buffered Streaming	**Q-PASTE**
1	1.48	1.33
2	2.67	2.62
5	5.21	5.36

7.2.4 Impact of Playback Time on Playout Buffer Size

As the playback starting time affects seriously the playout buffer size, we illustrate its impact through testing the two extreme cases in this section: immediate playback and after fast start playback. The playout buffer size fluctuates with the playback and is shown in Figure 7.9.

The results from Figure 7.9 show that compared with the scenario of after fast start playback, the playout buffer size is smaller during the first 2.5s of testing when immediate

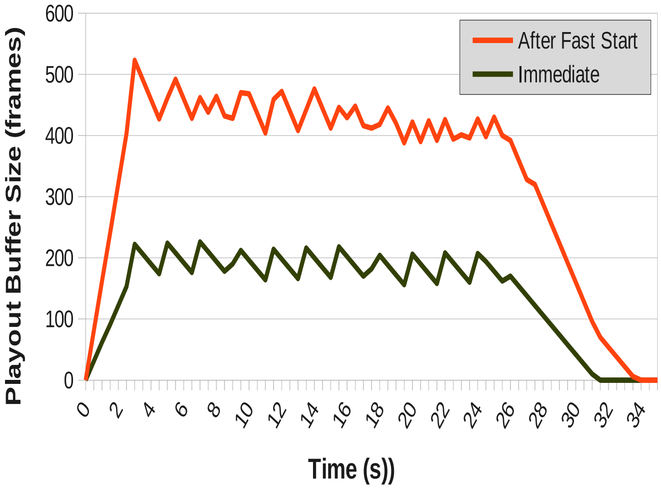


Figure 7.9 Playout buffer size with 2s buffering periods (Playback: immediate, Scheme: Q-PASTE)

playback is performed. The reason is that packets are being fetched earlier in the latter case, while the buffer filling rate is the same in the two scenarios. With immediate playback, it can be also observed that the playout buffer size decreases linearly after 26s until it reaches 0. This is due to the fact that all frames have been received by the player during the first 26s, after which the playout buffer is drained by the player without being filled by the server. When the playback starts after the fast start period, the buffer size reaches its peak at 2.5s, which is the end of fast start period, and then drops to a lower level as frames are being fetched for display when the AP is forwarding the rest of the frames to the player at the same time. The observations are similar to what has been analyzed in Section 6.3.3.

Although early start time leads to lower buffer size levels, it can be seen from Figure 7.9 that the buffer size does not fall to zero and therefore the frames can be fetched in time for display and smooth playback is provided.

7.2.5 Quality Evaluation

In this section, we assess the user quality of experience levels achieved by employing Q-PASTE at the service gateway and the client host. PSNR, the peak signal-to-noise ratio, is used to approximate the user perceived quality levels. The metric measures the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. In our case, the signal is the original data, and the noise is the error introduced by the adaptive data scheduling and WNIC sleep/wakeup control.

PSNR value is calculated according to eq. (6.2) [234], which is based on the maximum bit rate of the video, the expected throughput, and the actual throughput. The PSNR scores obtained for Buffered Streaming and Q-PASTE are listed in Table 7.4 which shows that the PSNR scores achieved by Q-PASTE is similar to that obtained by Buffered Streaming.

The actual throughput is affected by the buffering period, and it can be seen from Table 7.4 that longer buffering period leads to lower PSNR. The main reason is that longer buffering period leads to more packets being buffered at the AP, which has limited capacity and packet loss is incurred.

Table 7.4 PSNR for Buffered Streaming and Q-PASTE in prototype test

Buffering Periods: t_{bf} (s)	**PSNR(dB)**	
	Buffered Streaming	**Q-PASTE**
1	48.18	44.26
2	40.25	43.07
5	38.44	35.91

7.3 Summary

In this chapter, we present a real life testing on the performance of Q-PASTE in terms of quality, compared with that of other existing solutions such as Buffered Streaming.

The tests were conducted on two Thinkpad T400 laptops. One of the laptops works as the server, on which a multimedia server software were implemented and deployed. Both the AP and the client player software run on the other laptop. The AP software was running as a daemon, which receives packets from the server and according to the traffic shaping mechanism, forwards the data to the client software. Data communications are enabled by IP addresses and port numbers. The client player stores the received data in the playout buffer and fetches it from the buffer for display.

The video delivery quality of the compared solutions are individually studied. Two parameters, i.e. buffering period and playback staring time, are varied in different test cases in order to demonstrate their impacts on delivery performance. User quality of experience levels are measured in terms of packet delay, playout buffer size, and PSNR. Testing results show that Q-PASTE, as an energy efficient solution, maintains high delivery quality, through adaptive application layer traffic shaping and MAC layer WNIC control.

Chapter 8

Conclusion and Future Work

In this thesis we have proposed three energy efficient solutions for WLANs: static STELA, dynamic STELA and Q-PASTE.

This chapter presents a summary of our work and final outcomes achieved through different experiments. It also highlights the limitations of our work and [resents some suggestions for future work.

8.1 Problem Overview

The convergence of the two fastest-growing communications technologies of all time– mobile devices and the wireless networks makes possible the support for various new services and create a large market. Users are able to log onto the Internet anywhere, anytime. While serving the basic purpose of allowing for voice communication, mobile phones, also provide many additional functions. Among the various applications enabled by the mobile devices, the popularity of multimedia applications running on increasingly capable handheld devices is growing significantly. People can listen to radio, watch live football matches or stream video as long as they have Internet access, often via wireless technologies.

However, the integration of wireless network interfaces in mobile phones helped to achieve seamless connectivity, but the constraint of limited battery capacity is challenged by this integration. Battery life is critical for the success of any portable device platform as it has a significant impact on the user quality of experience. Hence a large body of research has focused on the energy efficiency of mobile devices.

Among all the components of a mobile device, the Wireless Network Interface Card (WNIC) consumes a large amount of energy and therefore is studied in this thesis. A typical WNIC has several states, each involving different energy consumption: transmit, receive, idle, and sleep. While in transmit and receive mode, the WNIC consumes the highest amount of energy. During sleep mode, which is also known as power save mode, it demands far less energy. The sleep/wakeup schedule of WNIC is controlled by the Medium Access Control (MAC) layer and can be wisely manipulated for the purpose of energy saving. Energy efficiency at MAC layer is normally achieved by switching the WNIC card to low energy states (i.e. sleep) when the device does not have data to send or receive. However, inactive WNIC might cause slow response to upcoming data traffic leading to long packet delay or even data loss. Therefore the proposed solutions in this thesis aim at providing high energy efficiency without compromising delivery quality, especially when delivering multimedia content, as multimedia streaming applications are sensitive to quality degradation.

8.2 Thesis Contributions

This thesis makes four major contributions to the state of the art: static STELA, dynamic STELA, Q-PASTE and a survey. They will be summarized next.

8.2.1 Comprehensive Survey

In this thesis, we presents the results of our extensive study conducted and provide a comprehensive survey of existing standards within the field of wireless mobile communications and related protocols which are widely deployed at each layer of the protocol stack. First, Wireless Wide Area Network (WWAN) is introduced. Technologies used for WWAN are presented from the first generation to the fourth generation, according to the development time-line. Basic architecture and the provided service are illustrated with details. The development of Wireless Local Area Network (WLAN) is then presented with details of each important version of the IEEE 802.11 family and important extension explained. Wireless Metropolitan Area Network (WMAN), as the supporting technology used for Worldwide Interoperability for Microwave Access (WiMAX), is also discussed. Corresponding protocols within the IEEE 802.16 family and deployment architecture is then illustrated. Last the Wireless Personal Area Network (WPAN), which is used for short range communications and widely applied in technologies such as Wireless USB, ZigBee, and Bluetooth is discussed. The bitrate and mobility of each technology is compared.

First the wireless network technologies are introduced, then the protocol stack is presented. Standards deployed at each network layer, from application to physical layer are individually described. Energy efficient features, if incorporated, are explained in details.

Energy efficiency is normally achieved through compromising data delivery quality, and therefore the tools used for measuring Quality of Service (QoS) and Qaulity of Experience (QoE) are introduced, respectively. On the one hand, QoS, as an objective measuring method, refers to the ability to provide differentiated treatment to different applications, users, or data flows, or to guarantee a certain level of performance to a data flow. Hence, metrics such as packet delay, loss and jitter are commonly used for QoS evaluation. On the other hand, QoE measures user's perception of the service quality and does not care about the network itself. Although QoE is subjective, it can also be evaluated using the results of objective evaluation. The most often used metrics include Mean Opinion Score (MOS), Perceptual Speech Quality Measurement (PSQM), Perceptual Evaluation of Audio Quality (PEAQ), E-model, etc.

8.2.2 Static STELA

Static STELA is proposed as an energy efficient MAC layer solution with configurable parameters which determine the balance between energy efficiency and QoS. Static STELA consists of three phases which control the sleeping/wakeup schedule of the WNIC: slow start, exponential increase, and linear increase. Different from most MAC layer solutions, such as IEEE 802.11 which adopts a static sleeping window of the WNIC, the termination and initiation of each phase is dynamically controlled based on the real time traffic conditions when static STELA is employed. The solution is based on the fact that there are bursty and relatively regular traffic patterns.

The slow start phase involves wireless devices waking up regularly to sample for incoming data. This guarantees the data within a burst is received successively without long gap. The exponential phase is repeated until a threshold value is achieved. During this procedure the sleep interval doubles if there is no incoming traffic, otherwise the slow start phase is started. High energy efficiency can be achieved as the WNIC is put into low energy consumption mode. The linear increase phase is activated once the threshold value is reached and deactivated when a packet is received by the client host. In this stage, the sleeping window increases by one each time the WNIC wakes up and the sleeping window does not grow too aggressively and therefore there is no serious quality degradation.

It is evident that the configuration of the threshold value determines the performance of static STELA. Large threshold values can be configured when the battery level is low and extending device usage time is more critical than high level of QoS. On the other hand, small values of the threshold provide better delivery quality, but shorter battery lifetime.

8.2.3 Dyanmic STELA

Dynamic STELA extends static STELA by incorporating an additional threshold adjusting phase which dynamically adjusts the threshold value.

The threshold adjusting phase is triggered every time when an application session terminates or a new session is established. For each specific application, the data arrival pattern exhibits high regularity which indicates the first round of observed data arrival interval could be used as a reference to the next schedule. Therefore the first two rounds of data bursts are monitored and the threshold value is set accordingly as prediction of the future traffic arrival pattern.

As the threshold value is adjusted according to historical data arrival pattern, it is highly likely that the WNIC is waken up in time before the next round of data bursts arrive. Both mathematical analysis and simulation results demonstrate that dynamic STELA provides an energy efficient solution without decreasing QoS level seriously, compared with existing solutions.

8.2.4 Q-PASTE

While dynamic STELA conserves power by wisely scheduling the WNIC at MAC layer, multimedia delivery-oriented solution–Q-PASTE further prolongs the sleeping interval by grouping small pieces of data from the server into large chunks of bursts at the service gateway before relaying them to the client.

Traffic shaping at the service gateway is performed at the application layer, and additional control information is used for cross-layer information exchange. As multimedia applications are very sensitive to packet delay, which might cause interrupted playback at the client side, fast streaming is employed at the beginning of the content streaming process. During fast streaming, data packets are directly released to the client without being buffered, with the purpose of filling the playout buffer as quickly as possible. After that, data is gathered into a large burst in the buffer by the service gateway. The burst releasing time is calculated according to the playout buffer size and encoding rate of the stream. Smooth playback is guaranteed as the playout buffer is never drained empty during the playback process.

Q-PASTE employs dynamic STELA at the MAC layer of the client side. Due to the deliberate traffic shaping, the feature of burstiness is better utilized by dynamic STELA as the prediction of threshold value according to historical traffic provides higher accuracy. It is proved that Q-PASTE is capable of delivering multimedia content in an energy efficient way while maintaining high QoS levels.

8.3 Publications Arising from this Work

- Y. Song, B. Ciubotaru, and G.-M. Muntean, “A Slow-start Exponential and Linear Algorithm for Energy Saving in Wireless Networks,” *Broadband Multimedia Systems and Broadcasting (BMSB), 2011 IEEE International Symposium on* , pp.1–5, June 2011.

- Y. Song, B. Ciubotaru, and G.-M. Muntean, “Application-aware Adaptive Duty Cycle-based Medium Access Control for Energy Efficient Wireless Data Transmissions,” *Local Computer Networks (LCN), 2012 IEEE 37th Conference on* , pp.172–175, Oct. 2012.

- Y. Song, B. Ciubotaru, and G.-M. Muntean, “Q-PASTE: A Cross-Layer Power Saving Solution for Wireless Data Transmission,” *IEEE International Conference on Communications (ICC), IEEE International Workshop on Energy Efficiency in Wireless Networks & Wireless Networks for Energy Efficiency (E2Nets)*, Jun. 2013.

- Y. Song, B. Ciubotaru, and G.-M. Muntean, “STELA: A Transceiver Duty Cycle Management Strategy for Energy Efficiency in Wireless Communications”, *Local Computer Networks (LCN), 2012 IEEE 38th Conference on* , Oct. 2013.

8.4 Future Work

There are several potential subjects for future work. Next each of them is discussed.

8.4.1 Accurate Data Estimation

The current data pattern estimation is based on the observation of first two rounds of data packets, but the accuracy could be further improved. For example, the real time fluctuation of data arrival interval could be taken into consideration based on which a mathematical estimation method could be employed. Moreover, both an average inter-packet interval and the most recent data arrival pattern can be taken into consideration when estimating the next packet arrival time. For example, the concept of time series can be introduced in data prediction. Time series refers to a sequence of observed values on a variable. It can be used to extract important features, i.e. burst arrival pattern in STELA, in order to build a model which can be used for forecasting [236] [237].

8.4.2 Ad-hoc Supported MAC scheme

All the proposed schemes in this thesis including static STELA, dynamic STELA and Q-PASTE are designed for infrastructure-based wireless LANs. Although this type of network is widely used in everyday life, i.e. the coverage area of Wi-Fi is expanding extremely quickly, it is not applicable in some scenarios such as underwater acoustic systems, which asks for ad-hoc-based networks and requires long battery life. In infrastructure-free networks, there is no access point available and the syncronization among all wireless nodes depends on the topology and protocol used. As battery life is an extremely critical factor in the deployment of ad-hoc networks, the energy efficient solutions proposed in this thesis can be extended to support this type of network and the buffering scheme needs to be modified. In our work, data packets are buffered at the access point when the WNIC of the wireless host is switched off. Access point compared to wireless hosts are not sensitive to battery shortage and therefore can afford the time of waiting when the receiver is sleeping. However, ad-hoc networks require an scheme to synchronize the two communicating nodes so that the sender and the receiver wake up at the same time in order not to waste energy of either of them.

8.4.3 Quality-oriented Solution for Various Application Types

Fast Start as a commonly used technique for multimedia streaming applications is utilized in Q-PASTE to conserve power for a single type of traffic. In the future work, Q-PASTE can be extended to adopt different techniques for various application types such as HTTP-based web browsing or FTP-based file transfer at the service gateway so that all types of application content can be enjoyed by the users at high quality levels and with low power consumption.

8.4.4 Subjective Testing

Whilst quality of service metrics including energy consumption, packet delay, jitter and quality of experience metrics such as R-factor and PSNR are used to evaluate the performance of the three proposed solutions, this could be further extended. For example, subjective quality evaluation could be performed such as those based on the Double Stimulus Impairment Scale (DSIS) [238], in order to demonstrate the benefits achieved by the proposed solutions in terms of quality levels.

8.4.5 Traffic Prioritization

Both static STELA and dynamic STELA are capable of conserve energy for mobile devices, however, Q-PASTE adopts an additional fast start mechanism as part of the traffic shaping, which requires modifications at the service gateway. The mechanism is specially employed for multimedia streaming applications which are very sensitive to quality level decreases. In order to provide power conservation for all type of traffic, the service gateway can incorporate a traffic prioritization scheme within the traffic shaping algorithm to differentiates traffic types according to their sensitivity to packet delay, jitter etc. In other words, data flows which ask for high quality are assigned higher priority and are released sooner than those of lower priority. For example, multimedia streaming content is given higher priority than file transfer data, and hence better user quality of experience levels at the client side can be provided.

Bibliography

[1] G. Prieto, I. Pichel, D. Guerra, P. Angueira, J. Matias, J. Ordiales, and A. Arrinda, "Digital radio mondiale: broadcasting and reception," in *Electrotechnical Conference, 2004. MELECON 2004. Proceedings of the 12th IEEE Mediterranean*, vol. 2, pp. 485–487 Vol.2, 2004.

[2] J. Morgade, A. Usandizaga, P. Angueira, D. de la Vega, A. Arrinda, M. Velez, and J. Ordiales, "3dtv roll-out scenarios: A dvb-t2 approach," *Broadcasting, IEEE Transactions on*, vol. 57, no. 2, pp. 582–592, 2011.

[3] R. D. Bari, M. Bard, A. Arrinda, P. Ditto, G. Araniti, J. Cosmas, L. Kok-Keong, and R. Nilavalan, "Measurement campaign on transmit delay diversity for mobile dvb-t/h systems," *Broadcasting, IEEE Transactions on*, vol. 56, no. 3, pp. 369–378, 2010.

[4] B. Lehane, N. O'Connor, A. F. Smeaton, and H. Lee., *A System For Event-Based Film Browsing*, vol. 4326 / 2006, pp. 334–345. Berlin / Heidelberg, Germany: Springer, 2006.

[5] "Cisco visual networking index: Global mobile data traffic forecast update, 2012 2017," tech. rep., Cisco System, Inc, Feb. 2012.

[6] M. Viredaz, L. Brakmo, , and W. Hamburgen, *Energy Management of Handheld Devices,*, pp. pp.44–52. New York: ACM Press, 2003.

[7] A. Kinane, D. Larkin, and N. O'Connor., "Energy-efficient acceleration of MPEG-4 compression tools," *EURASIP Journal on Embedded Systems*, vol. 2007, 2007.

[8] C. Kihwan, K. Kwanho, and P. Massoud, "Energy-aware MPEG-4 fgs streaming," in *Proceedings of the 40th annual Design Automation Conference*, DAC '03, (New York, NY, USA), pp. 912–915, ACM, 2003.

[9] H. Liu, E. Zarki, and Magda, "Adaptive source rate control for real-time wireless video transmission," *Mob. Netw. Appl.*, vol. 3, pp. 49–60, June 1998.

[10] B. Marchi, A. Grilo, and M. Nunes, "DTSN distributed transport for sensor networks," in *Proc. IEEE Symposium on Computers and Communications (ISCC'07*, pp. 165 – 172, 2007.

[11] A. Dunkels, J. Alonso, T. Voigt, and H. Ritter, "Distributed TCP caching for wireless sensor networks," in *Proceedings of the 3 rd Annual Mediterranean Ad-Hoc Networks Workshop*, 2004.

[12] R. C. Shah and J. Rabaey, "Energy aware routing for low energy ad hoc sensor networksenergy-efficient wake-up scheduling for data collection and aggregation," in *Proc. WCNC2002 Wireless Communications and Networking Conf. 2002 IEEE*, vol. 1, pp. 350–355, 2002.

[13] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, pp. 660–670, 2002.

[14] J. Cheon and H. Cho, "A delay-tolerant OFDMA-based MAC protocol for underwater acoustic sensor networks," in *Underwater Technology (UT), 2011 IEEE Symposium on and 2011 Workshop on Scientific Use of Submarine Cables and Related Technologies (SSC)*, pp. 1 –4, april 2011.

[15] I. Rhee, A.Warrier, M. Aia, M. Jeongki, and M. Sichitiu, "Z-MAC: A hybrid MAC for wireless sensor networks," *Networking, IEEE/ACM Transactions on*, vol. 16, pp. 511 –524, june 2008.

[16] A. Mena and J. Heidemann, “An empirical study of real audio traffic,” in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 1, pp. 101–110, 2000.

[17] S. Sarvotham, R. Riedi, and R. Baraniuk, “Connection-level analysis and modeling of network traffic,” in *Proceedings of Internet Measurement Workshop (IMW)*, 2001.

[18] “IEEE standard for local and metropolitan area networks– part 16: Air interface for fixed broadband wireless access systems,” 2001.

[19] V. Paxson, “Measurements and analysis of end-to-end internet dynamics,” April 1997.

[20] C.-F. Chiasserini and R. Rao, “Improving battery performance by using traffic shaping techniques,” *Selected Areas in Communications*, vol. 19, pp. 1385–1394, 2001.

[21] M. Uhlirz, “Concept of a gsm-based communication system for high-speed trains,” *IEEE 44th Vehicular Technology Conference*, vol. 2, pp. 1130–1134, 1994.

[22] “Digital cellular telecommunications system (phase 2+); general packet radio service (GPRS); overall description of the GPRS radio interface; stage 2(3GPP TS 03.64 version 8.9.0 release),” 1990.

[23] S. Zhao and X. Zhou, “Enhanced data rate for global evolution-EDGE,” *Publishing House of Electronic Industry*, 2009.

[24] E. S. M. Group, “Universal mobile telecommunications systems: Objectives and overview (UMTS 01.01),,”

[25] “3GPP TS 36.201- Evolved universal terrestrial radio access (E-UTRA): Long term evolution (LTE) physical layer,” 1997.

[26] “ITU global standard for international mobile telecommunications IMT-advanced,” 2012.

[27] “Wireless LAN medium access control (MAC) and physical layer (PHY) specification, ieee Std. 802.11.,” 1997.

[28] "IEEE 802.11b-1999, IEEE standard for local and metropolitan area networks specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications high speed physical layer extention in the 2.4 ghz band," 1999.

[29] "IEEE 802.11a-1999, IEEE standard for local and metropolitan area networks specific requirements part 11: Wireless LAN medium access control (MAC) and physical lyaer (PHY) specifications high speed physical layer in the 5 ghz band.," 1999.

[30] "IEEE 802.11g- 2003, IEEE standard for local and metropolitan area networks specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 4: Further higher data rate extension in the 2.4 gh band june," 2003.

[31] "IEEE standard for information technologytelecommunications and information exchange between systemslocal and metropolitan area networksspecific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 5: Enhancements for higher throughput.," 2009.

[32] "IEEE standard for information technologytelecommunications and information exchange between systemslocal and metropolitan area networksspecific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 1: Radio resource measurement of wireless LANs.," 2008.

[33] "IEEE standard for information technology–telecommunications and information exchange between systems–local and metropolitan area networks–specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 10: Mesh networking.," 2011.

[34] "Zigbee specification: Zigbee document 053474r13 version 1.1," 2006.

[35] "Specification of the bluetooth system version 1.1b. Bluetooth special interest group," 2001.

[36] R. Braden, "RFC 1122 - Requirements for internet hosts – communication layers," 1989.

[37] A. S. Tanenbaum, "Computer networks," *Prentice Hall 2002*.

[38] R. Fielding, J. Gettys, and J. M. et al., "RFC 2616 - Hypertext transfer protocol – http/1.1,," 1999.

[39] J. Postel and J. Reynolds, "RFC 959 - File transfer protocol (ftp)," 1985.

[40] J. B. Postel, "RFC 2821 : Simple mail transfer protocol," 2001.

[41] M. Rose and J. Myers, "RFC 1939 - post office protocol," 1996.

[42] M. Crispin, "RFC 3501 - Internet message access protocol," 2003.

[43] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RFC 3550 - RTP: a transport protocol for real-time applications," 2003.

[44] C. Huitema, "RFC 3605 - Real time control protocol (RTCP) attribute in session description protocol (sdp)," 2003.

[45] H. Schulzrinne, A. Rao, and R. Lanphier, "RFC 2326 - Real time streaming protocol (RTSP)," 1998.

[46] M. Handley, V. Jacobson, and C. Perkins, "RFC 4566 - SDP: Session description protocol," 2006.

[47] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "RFC 3261 - SIP: Session initiation protocol," 2002.

[48] "RFC 23009-1 : Information technology – dynamic adaptive streaming over http (dash) – part 1: Media presentation description and segment formats," 2002.

[49] I. S. Institute, "RFC 793 - Transmission control protocol," 1981.

[50] J.Poste, "RFC 768 - User datagram protocol," 1980.

[51] Sandeep, "An experimental study of TCPs energy consumption over a wireless link," in *European Personal Mobile Communications Conference, IEEE*, 2001.

[52] M. Zorzi and R. Ramesh, "Energy efficiency of TCP in a local wireless environment," *Mob. Netw. Appl.*, vol. 6, pp. 265–278, June 2001.

[53] S. Giannoulis, C. Antonopoulos, E. Topalis, A. Athanasopoulos, A. Prayati, and S. Koubias, "TCP vs. UDP performance evaluation for CBR traffic on wireless multihop networks," in *5th International Symposium on Communication Systems, Networks and Digital Signal Processing*, (Greece), pp. 154 – 158, 2006.

[54] R. Stewart, Q. Xie, and K. M. et al., "RFC 2960 - Stream control transmission protocol," 2000.

[55] M. Handley, S. Floyd, and E. Kohler, "RFC 4340 - Datagram congestion control protocol," 2006.

[56] I. S. Institute, "RFC 791 - Internet protocol," 1981.

[57] R. Hinden and S. Deering, "RFC 2460 - Internet protocol, version 6," 1998.

[58] J. Postel, "RFC 792 - Internet control message protocol," 1981.

[59] J. Olenewa and M. Ciampa, "Wireless guide to wireless communications," *Thomson Course Technology*, 2007.

[60] "ITU-T Recommendation E.800 : Terms and definitions related to quality of service and network performance including dependability,"

[61] G. Almes, S. Kalidindi, and M. Zekauskas, "RFC 2679 - A one-way delay metric for IPPM," 1999.

[62] G. Almes, S. Kalidindi, and M. Zekauskas, "RFC 2680 - A one-way packet loss metric for IPPM," 1999.

[63] C. Demichelis and P. Chimento, "RFC 3393 - IP packet delay variation metric for IP performance metrics (ippm)," 2002.

[64] "ITU-T Recommendation Y.1541 : Network performance objectives for ip-based services," 2011.

[65] "ITU-T Recommendation G.107, The e-model, a computational model for use in transmission planning," 1998.

[66] "ITU-T Recommendation 500-10: Methodology for the subjective assessment of the quality of the television pictures," 2000.

[67] Y. Wang, "Survey of objective video quality measurements," *Tech. report, Worcester Polytechnic Institute*.

[68] J. Klaue, B. Rathke, and A. Wolisz, "Evalvid - a framework for video transmission and quality evaluation," in *In Proc. of the 13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, pp. 255–272, 2003.

[69] "ITU-T Recommendation P.861 : Objective quality measurement of telephone-band (300-3400 hz) speech codecs," 1996.

[70] "ITU-T Recommendation P.862 : Perceptual evaluation of speech quality (pesq): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," 2001.

[71] "ITU-R Recommendation BS.1387 : Method for objective measurements of perceived audio quality (peaq)," 2001.

[72] "ITU-T Recommendation J.247 : Objective perceptual multimedia video quality measurement in the presence of a full reference," 2008.

[73] Z. Li, C. Wang, and R. Xu, "Computation offloading to save energy on handheld devices: a partition scheme," in *Proceedings of the 2001 international conference on Compilers, architecture, and synthesis for embedded systems*, CASES '01, (New York, NY, USA), pp. 238–246, ACM, 2001.

[74] S. Gitzenis and N. Bambos, “Joint task migration and power management in wireless computing,” *IEEE Transactions on Mobile Computing*, vol. 8, pp. 1189–1204, Sept. 2009.

[75] U. Kremer, J. Hicks, and J. M. Rehg, “A compilation framework for power and energy management on mobile computers,” in *In International Workshop on Languages and Compilers for Parallel Computing (LCPC01*, 2001.

[76] D. Huffman, “A Method for the Construction of Minimum-Redundancy Codes,” *Proceedings of the IRE*, vol. 40, pp. 1098–1101, Sept. 1952.

[77] G. G. Langdon, “Arithmetic coding,” *IBM J. Res. Develop*, vol. 23, pp. 149–162, 1979.

[78] N. Ahmed, T. Natarajan, and K. Rao, “Discrete cosine transfom,” *IEEE Trans. Comput.*, vol. 23, pp. 90–93, Jan. 1974.

[79] E. Y. Fisher, *Fractal image compression: theory and application*. Springer-Verlag, 1995.

[80] F. D. Jager, “Delta modulation-a method of PCM transmission using the one unit code,” tech. rep., Phillips Research, 1952.

[81] M. R. Schroeder and B. S. Atal, “Code-excited linear prediction(CELP): High-quality speech at very low bit rates,” in *International Conference on Acoustics, Speech, and Signal Processing*, 1985.

[82] “ITU-T Std. T.81: Information technology digital compression and coding of continuous-tone still images requirements and guidelines,” 1992.

[83] “ISO/IEC Std. 10 918-1: Information technology digital compression and coding of continuous-tone still images - requirements and guidelines,” 1994.

[84] “ISO/IEC Std.11 172: MPEG-1 coding of moving pictures and associated audio for digital storage media up to 1.5 mbits/s,” 1993.

[85] “ISO/IEC Std.13 818: MPEG-2 generic coding of moving pictures and associated audio information,” 1994.

[86] G.-M. Muntean, P. Perry, and L. Murphy, “A new adaptive multimedia streaming system for all-ip multi-service networks,” *IEEE Transactions on Broadcasting*, vol. 50, pp. 1–10, 2004.

[87] L. Guo, E. Tan, S. Chen, Z. Xiao, O. Spatscheck, and X. Zhang, “Delving into internet streaming media delivery: A quality and resource utilization perspective,” in *in Internet Measurement Conference Proceedings of the 6th ACM SIGCOMM on Internet measurement*, pp. 217–230, ACM Press, 2006.

[88] E. R. Pantos, “Http live streaming. internet-draft draft-pantoshttp-live-streaming-02,” Apple Inc., 2011.

[89] K. Choi and M. P. A. Iranli, “Energy-aware wireless video streaming,” *ESTImedia, G. Fohler and R. Marculescu, Eds*, pp. 48–55.

[90] J. Flinn and M. Satyanarayanan, “Energy-aware adaptation for mobile applications,” *SIGOPS Oper. Syst. Rev.*, vol. 33, pp. 48–63, Dec. 1999.

[91] R. Rejaie, H. Yu, M. Handley, and D. Estrin, “Multimedia proxy caching mechanism for quality adaptive streaming applications in the internet,” in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications*.

[92] S. Sen, J. Rexford, , and D. Towsley, “Proxy prefix caching for multimedia streams,” in *INFOCOM 99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*.

[93] R. Rejaie, H. Mark, H. Yu, and D. Estrin, “Proxy caching mechanism for multimedia playback streams in the internet,” in *In Proceedings of the 4th International Web Caching Workshop*, 1999.

[94] K. Wu, P. Yu, , and J. Wolf, "Segment-based proxy caching of multimedia streams," in *Proceedings of the 10th international conference on World Wide Web*, WWW '01, (New York, NY, USA), pp. 36–44, ACM, 2001.

[95] M. Burrows, D. J. Wheeler, M. Burrows, and D. J. Wheeler, "A block-sorting lossless data compression algorithm," tech. rep., HP Labs, 1994.

[96] T. Welch, "A technique for high-performance data compression," *Computer*, vol. 17, pp. 8–19, June 1984.

[97] J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 337–343, 1977.

[98] C. Timothy, J. Cleary, and I. Witten, *Text compression*. Prentice Hall, Jan. 1990.

[99] Iyer, R. Balakrishna, and D. Wilhite, "Data compression support in databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, (San Francisco, CA, USA), pp. 695–704, Morgan Kaufmann Publishers Inc., 1994.

[100] W. Ng and C. Ravishankar, "Block-oriented compression techniques for large statistical databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 9, pp. 314–328, 1997.

[101] S. Golomb, "Run-length encodings (corresp.)," *IEEE Trans. Inf. Theor.*, vol. 12, pp. 399–401, Sept. 2006.

[102] C. Kim and N. O'Connor., "Low complexity video compression using moving edge detection based on DCT coefficients," in *MMM 2009 - 15th international Multimedia Modeling Conference*, 2009.

[103] C. Cassius, "Patent 2605361 : Differential quantization of communication signals," July 1952.

[104] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," *IEEE J.Sel. A. Commun.*, vol. 15, pp. 983–1001, Sept. 2006.

[105] E. Masala, D. Quaglia, and J. C. D. Martin, "Adaptive picture slicing for distortion-based classification of video packets," in *in Proc. IEEE Workshop on Multimedia Signal Processing*, pp. 111–116, IEEE, 2001.

[106] J. Shin, J. Kim, and C.-C. J. Kuo, "Quality-of-service mapping mechanism for packet video in differentiated services network," *IEEE TRANS. MULTIMEDIA*, vol. 3, pp. 219–231, 2001.

[107] C. Im and S. Ha, "An energy optimization technique for latency and quality constrained video applications," in *Applications, First Workshop on Embedded Systems for Real-Time Multimedia (ESTIMedia03*, pp. 18–23, 2003.

[108] "H.261:Video codec for audiovisual services at p x 64 kbit/s," 1993.

[109] C. Chiasserini and E. Magli, "Energy consumption and image quality in wireless video-surveillance networks," in *In Proceedings of the 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2002.

[110] "ISO/IEC Std.14 496: Information technology coding of audio-visual objects," 2001.

[111] "ITU-T H264: Advanced video coding for generic audiovisual services," 2003.

[112] "ISO/IEC Std.15 938: Information technology multimedia content description interface," 2002.

[113] S. F. Chang, T. Sikora, and A. Puri, "Overview of the MPEG-7 standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, pp. 688–695, June 2001.

[114] C. N. Taylor, S. Dey, and D. Panigrahi, "Energy/latency/image quality tradeoffs in enabling mobile multimedia communication," in *in Enabling Mobile Multime-*

dia Communication, Software Radio: Technologies and Services 2001, pp. 55–66, Springer Verlag, 2001.

[115] P. Agrawal, J.-C. Chen, S. Kishore, P. Ramanathan, and K. Sivalingam, "Battery power sensitive video processing in wireless networking," *In Proceedings of IEEE PIMRC*, 1998.

[116] R. Luigi and V. Lorenzo, "Replacement policies for a proxy cache," *IEEE/ACM Trans. Netw.*, vol. 8, pp. 158–170, Apr. 2000.

[117] S. Michel, K. Nguyen, S. F. A. Rosenstein, L. Zhang, and V. Jacobson, "Adaptive web caching: towards a new global caching architecture," *Comput. Netw. ISDN Syst.*, vol. 30, pp. 2169–2177, Nov. 1998.

[118] S. Chen, H. Wang, X. Zhang, B. Shen, and S. Wee, "Segment-based proxy caching for internet streaming media delivery," *IEEE MultiMedia*, vol. 12, pp. 59–67, July 2005.

[119] Y. Chae, K. Guo, M. Buddhikot, S. Suri, and E. Zegura, "Silo, rainbow, and caching token: schemes for scalable, fault tolerant stream caching," *IEEE J.Sel. A. Commun.*, vol. 20, pp. 1328–1344, Sept. 2006.

[120] M. Allman, V. Paxson, and M. Stevens, "RFC 2581 - TCP Congestion Control," 1999.

[121] V. Jacobson, "Congestion avoidance and control," in *Symposium proceedings on Communications architectures and protocols*, SIGCOMM '88, (New York, NY, USA), pp. 314–329, ACM, 1988.

[122] V. Jacobson, "Modified TCP congestion avoidance algorithm," in *end2end-interest mailing list*, 1990.

[123] S. Floyd and T. Henderson, "RFC2582: The newreno modification to TCP's fast recovery algorithm," 1999.

[124] M. Mathis, J. Mahdavi, S. Floyd, , and A. Romanow, “RFC 2018 - TCP Selective Acknowledgment Options,” 1996.

[125] K. Fall, Kevin, and S. Floyd, “Simulation-based comparisons of Tahoe, Reno and SACK TCP,” *SIGCOMM Comput. Commun. Rev.*, vol. 26, pp. 5–21, July 1996.

[126] S. Alaa, G. Yacine, and S. Sidi-Mohammed, “A performance study of TCP variants in terms of energy consumption and average goodput within a static ad hoc environment,” in *Proceedings of the 2006 international conference on Wireless communications and mobile computing*, IWCMC ’06, (New York, NY, USA), pp. 503–508, ACM, 2006.

[127] S. Lawrence, W. S. W, , and L. Peterson, “TCP Vegas: new techniques for congestion detection and avoidance,” in *Proceedings of the conference on Communications architectures, protocols and applications*, SIGCOMM ’94, (New York, NY, USA), pp. 24–35, ACM, 1994.

[128] S. Mascolo, C. Casetti, M. Gerla, M. Sanadidi, and R. Wang, “TCP westwood: Bandwidth estimation for enhanced transport over wireless links,” in *Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom ’01, (New York, NY, USA), pp. 287–297, ACM, 2001.

[129] H. Zhang, A. Arora, Y. Choi, and G. Gouda, “Reliable bursty convergecast in wireless sensor networks,” in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc ’05, (New York, NY, USA), pp. 266–276, ACM, 2005.

[130] C. Wan, “PSFQ: A reliable transport protocol for wireless sensor networks,” in *in Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 1–11, 2002.

[131] S. Park, R. Vedantham, R. Sivakumar, and I. Akyildiz, “A scalable approach for reliable downstream data delivery in wireless sensor networks,” in *Proceedings of*

the 5th ACM international symposium on Mobile ad hoc networking and computing, MobiHoc '04, (New York, NY, USA), pp. 78–89, ACM, 2004.

[132] B. Fang, R. Sumit, and R. Govindan, "Quasi-static centralized rate allocation for sensor networks.," in *SECON*, pp. 361–370, 2007.

[133] C. Wan, S. Eisenman, , and A. Campbell, "CODA: congestion detection and avoidance in sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, pp. 266–279, 2003.

[134] V. Tsaoussidis and H. Badr, "TCP-probing: towards an error control schema with energy and throughput performance gains," in *Proceedings of the 2000 International Conference on Network Protocols*, 2000.

[135] C. Zhang and V. Tsaoussidis, "TCP-real: improving real-time capabilities of TCP over heterogeneous networks," in *Proceedings of the 11th international workshop on Network and operating systems support for digital audio and video*, pp. 189–198, 2001.

[136] S. Rangwala, R. Gummadi, R. Govindan, and K. Psounis, "Interference-aware fair rate control in wireless sensor networks," in *In Proceedings of the ACM SIGCOMM*, pp. 63–74, 2006.

[137] Y. G. Iyer, "STCP: A generic transport layer protocol for wireless sensor networks," in *Proc. of IEEE Intl. Conf. on Computer Communications and Networks (ICCCN)*, pp. 449–454, 2005.

[138] Y. Sankarasubramaniam, O. B. Akan, and I. F. Akyildiz, "ESRT: event-to-sink reliable transport in wireless sensor networks," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking and computing*, pp. 177–188, 2003.

[139] J. Paek and R. Govindan, "RCRT: Rate-controlled reliable transport protocol for wireless sensor networks," *ACM Trans. Sen. Netw.*, vol. 7, pp. 20:1–20:45, 2010.

[140] N. Tezcan and W. Wang, "ART: an asymmetric and reliable transport mechanism for wireless sensor networks," *Int. J. Sen. Netw.*, vol. 2, pp. 188–200, 2007.

[141] C. Wang, K. Sohraby, and B. Li, "SenTCP: A Hop-by-Hop Congestion Control Protocol for Wireless Sensor Networks," in *IEEE INFOCOM*, 2005.

[142] C. Wang, K. Sohraby, V. Lawrence, B. Li, and Y. Hu, "Priority-based congestion control in wireless sensor networks," in *Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, pp. 22–31, 2006.

[143] B. Hull, K. Jamieson, and H. Balakrishnan, "Mitigating congestion in wireless sensor networks," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pp. 134–147, 2004.

[144] A. Woo and D. Culler, "A transmission control scheme for media access in sensor networks," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, pp. 221–235, 2001.

[145] R. Rajamani, S. Kumar, and N. Gupta, "SCTP versus TCP: comparing the performance of transport protocols for web traffic," 2002.

[146] S. Singh, M. Woo, and C. Raghavendra, "Power-aware routing in mobile ad hoc networks," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, pp. 181–190, 1998.

[147] C. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," vol. 39, no. 6, pp. 138–147, 2001.

[148] B. Chen, K. Jamieson, R. Morris, and H. Balakrishnan, "Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," in *Proceeding of the 7th annual ACMinternational conference on Mobile computing and networking*, 2001.

[149] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 8 - Volume 8*, 2000.

[150] S. Lindsey, C. Raghavendra, and K. Sivalingam, "Data gathering algorithms in sensor networks using energy metrics," vol. 13, no. 9, pp. 924–935, 2002.

[151] S. Lindsey, S., and C. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," in *Proc. IEEE Aerospace*, vol. 3, pp. 3–1125, 2002.

[152] S. Ito and K. Yoshigoe, "Consumed-energy-type-aware routing for wireless sensor networks," in *Proc. Wireless Telecommunications Symp. WTS 2007*, pp. 1–6, 2007.

[153] Y. Yu, R. Govindan, and D. Estrin, "Geographical and energy aware routing: a recursive data dissemination protocol for wireless sensor networks," *UCLA Computer Science Department Technical Report*, 2001.

[154] V. Rodoplu and R. H. Meng, "Minimum energy mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, 1998.

[155] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, 2001.

[156] H. Zhang and H. Shen, "Energy-efficient beaconless geographic routing in wireless sensor networks," vol. 21, no. 6, pp. 881–896, 2010.

[157] K.Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie, "Protocols for self-organization of a wireless sensor network," *IEEE Personal Communications*, vol. 7, no. 5, pp. 16–27, 2000.

[158] O. Chipara, Z. He, G. Xing, X. W. Q. Chen and, C. Lu, J. Stankovic, and T. Abdelzaher, "Real-time power-aware routing in sensor networks," in *Proc. 14th IEEE Int. Workshop Quality of Service IWQoS 2006*, pp. 83–92, 2006.

[159] M. EffatParvar, A.Dareshorzadeh, M. Dehghan, , and M. EffatParvar, "Quality of service support and local recovery for ODMRP multicast routing in ad hoc networks," in *Proc. 4th Int. Conf. Innovations in Information Technology IIT '07*, pp. 695–699, 2007.

[160] T. Taleb, K. Kashibuchi, A. Leonardi, S. Palazzo, K. Hashimoto, N. Kato, and Y. Nemoto, "A cross-layer approach for an efficient delivery of TCP/RTP-based multimedia applications in heterogeneous wireless networks," vol. 57, no. 6, pp. 3801–3814, 2008.

[161] N. Ouferhat and A. Mellouk, "Energy and delay efficient state dependent routing algorithm in wireless sensor networks," in *Proc. IEEE 34th Conf. Local Computer Networks LCN 2009*, pp. 1069–1076, 2009.

[162] C. Shanti and A. Sahoo, "DGRAM: A delay guaranteed routing and MAC protocol for wireless sensor networks," vol. 9, no. 10, pp. 1407–1423, 2010.

[163] M. Tariq, Y. Kim, J. Kim, Y. Park, and E. Jung, "Energy efficient and reliable routing scheme for wireless sensor networks," in *Proc. Int. Conf. Communication Software and Networks ICCSN '09*, pp. 181–185, 2009.

[164] W. B. Heinzelman, W.Rabiner, K. Joanna, and H.Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pp. 174–185, 1999.

[165] F. Silva, J. Heidemann, R. Govindan, and D. Estrin, *Frontiers in Distributed Sensor Networks*, ch. Directed Diffusion. Boca Raton, Florida, USA: CRC Press, Inc., October 2003.

[166] H. Miura and M. Yamamoto, "Content routing with network support using passive measurement in content distribution networks," in *Proc. Eleventh Int Computer Communications and Networks Conf*, pp. 96–101, 2002.

[167] J. Ni, D. Tsang, I. Yeung, and X. Hei, "Hierarchical content routing in large-scale multimedia content delivery network," in *Proc. IEEE Int. Conf. Communications ICC '03*, vol. 2, pp. 854–859, 2003.

[168] A. Mitra, M. Maheswaran, and J. Rueda, "Wide-area content-based routing mechanism," in *Proc. Int. Parallel and Distributed Processing Symp*, 2003.

[169] Y. Liu, N. Xiong, Y. Zhao, A. Vasilakos, J. Gao, and J. Jia, "Multi-layer clustering routing algorithm for wireless vehicular sensor networks," 2010. Communications, IET.

[170] R. Ding and G.-M. Muntean, "An energy-oriented node characteristics-aware routing algorithm for wireless LAN," in *IEEE Int Broadband Multimedia Systems and Broadcasting (BMSB) Symp*, 2011.

[171] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 12, pp. 493–506, June 2004.

[172] V. Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, pp. 171–180, 2003.

[173] Y. Wei, S. Chandra, and S. Bhandarkar, "A statistical prediction-based scheme for energy-aware multimedia data streaming," in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, pp. 2053–2057, 2004.

[174] M. Stemm, P. Gauthier, D. Harada, and Y. H. Katz, "Reducing power consumption of network interfaces in hand-held devices (extended abstract)," 1996.

[175] A. El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks," in *Communications, 2002. ICC 2002. IEEE International Conference on*, vol. 5, pp. 3418 – 3423 vol.5, 2002.

[176] J. Polastre, J. Hill, and D. Culler, “Versatile low power media access for wireless sensor networks,” in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pp. 95–107, 2004.

[177] S. Singh and C. Raghavendra, “PAMAS: Power aware multi-access protocol with signalling for ad hoc networks,” 1999.

[178] V. Rajendran, K. Obraczka, and J. Garcia-Luna-Aceves, “Energy-efficient collision-free medium access control for wireless sensor networks,” in *Proceedings of the 1st international conference on Embedded networked sensor systems*, pp. 181–192, 2003.

[179] L. van Hoesel and P. Havinga, “A lightweight medium access protocol (lmac) for wireless sensor networks: Reducing preamble transmissions and transceiver state switches,” in *1st International Workshop on Networked Sensing Systems (INSS*, pp. 205–208, Society of Instrument and Control Engineers (SICE), 2004.

[180] G. Zhou, C. Huang, T. Yan, T. He, J. Stankovic., and T. F. Abdelzaher, “MMSN: Multi-frequency media access control for wireless sensor networks,” in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pp. 1 –13, april 2006.

[181] Z. Zhou, S. Le, and C. Jun-Hong, “An OFDM based MAC protocol for underwater acoustic networks,” in *Proceedings of the Fifth ACM International Workshop on UnderWater Networks*, WUWNet ’10, (New York, NY, USA), pp. 6:1–6:8, ACM, 2010.

[182] B. Yahya and J. Ben-Othman, “An energy efficient hybrid medium access control scheme for wireless sensor networks with quality of service guarantees,” in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, pp. 1 –5, 30 2008-dec. 4 2008.

[183] M. Adamou, I. Lee, and I. Shin, "An energy efficient real-time medium access control protocol for wireless ad-hoc networks," in *Proceedings of 22nd IEEE Real-Time Systems Symposium (RTSS 2001)*, RTSS 2001, (London, UK), 2001.

[184] A. A. Syed, W. Ye, and J. S. Heidemann, "Comparison and evaluation of the T-Lohi MAC for underwater acoustic sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 9, pp. 1731–1743, 2008.

[185] C. Huan and C. Huang, "Power management modeling and optimal policy for IEEE 802.11 WLAN systems," in *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, vol. 6, pp. 4416 – 4421 Vol. 6, sept. 2004.

[186] J. Miller and N. Vaidya, "A MAC protocol to reduce sensor network energy consumption using a wakeup radio," *IEEE Transactions on Mobile Computing*, vol. 4, pp. 228–242, 2005.

[187] P. J. Havinga and G. J. Smit, "Energy-efficient TDMA medium access control protocol scheduling," in *Asian International Mobile Computing Conference, AMOC*, pp. 1–10, 2000.

[188] J. Zhang, G. Zhou, C. Huang, S. Son, and J. Stankovic, "TMMAC: An energy efficient multi-channel MAC protocol for ad hoc networks," in *Communications, 2007. ICC '07. IEEE International Conference on*, pp. 3554 –3561, june 2007.

[189] X. Chen, P. Han, Q. He, S. Tu, and Z. Chen, "A multi-channel MAC protocol for wireless sensor networks," in *Computer and Information Technology, 2006. CIT '06. The Sixth IEEE International Conference on*, p. 224, sept. 2006.

[190] M. Salajegheh, H. Soroush, and A. Kalis, "HYMAC: Hybrid TDMA/FDMA medium access control protocol for wireless sensor networks," in *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1 –5, 2007.

[191] S. Tao and K. Marwan, “Energy-efficient power/rate control and scheduling in hybrid tdma/cdma wireless sensor networks,” *Comput. Netw.*, vol. 53, pp. 1395–1408, 2009.

[192] J. Zhang, T. Wang, and A. N., “An efficient MAC protocol for asynchronous ONUs in OFDMA PONs,” in *Optical Fiber Communication Conference and Exposition (OFC/NFOEC), 2011 and the National Fiber Optic Engineers Conference*, pp. 1 –3, march 2011.

[193] W. Wang, H. Wang, D. Peng, and H. Sharif, “An energy efficient pre-schedule scheme for hybrid CSMA/TDMA MAC in wireless sensor networks,” in *10th IEEE Singapore International Conference on Communication systems*, pp. 1 –5, oct. 2006.

[194] A. El-Hoiydi, “Spatial TDMA and CSMA with preamble sampling for low power ad hoc wireless sensor networks,” in *Proceedings of the Seventh International Symposium on Computers and Communications (ISCC’02)*, 2002.

[195] I. Chlamtac, “Mobile ad hoc networking: imperatives and challenges,” *Ad Hoc Networks*, vol. 1, no. 1, pp. 13–64, 2003.

[196] W. Stark, H. Wang, A. Worthen, S. Lafortune, and D. Teneketzis, “Low-energy wireless communication network design,” *Wireless Communications, IEEE*, vol. 9, pp. 60 – 72, aug. 2002.

[197] V.Kawadia and P. Kumar, “A cautionary perspective on cross-layer design,” *Wireless Communications, IEEE [see also IEEE Personal Communications]*, vol. 12, no. 1, pp. 3–11, 2005.

[198] S. Khan, Y. Peng, E. Steinbach, M. Sgroi, and W. Kellerer, “Application-driven cross-layer optimization for video streaming over wireless networks,” *Communications Magazine, IEEE*, vol. 44, pp. 122 – 130, jan. 2006.

[199] C. L.-U. Choi, W. Kellerer, and E. Steinbach, “Cross layer optimization for wireless multi-user video streaming,” in *Image Processing, 2004. ICIP ’04. 2004 International Conference on*, vol. 3, pp. 2047 – 2050 Vol. 3, oct. 2004.

[200] M. Schaar, S. Krishnamachari, C. Sunghyun, and X. Xiaofeng, “Adaptive cross-layer protection strategies for robust scalable video transmission over 802.11 WLANs,” *Selected Areas in Communications, IEEE Journal on*, vol. 21, pp. 1752 – 1763, dec. 2003.

[201] F. Liu, C. Tsui, and Y. Zhang, “Joint routing and sleep scheduling for lifetime maximization of wireless sensor networks,” vol. 9, no. 7, pp. 2258–2267, 2010.

[202] M. Rosu, C. Olsen, C. Narayanaswami, and L. Luo, “PAWP: a power aware web proxy for wireless LAN clients,” in *Sixth IEEE Workshop on Mobile Computing Systems and Applications*, pp. 206 – 215, 2004.

[203] S. Chandra and M. Vahdat, “Application-specific network management for energy-aware streaming of popular multimedia formats,” in *Proceedings of the General Track of the annual conference on USENIX Annual Technical Conference*, pp. 329–342, 2002.

[204] M. Hoque, M. Siekkinen, and J. Nurminen, “On the energy efficiency of proxy-based traffic shaping for mobile audio streaming,” in *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*, pp. 891–895, 2011.

[205] S. Ergen and P.Varaiya, “PEDAMACS: power efficient and delay aware medium access protocol for sensor networks,” *Mobile Computing, IEEE Transactions on*, vol. 5, pp. 920 – 930, july 2006.

[206] G. Lu, B. Krishnamachari, and C. Raghavendra, “An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks,” in *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, p. 224, april 2004.

[207] J. Chen, T. Lv, and H. Zheng, “Cross-layer design for QoS wireless communications,” in *Circuits and Systems, 2004. ISCAS ’04. Proceedings of the 2004 International Symposium on*, vol. 2, pp. II – 217–20 Vol.2, may 2004.

[208] Y. Liu, I. Elhanany, and Q. Hairong, "An energy-efficient QoS-aware media access control protocol for wireless sensor networks," in *Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on*, pp. 3 pp. –191, nov. 2005.

[209] Q. Liu, S. Zhou, and G. Giannakis, "Cross-layer scheduling with prescribed QoS guarantees in adaptive wireless networks," *Selected Areas in Communications, IEEE Journal on*, vol. 23, pp. 1056 – 1066, may 2005.

[210] R. Kravets and P. Krishnan, "Application-driven power management for mobile communication," *Wirel. Netw.*, vol. 6, pp. 263–277, July 2000.

[211] M. Alicherry, R. Bhatia, and L. L. (Erran), "Finding disjoint paths with related path costs," *Journal of Combinatorial Optimization*, no. 12, pp. 83–96, 2006.

[212] C. Chiasserini and R. Rao, "A model for battery pulsed discharge with recovery effect," in *Wireless Communications and Networking Conference, 1999. WCNC. 1999 IEEE*, pp. 636 –639 vol.2, 1999.

[213] C. Chiasserini and R. Rao, "Improving battery performance by using traffic shaping techniques," *Selected Areas in Communications, IEEE Journal on*, vol. 19, pp. 1385 –1394, jul 2001.

[214] C. Chiasserini and R. Rao, "Pulsed battery discharge in communication devices," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, MobiCom '99, (New York, NY, USA), pp. 88–95, ACM, 1999.

[215] F. R. Dogar and P. Steenkiste, "Catnap: Exploiting high bandwidth wireless interfaces to save energy for mobile devices," in *Proc. Int. Conf. Mobile Systems, Applications and Services (MobiSys)*, 2010.

[216] M. C. Rosu, C. M. Olsen, L. Luo, and C. Narayanaswami, "The power-aware streaming proxy architecture," 2005.

[217] P. Shenoy and P. Radkov, "Proxy-assisted power-friendly streaming to mobile devices," in *MMCN*, pp. 177–191, 2003.

[218] M. Gundlach, S. Doster, H. Yan, D. Lowenthal, S. Watterson, and S. Chandra, "Dynamic, power-aware scheduling for mobile clients using a transparent proxy," in *Parallel Processing, 2004. ICPP 2004. International Conference on*, pp. 557 – 565 vol.1, aug. 2004.

[219] S. A. Akella, R. K. Balan, and N. Bansal, "Protocols for low-power," tech. rep., Carnegie-Mellon University, 2001.

[220] P. Suriyachai, U. Roedig, and A. Scott, "Implementation of a MAC protocol for QoS support in wireless sensor networks," in *Pervasive Computing and Communications, 2009. PerCom 2009. IEEE International Conference on*, pp. 1 –6, march 2009.

[221] M. Sichitiu, "Cross-layer scheduling for power efficiency in wireless sensor networks," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 3, pp. 1740 –1750 vol.3, march 2004.

[222] Y. Wu, X. Li, Y. Liu, and W. Lou, "Energy-efficient wake-up scheduling for data collection and aggregation," vol. 21, no. 2, pp. 275–287, 2010.

[223] A. Bernardos, P. Tarrio, and J. Casar, "An energy aware routing algorithm for ad hoc and sensor networks: Concept and performance," in *Proc. IEEE 17th Int Personal, Indoor and Mobile Radio Communications Symp*, pp. 1–5, 2006.

[224] N. Saxena, A. Roy, and J. Shin, "Dynamic duty cycle and adaptive contention window based QoS-MAC protocol for wireless multimedia sensor networks," *Comput. Netw.*, vol. 52, pp. 2532–2542, September 2008.

[225] S. T. Raja and J. Zhang, "Opportunistic scheduling for streaming video in wireless networks," in *Hopkins University*, 2003.

[226] H. Jiang and C. Dovrolis, "Source-level ip packet bursts: Causes and effects," in *In Proc. IMC*, pp. 301–306, 2003.

[227] B. Vandalore, V. Bobby, S. Kalyanaraman, R. Jain, R. Goyal, S. Fahmy, and S. Kim, "Performance of bursty world wide web (www) sources over abr," in *WebNet*, (Toronto), 1997.

[228] J. Touch, J. Heidemann, and A. Hughes, "Issues in TCP slow-start restart after idle," in *IETF Internet Draft*, Mar. 1998.

[229] S. Chandra, "Wireless network interface energy consumption: implications for popular streaming formats," *Multimedia Syst.*, vol. 9, pp. 185–201, Aug. 2003.

[230] H. Yan, S. Watterson, D. Lowenthal, K. Li, R. Krishnan, and L. Peterson, "Client-centered, energy-efficient wireless communication on ieee 802.11b networks," *Mobile Computing, IEEE Transactions on*, vol. 5, no. 11, pp. 1575–1590, 2006.

[231] R. Krashinsky and H. Balakrishnan, "Minimizing energy for wireless web access with bounded slowdown," *Wirel. Netw.*, vol. 11, pp. 135–148, Jan. 2005.

[232] C. E. Jones, K. Sivalingam, P. Agrawal, and J. Chen, "A survey of energy efficient network protocols for wireless networks," *Wirel. Netw.*, vol. 7, pp. 343–358, Sept. 2001.

[233] N. Lago and F. Kon, "The quest for low latency," in *PROCEEDINGS OF THE INTERNATIONAL COMPUTER MUSIC CONFERENCE (ICMC2004*, pp. 33–36, 2004.

[234] S. Lee, G.-M. Muntean, and A. Smeaton, "Performance-aware replication of distributed pre-recorded IPTV content," *Broadcasting, IEEE Transactions on*, vol. 55, pp. 516 –526, Jun. 2009.

[235] R. Cole and J. Rosenbluth, "Voice over IP performance monitoring," *SIGCOMM Comput. Commun. Rev.*, vol. 31, pp. 9–24, Apr. 2001.

[236] J. D. Hamilton, *Time Series Analysis*. NJ: Princeton University Press, 1994.

[237] R. Harris and R. Sollis, *Applied time series modelling and forecasting*. Wiley, 2003.

[238] “ ITU-R BT.500-11: subjective television picture assessment,” 2002.