Institutionen för systemteknik
Department of Electrical Engineering

Examensarbete

An adaptive solution for power efficiency and QoS optimization in WLAN 802.11n

Master thesis performed at *NEC Laboratories Europe*

by

Manil Dev Gomony

LiTH-ISY-EX--10/4276--SE
Linköping 2010

TEKNISKA HÖGSKOLAN
LINKÖPINGS UNIVERSITET

Department of Electrical Engineering
Linköping University
S-581 83 Linköping, Sweden

Linköpings tekniska högskola
Institutionen för systemteknik
581 83 Linköping

An adaptive solution for power efficiency and QoS optimization in WLAN 802.11n

Master thesis in Communication Systems
at Linköping Institute of Technology
by

Manil Dev Gomony

LiTH-ISY-EX--10/4276--SE

Supervisors: **Xavier Pérez Costa**
NEC Laboratories Europe

Daniel Camps Mur
NEC Laboratories Europe

Examiner: **Danyo Danev**
ISY, Linköpings Universitet

Linköping, March 2010

Avdelning, institution
Division, department

Institutionen för systemteknik
Department of Electrical Engineering

Publishing Date:
(Electronic version)

Presentation Date: 25th March 2010

Linköpings Universitet

Språk
Language

- [ ] Svenska/Swedish
- [x] Engelska/English
- [ ] ____________

Rapporttyp
Report category

- [ ] Licentiate thesis
- [x] Degree thesis
- [ ] Thesis C-level
- [ ] Thesis D-level
- [ ] Report
- ____________

ISBN —

ISRN LiTH-ISY-EX--10/4276--SE

Serietitel och serienummer
Title of series, numbering

ISSN —

URL för elektronisk version
URL, Electronic Version

http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-54764

Titel
Title

An adaptive solution for power efficiency and QoS optimization in WLAN 802.11n

Förf**attare**
Author

Manil Dev Gomony

Sammanfattning
Abstract

The wide spread use of IEEE Wireless LAN 802.11 in battery operated mobile devices introduced the need of power consumption optimization while meeting Quality-of-Service (QoS) requirements of applications connected through the wireless network. The IEEE 802.11 standard specifies a baseline power saving mechanism, hereafter referred to as standard Power Save Mode (PSM), and the IEEE 802.11e standard specifies the Automatic Power Save Delivery (APSD) enhancement which provides support for real-time applications with QoS requirements. The latest amendment to the WLAN 802.11 standard is the IEEE 802.11n standard which enables the use of much higher data rates by including enhancements in the Physical and MAC Layer. In this thesis, different 802.11n MAC power saving and QoS optimization possibilities are analyzed comparing against existing power saving mechanisms.

Initially, the performance of the existing power saving mechanisms PSM and Unscheduled-APSD (UAPSD) are evaluated using the 802.11n process model in the OPNET simulator and the impact of frame aggregation feature introduced in the MAC layer of 802.11n was analyzed on these power saving mechanisms. From the performance analysis it can be concluded that the frame aggregation will be efficient under congested network conditions. When the network congestion level increases, the signaling load in UAPSD saturates the channel capacity and hence results in poor performance compared to PSM. Since PSM cannot guarantee the minimum QoS requirements for delay sensitive applications, a better mechanism for performance enhancement of UAPSD under dynamic network conditions is proposed.

The functionality and performance of the proposed algorithm is evaluated under different network conditions and using different contention settings. From the performance results it can be concluded that, by using the proposed algorithm the congestion level in the network is reduced dynamically thereby providing a better power saving and QoS by utilizing the frame aggregation feature efficiently.

Nyckelord
Keywords

WLAN, 802.11n, Frame Aggregation, Power Efficiency, QoS

Abstract

The wide spread use of IEEE Wireless LAN 802.11 in battery operated mobile devices introduced the need of power consumption optimization while meeting Quality-of-Service (QoS) requirements of applications connected through the wireless network. The IEEE 802.11 standard specifies a baseline power saving mechanism, hereafter referred to as standard Power Save Mode (PSM), and the IEEE 802.11e standard specifies the Automatic Power Save Delivery (APSD) enhancement which provides support for real-time applications with QoS requirements. The latest amendment to the WLAN 802.11 standard is the IEEE 802.11n standard which enables the use of much higher data rates by including enhancements in the Physical and MAC Layer. In this thesis, different 802.11n MAC power saving and QoS optimization possibilities are analyzed comparing against existing power saving mechanisms.

Initially, the performance of the existing power saving mechanisms PSM and Unscheduled-APSD (UAPSD) are evaluated using the 802.11n process model in the OPNET simulator and the impact of frame aggregation feature introduced in the MAC layer of 802.11n was analyzed on these power saving mechanisms. From the performance analysis it can be concluded that the frame aggregation will be efficient under congested network conditions. When the network congestion level increases, the signaling load in UAPSD saturates the channel capacity and hence results in poor performance compared to PSM. Since PSM cannot guarantee the minimum QoS requirements for delay sensitive applications, a better mechanism for performance enhancement of UAPSD under dynamic network conditions is proposed.

The functionality and performance of the proposed algorithm is evaluated under different network conditions and using different contention settings. From the performance results it can be concluded that, by using the proposed algorithm the congestion level in the network is reduced dynamically thereby providing a better power saving and QoS by utilizing the frame aggregation feature efficiently.

Acknowledgments

First of all I would like to thank my Supervisors at NEC Xavier Pérez Costa and Daniel Camps Mur for their constant support during the entire thesis work and for the friendly relationship we had during my stay at NEC. Especially I thank Daniel for his constant technical guidance throughout the project and for our fruitful discussions which made me achieve good results in the end. Also, thanks to all my colleagues and friends at NEC for their support and making my stay at NEC a memorable one.

I would like to thank my thesis Examiner at Linköping University Danyo Danev for spending his valuable time towards verifying my thesis report. I would also like to thank my thesis opponent Prasanna Shanmuga Sundaram for his time and efforts in reviewing my thesis report.

Finally I would like to thank my family and friends for their constant love and support.

Contents

1. Introduction

1.1. Background

During the recent years, the use of Wireless Local Area Network (WLAN) in mobile terminals has increased rapidly primarily due to the advantages that it possess, such as interoperability, mobility and flexibility that enables faster access to the Internet services. However, battery operated devices have to operate with the minimum power consumption and at the same time meet the Quality-of-Service (QoS) requirements of the applications connected through WLAN. The IEEE 802.11 standard specifies a Power Save Mode (PSM) that can be used in such devices to minimize power consumption and IEEE 802.11e specifies Automatic Power Save Delivery (APSD) which is an extension to the PSM to guarantee the QoS requirements of applications in addition to the power saving.

The latest amendment to the WLAN 802.11 standard is the IEEE 802.11n that has enhancements both in the *Physical (PHY)* layer and in the *Medium Access Control (MAC)* layer providing data rates beyond 200Mbps using the Physical throughput of 600Mbps [3].
The enhancement in the MAC Layer of 802.11n includes frame aggregation in which multiple MAC frames are sent as one PHY layer packet to reduce overhead and this could have an impact on the performance of the power saving mechanisms. Hence it is required to study the performance of the power saving mechanisms in the 802.11n standard and explore more possibilities to further optimize the power saving and QoS.

This thesis work is performed at NEC Network Laboratories as a part of the research project for the configuration of the IEEE 802.11n standard into the NEC mobile terminals.

1.2. Thesis Scope

- Study of the IEEE 802.11 WLAN standard

- Study of the MAC layer enhancements introduced in the IEEE 802.11n amendment

- Study of the power saving mechanisms – Standard Power Save Mode (PSM) and Automatic Power Save Delivery (APSD)

- Implementation of PSM and APSD in 802.11n OPNET process model

- Performance evaluation of PSM and APSD in the 802.11n with MAC layer enhancements

- Propose possible enhancements for power saving and QoS optimization in 802.11n

1.3. Thesis Goal

- To evaluate the performance of the power saving mechanisms - Standard Power Save Mode (PSM) and Automatic Power Save Delivery (APSD) on the IEEE 802.11n standard and to study the impact of the MAC layer enhancements introduced in the 11n standard on the power saving mechanisms.

- From the analysis of the performance results, propose possible enhancements to further optimize power savings and Quality-of-Service (QoS) utilizing the MAC layer enhancements in 802.11n.

1.4. Method

- Initially, literature study was done to understand the WLAN 802.11 standard and the power saving mechanisms – Power Save Mode (PSM) and the Automatic Power Save Delivery (APSD).

- The power saving mechanisms is then implemented in the 802.11n OPNET process model and the performance evaluation is done by performing simulations in OPNET network simulator.

- The statistics to measure the power consumption and QoS are collected during the simulations and the results are processed and plotted using Matlab.

- The proposed enhancement is then implemented in the OPNET process model and its functionality is verified by performing simulations.

- The statistics are again collected by performing simulations with the proposed enhancement and the performance results are then compared with the results from the first experiment to evaluate the gain achieved in power saving and QoS by using the enhancement.

1.5. Structure of the Thesis

Chapter 2 gives a brief introduction to the theoretical background of WLAN

Chapter 3 describes the power saving mechanisms PSM and APSD

Chapter 4 gives a brief introduction to the OPNET simulator and the simulation environment

Chapter 5 explains the performance results of the power saving mechanisms in 802.11n

Chapter 6 describes the proposed enhancements and its performance results

Chapter 7 concludes the thesis work with future work

1.6. Abbreviations

AC	Access Category
AIFS	Arbitration Inter-Frame Space
AIFSN	Arbitration Inter-Frame Space Number
APSD	Automatic Power Save Delivery
AP	Access Point
BAR	Block Acknowledgment Request
BSS	Basic Service Set
CCA	Clear Channel Assessment
CFP	Contention Free Period
CP	Contention Period
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
DCF	Distributed Coordination Function
DIFS	DCF Inter-Frame Space
DSSS	Direct Sequence Spread Spectrum
EDCA	Enhanced Distributed Channel Access
FHSS	Frequency Hopping Spread Spectrum
HCCA	HCF Controlled Channel Access
HCF	Hybrid Coordination Function
IBSS	Independent Basic Service Set
IEEE	Institute of Electrical and Electronics Engineers
IR	Infrared
MAC	Medium Access Control
MPDU	MAC Protocol Data Unit
MSDU	MAC Protocol Service Unit
NAV	Network Allocation Vector

OSI	Open System Interconnection
PC	Point Coordinator
PCF	Point Coordination Function
PHY	Physical Layer
PSM	Power Save Mode
QoS	Quality of Service
SAPSD	Scheduled Automatic Power Save Delivery
SME	Station Management Entity
TBTT	Targeted Beacon Transmission Time
TCP	Transmission Control Protocol
TSF	Timing Synchronization Function
TXOP	Transmission Opportunity
UAPSD	Unscheduled Automatic Power Save Delivery
UP	User Priority
VoIP	Voice over Internet Protocol
WLAN	Wireless Local Area Network
WM	Wireless Medium

2. Theoretical Background

2.1. Wireless LAN

A Wireless LAN is a *Wireless Local Area Network (WLAN)* that interconnects two or more wireless terminals or stations each other through a radio link so that it gives the users mobility to move around within a broad coverage area still being connected to the network. In 1997, the *Institute of Electrical and Electronics Engineers (IEEE)* adopted the first Wireless LAN standard IEEE 802.11-1997 which defines the two lowest layers *Physical (PHY)* layer and *Media Access Control (MAC)* layer of the *Open System Interconnection (OSI)* Reference Model [1] as shown in Figure [2.1]. The *Logical Link Control (LLC)* Layer is common for all 802 families.

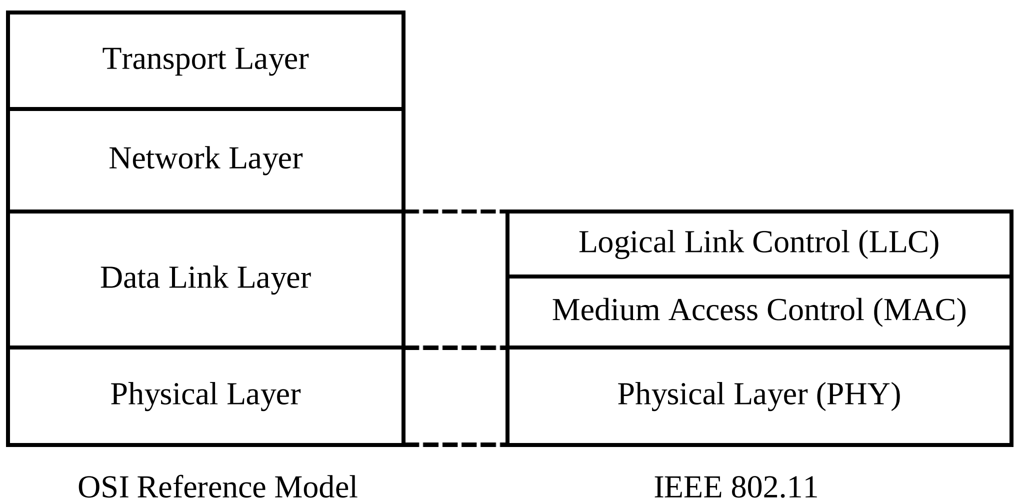

Figure [2.1]: IEEE 802.11 standard mapped to the OSI Reference Model

2.2. WLAN Architecture

The Wireless LAN station is the basic unit in a Wireless LAN network that contains the functionality of the 802.11 protocol in the Physical layer and in the MAC layer. The basic building block of the Wireless LAN is the *Basic Service Set (BSS)* which consists of two or more stations that can communicate with each other. The Figure [2.2] below shows two BSS's, each of them has two stations that are members of the BSS [1].

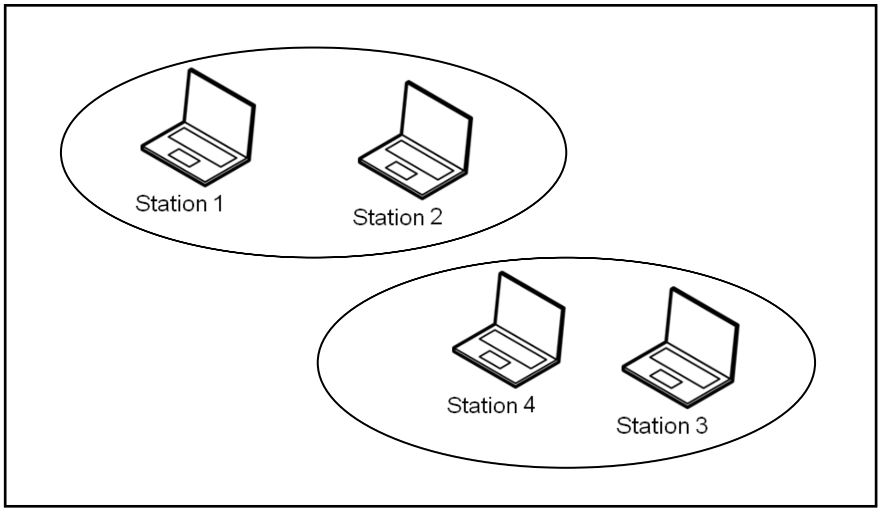

Figure [2.2]: Basic Service Set

The most basic type of IEEE 802.11 LAN topology in which two or more stations are connected via the wireless media is called the *Independent Basic Service Set (IBSS)*. The set of stations in an IBSS communicate directly with each other as shown in Figure [2.3]. This type of network is often referred to as an *ad hoc* network.

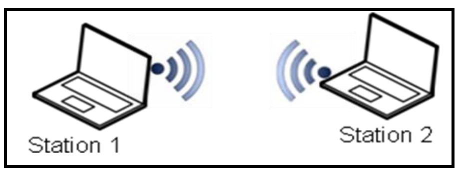

Figure [2.3]: Ad-hoc Network

In a Distribution System, an *Access Point* (*AP)* provides access to distribution services via the *Wireless Medium (WM)* to the associated stations. The *association* process is performed to establish a logical connection between the AP and the station before they can start exchanging the frames. The AP communicates with other AP's to exchange data with the Stations in their respective BSS's as shown in Figure [2.4]. The AP's may be connected by a wired network or a wireless network that exchanges information between each other. A BSS which includes an AP is called an *Infrastructure BSS*. All stations within the Infrastructure Basic Service Set communicate via the AP and the frames to the stations are relayed by the Access Point. In this thesis, we consider an Infrastructure BSS.

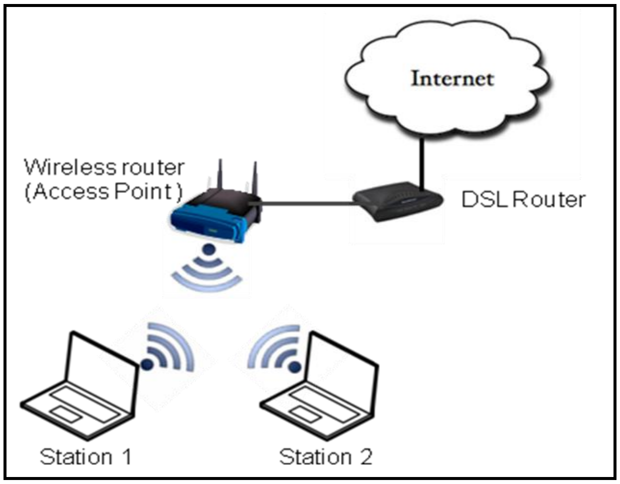

Figure [2.4]: Distribution System

2.3. 802.11 Media Access Control

The *Media Access Control (MAC)* manages and maintains the communication between different stations in the network. The IEEE 802.11 uses the MAC layer known as *Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)* that operates as a listen-before-talk scheme and is referred to as *Distributed Coordination Function (DCF)*. Because of the distributed approach, the 802.11 MAC possesses some important characteristics such as simplicity and robustness against failures [2].

Carrier Sense Multiple Access

The listen-before-talk scheme of the DCF is based on Carrier Sense Multiple Access. A station will individually determine when to access the Wireless Medium or channel for transmission by applying the DCF. Hence the decision making process about accessing the medium is distributed among all stations. When two or more stations detect at the same time the channel as idle, inevitably a collision occurs [2].

The channel sensing function is called *Clear Channel Assessment (CCA)* where the channel status is determined by sensing the signal power level in the channel. If the station finds that the power level in the station is above a predefined threshold, the channel is considered to be *busy,* otherwise *idle*.

In addition to the CCA, there also exists a virtual channel sensing mechanism using the *Network Allocation Vector (NAV)*. The NAV is a timer maintained by every station of duration for which the station is not allowed to transmit into the channel. The duration of the timer is obtained from a received frame from the station which is accessing the channel currently. The NAV timer is decremented continuously regardless of the status of the channel and as long as the NAV timer is active, a station assumes that the medium is busy and does not initiate a transmission.

Collision Avoidance

The *Collision Avoidance (CA)* mechanism in the DCF reduces the probability of collisions, by which a station performs a so called *backoff* procedure before initiating a transmission. Every station keeps sensing the medium for an additional random time called the *backoff* time after detecting the medium as idle for a minimum duration called *DCF Inter-Frame Space (DIFS)*. The station will initiate its transmission only if it finds that the medium remains idle after this additional random time. The duration of this random time is determined as a multiple of *slot time* (theoretical maximum time taken by a frame to travel from one node to another) by every station individually and the value changes randomly during each new transmission attempt [2].

Synchronization

In an Infrastructure BSS, the *Beacon* frames are the management frames transmitted from the AP periodically so as to maintain synchronization between the stations. The Beacon frame carries information such as *Targeted Beacon Transmission Time (TBTT)* which is the time of arrival of the next Beacon frame (used by stations in power save mode) and the timestamp from the local clock in the AP (used to synchronize the stations).

All stations within the same BSS are synchronized with a common clock using a *Timing Synchronization Function (TSF)* and every station maintains a local timer called *TSF timer*. In an infrastructure BSS, the AP shall be the timing master for the TSF. The Beacon frames transmitted from the AP periodically contains a copy of its TSF timer to synchronize the TSF timers of other stations. When a station receives the Beacon frame, it shall always accept the timing information in the frame sent from the AP. If the TSF timer value of a station is different from the timestamp in the received Beacon frame, the station will set its local TSF timer to the received timestamp value [1].

Link Adaptation

Link Adaptation is done for the matching of the modulation scheme, coding rate and other protocol parameters between communicating stations according to the conditions on the radio link so as to perform error-free frame exchange. A station can request another station in the network to provide the *Modulation and Coding Scheme (MCS)* feedback. The MCS information includes the specification of the PHY parameters that consists of modulation order and coding rate used for the transmission. The MCS information is then exchanged along with control frames known as link adaptation frames [11].

Point Coordination Function

The DCF method defined in IEEE 802.11 does not provide any support to provide *Quality of Service (QoS)* for the time bound services (e.g. voice) since there is no mechanism to provide priority access to the radio channel for the stations. Therefore the *Point Coordination Function (PCF)* access method is defined that uses a *Point Coordinator (PC)*, which is typically located at the AP, to determine which station currently has the right to transmit by polling the stations [2].

The time within an 802.11 station operates is divided into repeated periods called *superframes*. A superframe is composed by *Contention Free Periods (CFP)* and *Contention Periods (CP)*. During the CFP the PCF is used to access the channel and during the CP, DCF is used. In CFP there is no contention to access the channel; instead the stations are polled by the PC. At the beginning of CFP the PC will transmit a Beacon frame and after the Beacon, the PC starts polling the stations using CF-Poll frame. When the PC has a pending frame to a station, it transmits by piggy-backing the CF-Poll frame on to the data frame. To prevent non-polled transmissions by stations during the CFP, the NAV is set to the maximum CFP duration at the beginning of the CFP through the Beacon frame. The PC continues polling stations until the CFP expires and is indicated to the stations by sending a CF-End frame. An example PCF operation involving two stations and a PC is shown in the Figure [2.5] below [2].

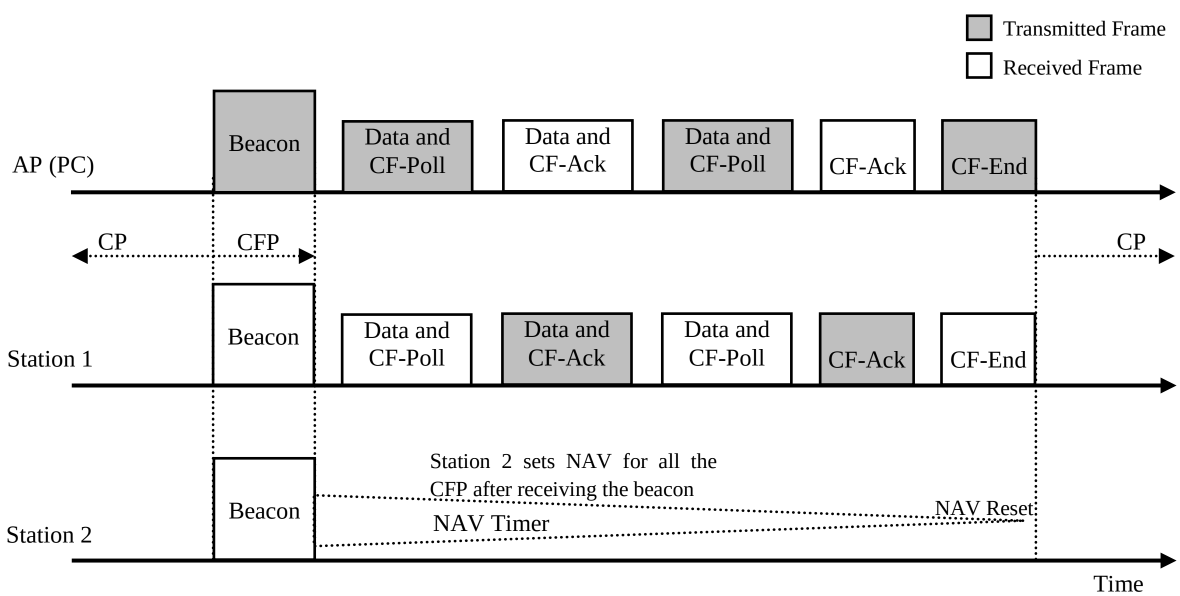

Figure [2.5]: PCF Operation

Although PCF access mechanism provides QoS to time bounded services, it has some issues like unpredictable Beacon delays and unknown transmission durations of the polled stations that will result in a limited QoS to the other stations polled in the same CFP.

2.4. 802.11 MAC QoS Extensions

The IEEE 802.11e is an extension provided to the 802.11 standard mainly to provide QoS support to time bounded applications – guaranteeing a minimum throughput for the services at the same time restricting the delay limited to the application requirements. The main enhancement in the extension is the introduction of the *Hybrid Coordination Function (HCF)* that combines functions from the DCF and PCF with some enhanced, QoS-specific mechanisms and frame subtypes to allow a uniform set of frame exchange sequences to be used for QoS data transfers during both the CP and CFP. The HCF uses both a contention-based channel access method, called the *Enhanced Distributed Channel Access (EDCA)* mechanism for contention-based transfer and a controlled channel access, referred to as the *HCF Controlled Channel Access (HCCA)* mechanism, for contention-free transfer [1].

HCF contention-based channel access (EDCA)

To support QoS, EDCA delivers traffic with differentiated treatment according to different *User Priorities (UP).* The following parameters are varied for the different UP's to achieve the differentiation: [1]

- The amount of time a station senses the channel to be idle before backoff or transmission
- The length of the contention window to be used for the backoff
- The duration for which a station may transmit after it acquires the channel.

The EDCA mechanism uses eight different UPs to provide differentiated access for the stations to the Wireless Medium. The EDCA mechanism defines four *Access Categories (AC's)* from the UP's to provide support for the delivery of traffic with priorities at the station. The AC's are derived from the UPs as shown in Table [2.1] [1].

Priority	UP (Same as 802.1D user priority)	802.1D designation	AC	Designation (informative)
Lowest	1	BK	AC_BK	Background
	2	--	AC_BK	Background
	0	BE	AC_BE	Best Effort
	3	EE	AC_BE	Best Effort
	4	CL	AC_VI	Video
	5	VI	AC_VI	Video
	6	VO	AC_VO	Voice
Highest	7	NC	AC_VO	Voice

Table [2.1]: UP-to-AC Mappings [1]

The EDCA parameters for each AC to be used by the backoff entities to obtain a *Transmission Opportunity (TXOP)* are distributed by the *Hybrid Coordinator (HC)*, typically the AP, and can be modified over time via information fields included in the Beacon frames.

EDCA modifies the 802.11 transmission procedure of a backoff entity by introducing a new inter-frame space defined as *Arbitration Inter-Frame Space (AIFS)*. The duration of AIFS for an AC is calculated based on the *Arbitration Inter-Frame Space Number (AIFSN)* as follows: AIFS = SIFS + AIFSN × aSlotTime; where *aSlotTime* refers to the duration of a slot and SIFS is the *Short Inter-Frame Space* SIFS is the shortest of the IFSs. SIFS is used to prevent other stations from attempting to access the medium after a station has gained access to the channel thereby giving priority for the station to perform the frame exchange sequence in progress [1]. A smaller AIFSN results in a higher priority for the AC while accessing the medium since the backoff counter starts decrementing after detecting the medium being idle for a duration equivalent to the AIFS.

EDCA mechanism also introduced *Contention Window minimum (CWmin)* and *Contention Window maximum (CWmax)* size to be dependent on the AC. CWmin value is used as the initial value to determine the random backoff period, the smaller the CWmin value for a specific AC the higher the priority for the AC to access the Wireless Medium. The CWmax value refers to the maximum limit of the CW value that can be reached by a certain AC, the smaller this value is the higher chances to win a contention with other stations when trying to gain access to the shared Wireless Medium. The default EDCA Parameter Set element parameter values are given in the Table [2.2] below: [1]

AC	CWmin	CWmax	AIFSN	TXOP limit
AC_BK	aCWmin	aCWmax	7	0
AC_BE	aCWmin	aCWmax	3	0
AC_VI	(aCWmin+1)/2 – 1	aCWmin	2	3.008ms
AC_VO	(aCWmin+1)/4 – 1	(aCWmin+1)/2 – 1	2	1.504ms

Table [2.2]: Default EDCA Parameter Set element parameter values [1]

The difference between a legacy 802.11 station and 802.11e station is shown in the Figure [2.6] below. In an 802.11e station, when two backoff entities try to access the same slot in parallel, the higher priority AC backoff entity transmits finally. The other backoff entities act as if a collision occurred and this is referred to as *Virtual collision* mechanism as represented in the figure.

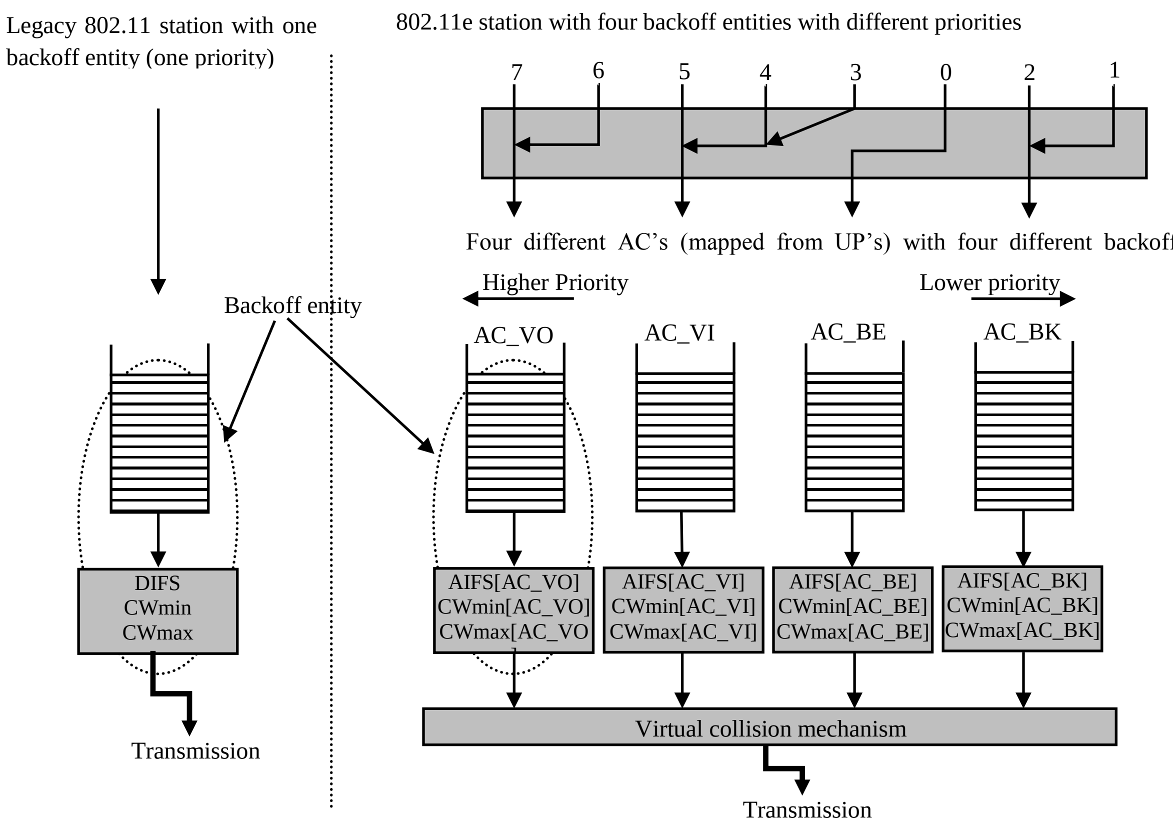

Figure [2.6]: Comparison between legacy 802.11 and 802.11e stations [2]

HCF Controlled Channel Access (HCCA)

The HCF Controlled Channel Access (HCCA) allows for the reservation of TXOP's with the HC, which is typically the AP. A station based on its requirements requests the HC for TXOP's, both for its own transmission as well as for the transmission from the AP to itself. The HC either accepts or rejects the request based on an admission control policy. If the request is accepted, the HC schedules TXOP's for both the AP and the station. For transmissions from the station, the HC polls the stations based on the parameters supplied by the station at the time of its request. For transmissions to the station, the AP directly obtains TXOPs from the HC within the AP and delivers the buffered frames to the station, again based on the parameters supplied by the station. This mechanism is suitable for time bounded applications such as voice and video, which may need periodic service from the HC to meet their QoS.

MAC Architecture

To summarize, the MAC architecture can be represented as shown in Figure [2.7] as providing the PCF and HCF through the services of the DCF. Note that in a non-QoS station, HCF is not present and in a QoS station, DCF and HCF are present. PCF is optional in all stations. [1]

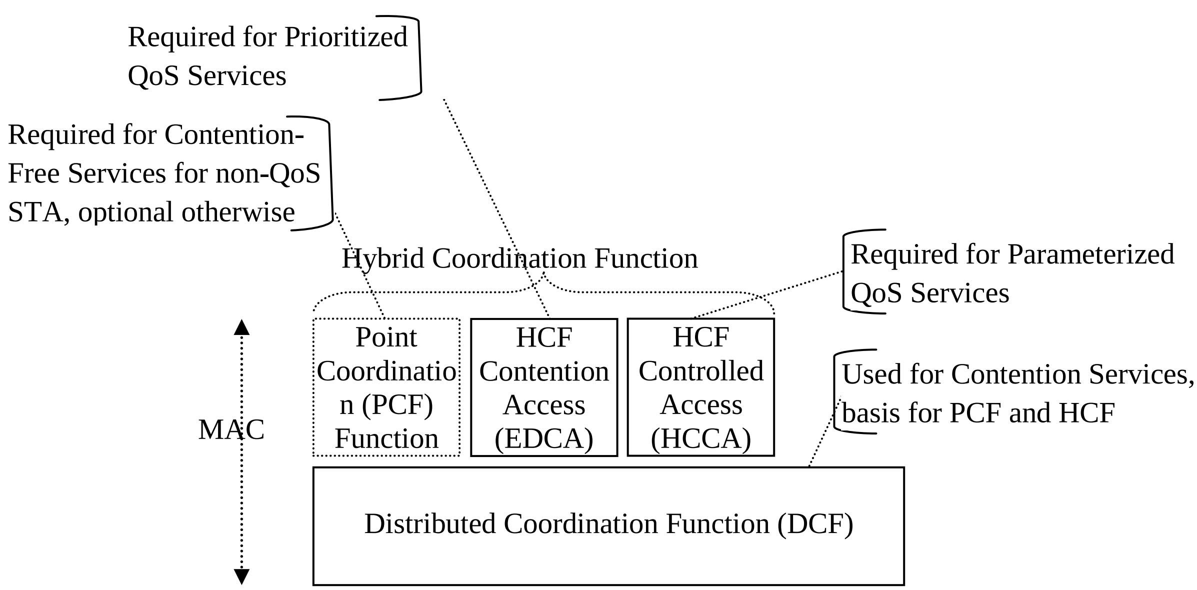

Figure [2.7]: MAC Architecture [1]

2.5. 802.11 Physical Layer

The 802.11 Physical layer is the interface between the Wireless Medium and the MAC layer. The legacy 802.11 standard defines three different types of physical layers *Frequency Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum (DSSS)* and *Infrared (IR).* In this section, only the basic information regarding the Physical layers is discussed as a detailed explanation is beyond the scope of this thesis.

Frequency Hopping Spread Spectrum

Spread Spectrum is a technique by which the electromagnetic energy generated in a particular bandwidth is spread in the frequency domain so that the transmitted bandwidth of the resultant signal is increased by reducing the peak power. In FHSS, the whole frequency band is divided into a set of narrow channels and the station jumps from one channel to another at a predefined cyclic pattern. The sequence of frequencies of the channels is predefined and is known to both transmitter and receiver.

Direct Sequence Spread Spectrum

In DSSS, the station uses the same center frequency but the signal is spread by multiplexing with different spreading codes to reduce the interference between signals and the background noise. The receiver then decodes the original signal using the same code used by the transmitter.

Infrared

In Infrared based Physical layer, infrared light is used to transmit binary data using a specific modulation scheme such as *Pulse Position Modulation (PPM)* in which the different binary represented by varying the position of the pulse.

2.6. Protocols

The initial draft of the WLAN 802.11 standard supports data rates up to 2 Mbps, and later several amendments were made to the original version such as 802.11a/b/g/n to support higher data rates and provide more enhanced features.

802.11b

802.11b uses a physical layer with DSSS and with the complementary-code keying modulation scheme, provides data rates up to 11 Mbps. But, because of the packet overheads the effective throughput achieved was around 5 Mbps and this was insufficient for many applications such as Video.

802.11a

The IEEE 802.11a amendment specifies the Physical layer operating at 5.2 GHz using a transmission scheme known as *Orthogonal Frequency Division Multiplexing (OFDM)* allowing data rates up to 54 Mbps. Although 802.11a provides much higher data rates, the use of 5.2 GHz prevented it replacing the 802.11b standard because of the production cost of manufacturing devices capable of supporting both 2.4 GHz and 5.2 GHz, to support both standards. Also 802.11a was introduced only in North America and the use of 5.2 GHz carrier in Europe was restricted. [3]

802.11g

The 802.11g standard defines a PHY layer with similar specifications as 802.11a, using the OFDM transmission scheme and providing PHY data rates up to 54 Mbps, but based on a 2.4 GHz carrier. This standard also allows 802.11b devices to join the 802.11g networks thereby providing a backward compatibility which was one of the main challenges during the development of this standard. [3]

802.11n

The 802.11n amendment describes enhancements in both the PHY and MAC layers and has the potential of offering higher data rates up to 200 Mbps based on the physical layer data rates up to 600 Mbps. The key changes in the Physical layer includes the use of multiple antennas for transmit and receiving, referred to as *Multiple Input Multiple Output (MIMO)* that uses techniques such as *Spatial Division Multiplexing (SDM)*, transmitter beamforming and *Space Time Block Coding (STBC)* which also helps to improve the range of reception. [3]

The scope of this thesis is only limited to the MAC layer enhancements introduced in the 11n amendment. Hence the enhancements in the MAC layer are described in detail in the next section.

2.7. MAC Enhancements in 80211n

Frame aggregation is one of the key enhancements introduced in the 802.11n MAC layer in which multiple MAC frames are sent as one PHY layer packet to reduce overhead. Frame aggregation can be performed by aggregating *MAC Protocol Service Unit (A-MSDU)* or the *MAC Protocol Data Unit (A-MPDU).* MSDU corresponds to the frames exchanged between the upper part of MAC sub layer and the higher layers, whereas MPDU corresponds to frames that are exchanged between the lower part of the MAC and the Physical Layer.

In A-MSDU, a single MPDU is formed by concatenating multiple MSDU's to the same receiving station. This improves the efficiency of the MAC layer especially when there are several small MSDU's, e.g. TCP acknowledgements. To form an A-MSDU, the top layer of the MAC receives and buffers multiple packets (MSDU's) and the A-MSDU is completed either when the size of the waiting packets in the buffer reaches the maximal A-MSDU threshold or the maximal delay of the oldest packet reaches a pre-assigned value [3].

In A-MPDU aggregation, multiple MPDU subframes are concatenated with a single leading PHY header. The main difference between A-MPDU and A-MSDU aggregation is that A-MPDU functions after the MAC header encapsulation process. Hence, the A-MPDU does not have the restriction of aggregating the frames with matching *Traffic Identifiers (TID)* as in A-MSDU aggregation. However, all the MPDUs within an A-MPDU must be addressed to the same receiver address. The number of MPDU's to be aggregated depends only on the number of packets already in the transmission queue and hence there is no waiting or holding time to form an A-MPDU as in A-MSDU. In an aggregated frame exchange sequence the acknowledgement for the multiple MPDUs are made with a single block acknowledgement frame in response to a *Block Acknowledgment Request (BAR)* sent along with the aggregated frame. [3]

Another key enhancement in the 802.11n is a bidirectional data transfer method over a single TXOP, known as *reverse direction* in which the exchange of data frames are done in both directions in one TXOP. In the usual case, by a TXOP the sender station informs the surrounding stations about how long the wireless medium will be engaged. However, this method is not efficient since quite often the transmission ends before the TXOP expires and the contended stations assume that the channel is still busy when it is not. Using the reverse direction method, the remaining TXOP time can be used by the receiving station to send back any packets available that are addressed to the sender. [3]

The *long Network Allocation Vector (long-NAV)* is another enhancement that improves scheduling by which a station that has access to the channel may set a longer NAV value intended to protect multiple frames. *Phased Coexistence Operation (PCO)* is another important feature to protect stations using either 20 MHz or 40 MHz channel spectrum at the same time. Finally, the SIFS as defined in the legacy standard (16 μs) is reduced to a much smaller value, *Reduced IFS (RIFS)* of 2 μs between multiple frames. [3]

3. Power Saving Modes

The IEEE 802.11 standard defines power management mechanisms depending on whether infrastructure or ad-hoc mode is used. The focus of this thesis is on the power saving mechanisms in stations connected in an infrastructure network, where the WLAN stations are connected to an AP. Using these power saving mechanisms, the wireless terminals or stations can turn off the transmitter and receiver chipset to minimize its power consumption while connected to the network. In this section, the two power saving mechanisms are described - the standard infrastructure power saving mechanism known as *Standard Power Save Mode (PSM)* defined in the legacy 802.11 and *Automatic Power Save Delivery (APSD)* which is an extension to the PSM, provided in the 802.11e standard.

3.1. Standard Power Save Mode (Std PSM)

Access Point Operation

In infrastructure networks, the power management is centralized in the AP. The AP maintains the power management status of the stations associated with it and the information regarding the power management mode in which the station is operating is communicated to the AP during the association process by means of the power management bits in the control field of the transmitted frames.

The data frames are buffered at the AP that has to be transmitted to the associated stations which are in the power save mode. The AP informs the stations in the power save mode whether there is data buffered in the buffer or not with the *Traffic Indication Map (TIM)* field in the Beacon frame. During the association process a unique *Association ID (AID)* code is assigned to every station and the AID indicates with a single bit in the TIM if there are frames buffered for the specific station.

The Beacon frames are transmitted periodically at every TBTT and the stations can determine the next Beacon frame arrival time using the TBTT information transmitted from the AP before. The stations wakeup at every Beacon arrival time and if there is data buffered in the AP for a station, it requests the delivery of the buffered frames by sending a *Power Save Poll (PS-Poll)* frame. The AP then delivers a single frame for the corresponding PS-Poll. Once the frame is successfully delivered and on receiving a new PS-Poll, the next buffered frame is delivered.

Station Operation

A station in the power save mode wakes up every *n* Beacon arrivals, where *n* is an integer ≥ 1. The wakeup interval of the station is communicated to the AP at the association time. The station learns from the TIM field in the Beacon if there is any data buffered for the station in the AP. The station sends the PS-Poll frame to the AP if it finds that there is data buffered for the station. The AP responds to the PS-Poll with an *Acknowledgement (ACK)* frame or directly with the data frame. In this thesis, it is assumed that the AP responds with an ACK frame for the PS-Poll and then the data frame is transmitted for which the station sends back an ACK. If the station does not receive a response to the PS-Poll, it retries the sequence and start sending another PS-Poll frame and this is repeated until the retry limit for a frame transmission is reached.

The AP sets a flag called the *More Data (MD)* in the frame control field of the data frame corresponding to if there is more data buffered in the AP for the particular station. The station sends another PS-Poll to the AP if it finds the MD bit is set in the last received frame and this is repeated until all frames buffered in the AP is delivered to the station. When the station has to transmit a data frame to the AP in the *Uplink (UL),* it wakes up immediately and starts the transmission sequence irrespective of the Beacon arrival. The Figure [3.1] below illustrates an example of the Standard PSM operation in the infrastructure mode. [7]

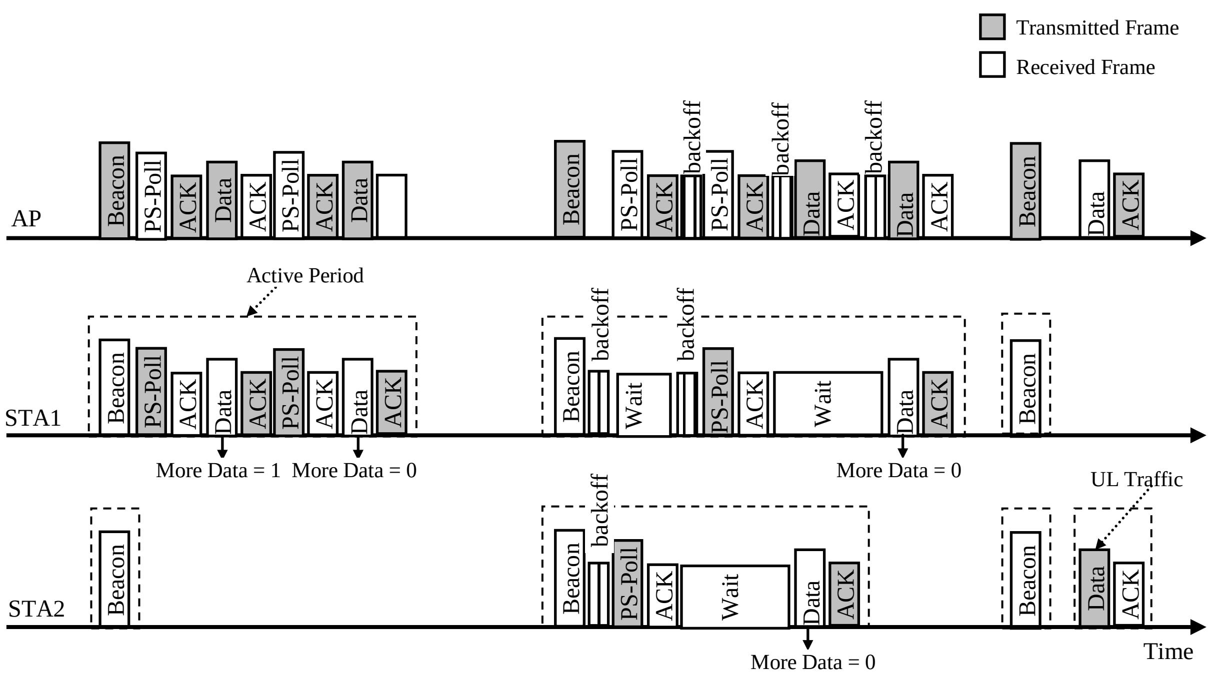

Figure [3.1]: Standard Power Save Mode operation

In the beginning of the sequence, it can be seen that the Station 1 (STA1) and Station 2 (STA2) wakes up for receiving the Beacon. After receiving the Beacon, STA2 finds from the TIM information that there is no packets buffered for it in the AP and hence it goes back to sleep state. But STA1 finds that there is data buffered for it in the AP and it sends a PS-Poll in the uplink. On receiving the PS-Poll, the AP sends back an ACK frame for the PS-Poll and followed by a data frame that is buffered at the AP for the station. When STA1 receives the data frame, it sends an ACK for the same and at the same time it finds from the MD flag that there is more data buffered in the AP. Now STA1 sends another PS-Poll and the AP delivers the next data frame from its buffer. From the received data frame, STA1 finds that there is no more data buffered at the AP and then it goes to the sleep state.

At the next Beacon arrival interval, the stations wakes up to receive the Beacon frame and both the stations finds that there is data buffered for them in the AP. When both the stations is ready to transmit a PS-Poll frame in the uplink, the stations perform backoff procedure to avoid a collision and after the contention process STA2 gains access to the channel first and starts transmits the PS-Poll frame. During the time when STA2 is accessing the channel, STA1 will be in the wait state and once STA2 finishes its transmission STA1 gains access to the channel and transmits its PS-Poll frame. During the time when STA1 is transmitting the PS-Poll, STA2 will be waiting for the data from the AP. Finally data is delivered by the AP to the stations in the order they were requested. As can be seen in the figure, for Uplink transmission STA2 wakes up immediately and transmits the data irrespective of the Beacon arrival as can be seen in the figure after the third Beacon interval.

3.2. Automatic Power Save Delivery (APSD)

Automatic Power Save Delivery (APSD) included in the 802.11e draft is an optional extension to the previously defined 802.11 Power Save Mode. The main difference of APSD with the Standard Power Save Mode is that in APSD the station is awake during a *Service Period (SP)* instead of being awake for the duration a PS-Poll is transmitted until all the frames in the AP buffer are delivered to the station. During a SP the AP can deliver one or more frames to the station without a need of a PS-Poll for every frame as in the Standard PSM. Two types of SP’s are possible with APSD - *Unscheduled* and *Scheduled*.

In *Unscheduled APSD (UAPSD)*, the uplink stream is used to retrieve the frames buffered at the AP. The SP starts when the station sends trigger frame i.e. a data or a *null frame (QoS-Null)* to the AP and the SP ends when the station receives a data or null frame with the *End-of-Servive-Period (EOSP)* bit in the frame control field set.

In *Scheduled APSD (SAPSD)* the duration of SP and the start of the SP are determined by the AP and indicated to every station so that the overlap between the SP's of different stations are minimized, and this helps the stations to reduce the time spent in the listen state by waking up only for the SP assigned to the particular station.

The focus of this thesis is on UAPSD since it is more suitable for providing differential QoS treatment to the different application categories. An example operation of the UAPSD is shown in the Figure [3.2] below in which the frame exchanges between AP and a Station (STA 1) is shown considering that the Access categories (AC) - Voice (AC_VO) and Video (AC_VI) are configured as *trigger* and *delivery enabled;* Best Effort (AC_BE) and Background (AC_BK) uses the Standard PSM. When an AC is configured to be trigger enabled, the frames of type data and QoS-Null from the station schedules starts an unscheduled SP if there is not one in progress. If the AC's are configured to be delivery enabled, then the AP can use EDCA to deliver the traffic during an unscheduled SP triggered by the station. The maximum SP length is assumed to be configured as *all* frames that deliver all the frames buffered at the AP in a SP.

In the first active period, the station wakes up to receive the Beacon frame and it finds from the TIM that AC_BE packets are buffered in the AP. So it sends a PS-Poll to the AP to download the buffered AC_BE packets. Since the MD flag is set to zero, the station goes to sleep after sending the ACK for the data frame.

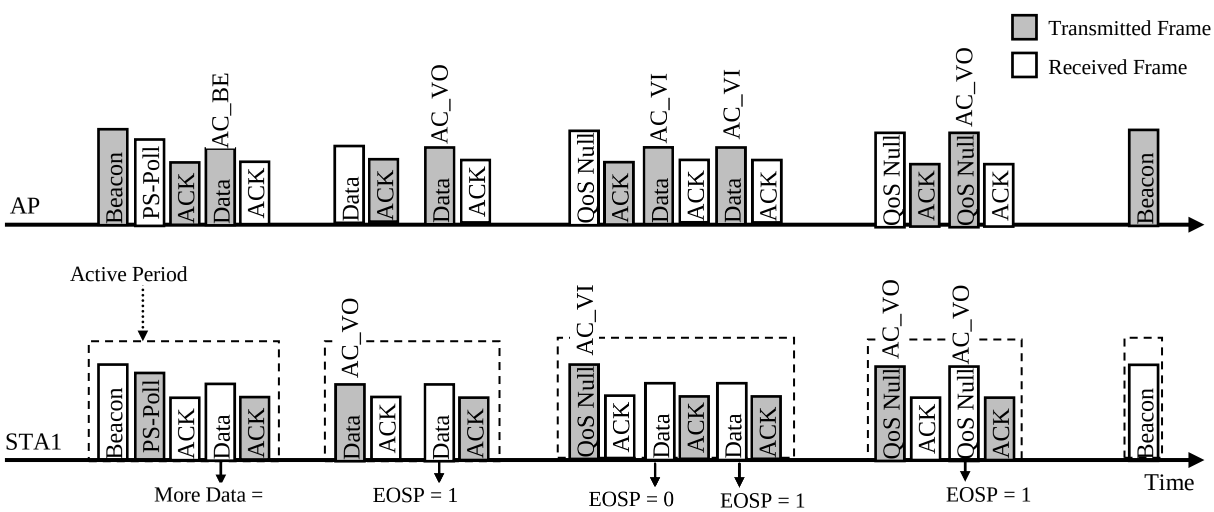

Figure [3.2]: Automatic Power Saver Delivery operation

In the second active period, the station wakes up because it has to transmit a AC_VO data packet in the uplink. Since the AC_VO is configured as trigger enabled, the SPis started and the AP sends back the buffered AC_VO packet. Now, the EOSP bit is set in the data frame received that indicates the end of the SP since there is no more buffered AC_VO packets at the AP and the station goes to the sleep state. In the next active period, the station wakes up automatically using an internal algorithm to send the QoS-Null trigger frame to meet the QoS requirements of the Video downlink traffic. On receiving the QoS-Null frame a new SP is triggered and the AP sends back the buffered AC_VI packets for the duration of the SP.

In the last active period, the station wakes up to transmit the QoS-Null trigger frame to initiate the download of AC_VO packets buffered at the AP in order to meet the QoS requirements of Voice application. The time period of the transmission of the QoS-Null trigger frames are called *Service Interval (SI)* and is configured according to the QoS requirements of different applications. For example, the Voice applications use 20ms and Video 40ms.

UAPSD Continuous mode

The *UAPSD continuous mode* is suitable for applications that do not require real-time service such as Web (AC_BE) and FTP (AC_BK). In this mode of UAPSD, the station receives the Beacon frame from the AP as in the Standard PSM and if the station finds from the TIM that there are packets buffered at the AP, a single QoS null frame is transmitted to retrieve the packets. The number of frames delivered by the AP depends on the SP configuration, and with a SP duration configuration of *all*, all the frames buffered in the AP are delivered to the station.

3.3. Advantages of UAPSD compared to Standard PSM

The main advantage in UAPSD is its possibility of providing differential QoS treatment to the different applications by generating triggers with different access priorities at any point of time for retrieving the packets buffered at the AP. Hence the packet delivery delay can be configured according to the application QoS requirements compared to the Std PSM where the minimum delay cannot be less than the Beacon inter-arrival period. Also in Std PSM the PS-Polls are transmitted from all application categories with the same priority (as of the Best Effort traffic) and hence all the applications have the equal priority in getting access to the channel. [7]

In UAPSD, the overhead of signaling frames (PS-Poll, QoS-Null) would be less since data frames in the uplink can also act as trigger to retrieve the frames buffered in the AP. Also the overhead in retrieving the packets buffered at the AP is less because with a single trigger frame more than one packet is delivered according to the duration of the SP.

4. Simulation

In this section we discuss about the features of *OPNET*, the simulator used during this thesis work for the performance evaluation of the power saving mechanisms. The features of the OPNET simulator are explained with reference to the OPNET Modeling Concepts Reference Manual. [4] Later in this section we also discuss about the OPNET use case scenarios and the application configurations used for the performance evaluation of the power saving mechanisms.

4.1. OPNET features

OPNET is the discrete event simulator from OPNET Technologies [8] that supports modeling of communication networks and distributed systems. The system models in OPNET can be analyzed for both performance and behaviour by performing discrete event simulations. The detailed library models provide major *Local Area Network (LAN)* and *Wide Area Network (WAN)* protocols and hence it can be used for standards-based LAN and WAN performance modeling. Configurable application models are also provided by the library, or new ones can be created. OPNET allows specification of fully general logic and provides extensive support for communications-related applications. Finite state machines provide a natural representation for protocols. OPNET allows development of sophisticated, adaptive, application-level models, as well as the underlying communications protocols and links. Customized performance metrics can be computed and recorded, scripted and/or stochastic inputs can be used to drive the simulation model, and processes can dynamically monitor the state of objects in the system via formal interfaces provided by statistic wires.

The key features of OPNET are:

Object Orientation: The systems specified in OPNET consist of objects, each with configurable sets of attributes. Objects belong to classes, which provide them with their characteristics in terms of behaviour and capability. Definition of new classes is supported to address as a scope of systems as possible.

Hierarchical models: OPNET models are hierarchical, naturally paralleling the structure of actual communication networks.

Graphical specification: Wherever possible, models are entered via graphical editors. These editors provide an intuitive mapping from the modeled system to the OPNET model specification.

Flexibility to develop detailed custom models: OPNET provides a flexible, high-level programming language with extensive support for communications and distributed systems. This environment allows realistic modeling of all communications protocols, algorithms, and transmission technologies.

Automatic generation of simulations: Model specifications are compiled automatically into executable, efficient, discrete-event simulations implemented in the C programming language. Advanced simulation construction and configuration techniques minimize compilation requirements.

Application-specific statistics: OPNET provides built-in performance statistics that can be collected automatically during simulations. Modeler's can also augment this set with new application-specific statistics that are computed by user-defined processes.

Integrated post-simulation analysis tools: Performance evaluation and trade-off analysis require large volumes of simulation results to be interpreted. OPNET includes a sophisticated tool for graphical presentation and processing of simulation output.

Interactive analysis: All OPNET simulations automatically incorporate support for analysis via a sophisticated interactive debugger.

4.2. 802.11n OPNET process model

The 802.11n process model has been developed at NEC Laboratories Europe from the 802.11 process model provided by the 802.11 Task Group (TG) [9]. This process model supports the important PHY and MAC layer features in the 802.11n such as MIMO, A-MPDU aggregation and Reverse Direction protocol. In this thesis work, the process model is extended with the power saving mechanisms Standard PSM and the UAPSD to evaluate their performance.

4.3. Scenarios and application configurations

The use case scenario considered in this evaluation is the *Hot Spot* scenario where most of the traffic goes through the internet and the duration of the session will be limited [5]. In such a Hot Spot scenario, all the stations are distributed over a circular area within 30meters from the AP. The Figure [4.1] below shows the snapshot view from such a scenario created in OPNET simulation environment.

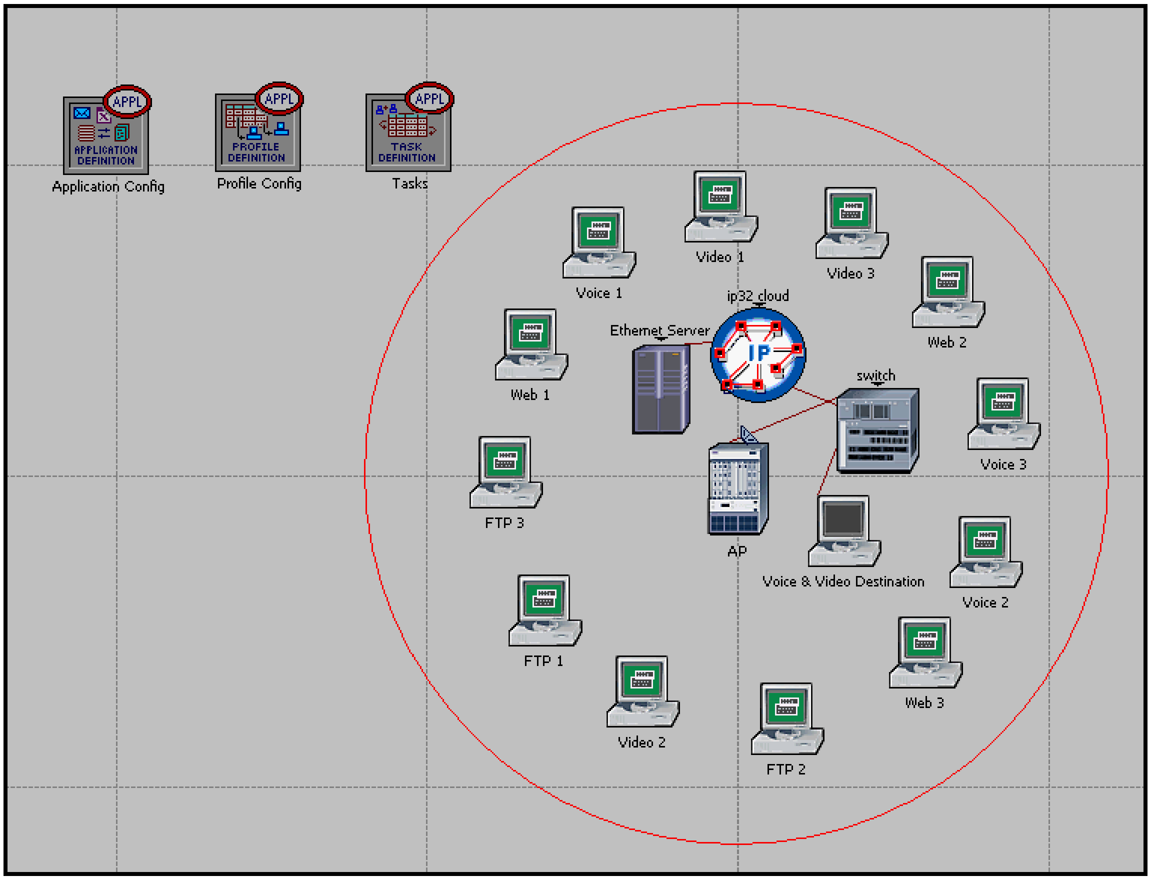

Figure [4.1]: Snapshot image of the Hot Spot scenario in OPNET

The basic scenario is composed of four WLAN stations, each of them configured to send and receive traffic in the four different application categories *Voice*, *Video*, *FTP* and *Web*. In the figure there are 12 stations, 3 of each application categories. Also in the figure, the *Ethernet Server* generates the FTP and Web traffic and the module *ip_32_cloud* is added to introduce a *Round Trip Time (RTT)* between the Access Point (AP) and the Ethernet Server to match with the real time TCP traffic. The default value of RTT is configured to be 20ms. The *Voice and Video Destination* node generates the traffic for the Voice and Video Stations. The simulation settings used for the different access categories are given below: [7]

Voice: G.711 Voice Codec with silence suppression. Data rate of 64Kbps and Frame length of 20ms. Talk spurt exponential with mean 0.35 seconds and silence spurt exponential with mean 0.65 seconds

Video: MPEG-4 real traces with average rate of 1Mbps and peak of 11Mbps. Frames are generated with an interval of 40ms.

Web: HTTP version - HTTP 1.1, Page inter-arrival time is exponentially distributed with mean time of 15 seconds. Page size of 10KB with 20 to 80 objects of size uniformly distributed between 5 and 10KB.

FTP: Download file size of 20MB.

The Beacon interval is configured to 100ms at the access point and the listen interval also configured to the same value in the stations so that the stations configured to use the Standard PSM and UAPSD Continuous mode wakes up to receive every beacon transmitted from the AP.

In UAPSD mode, the voice stations are configured to send QoS-Null trigger every 20ms and Video stations at every 40ms; FTP and Web stations are configured to use the UAPSD Continuous mode and the SP duration configured for all the stations is '*All*'.

The simulations were performed for a total duration of 150 seconds including a warm up time of 25 seconds and three seeds of each simulation were performed. The number of stations of each access category is increased in multiples of three every time and a new simulation is performed and the statistics for performance evaluation are collected as explained in the next section.

4.4. Statistics collection and results processing

Average Power Consumption: The time spent by every station in each of the four different states – *Sleep, Idle, Receive and Transmit* is measured during the simulations. The power consumption in every station is then calculated by computing the percentage of time spent in each of the state with respect to the simulation time and then multiplying the value with the power consumption of a standard Wireless LAN chipset. The average of this value from all the stations of the same access category is calculated to find the Average Power Consumption. The Wireless LAN chipset under consideration has the following values for power consumption. [6]

States	**Sleep**	**Idle**	**Receive**	**Transmit**
Power Consumption (mW)	20	390	1500	2000

Table [5.1]: Typical Wireless LAN chipset power consumption values

To obtain a more reliable plot, the results from the three seeds of simulations are plotted with 95% *Confidence Interval*. The 95% Confidence Interval is calculated using the following relation:

$$\frac{1.96 * \text{Standard deviation (Results per seed)}}{\text{Square root (Number of seeds)}}$$

Delay: The delay in the packet transmission is measured by computing the difference between the time the packet arrives to the MAC layer from the higher layer for transmission and the time when the packet reaches the destination station. For reliability, the average of all the packet delays that falls within the 95% of the total number of packet delays is calculated and plotted.

Throughput: Throughput for Voice and Video traffic is measured by measuring the number of number of packets successfully received at the destination station from the source station and dividing it over the simulation time to get the data rate. Throughput for FTP stations is measured considering only the duration for which the file download actually takes place. The throughput also is plotted with 95% Confidence Interval.

Page Response Time: Web page response time is the time taken by a station to download a complete web page from the time the request for download is sent to the Web server. Measuring throughput for Web traffic does not help for the analysis of performance since the small sized individual objects comprised in the Web page arrives along with every window of TCP packets and the stations goes to sleep (because of power saving mechanism) before the next window of TCP packets and as a result the effective round-trip-time as seen by the TCP server increases and the throughput will be less. [7]

Signaling Load: Signaling load is calculated by measuring the time taken for the transmission of the signaling frames (Link adaptation, PS-Poll and QoS-Null frames) over the simulation time. Signaling load is measured to validate the Power and QoS performance results.

Number of Collisions: When two or more stations transmit at the same time, a collision occurs. This parameter is also measured to validate the Power and QoS performance results, especially when the network congestion increases.

Number of Power Captures: Power capture or *Capture effect* is a phenomenon observed when two or more frames are transmitted at the same time from different stations in the Uplink and the frame with the strongest power may capture the receiver (AP) [10]. This should have been inevitably resulted in a collision, but the AP decodes one of the packets based on the power levels of the received signals and acknowledges the corresponding station. The number of Power captures is measured in order to explain its impact on the performance of the different power saving mechanisms.

5. Performance Evaluation of Power Saving Modes

To study the impact of frame aggregation on the power saving modes PSM and UAPSD, the simulations are performed using the 802.11n OPNET process model with frame aggregation and without using frame aggregation feature. Hence there are four different simulation configurations:

- Standard PSM without frame aggregation (Std PSM)
- Standard PSM with frame aggregation (Std PSM+Agg)
- UAPSD without frame aggregation (UAPSD) and
- UAPSD with frame aggregation (UAPSD+Agg).

In this section we discuss the performance results of the power saving modes considering real-time application (Voice and Video) and non real-time application (Web and FTP) categories individually. Under each application category the performance metrics like Power consumption and QoS are discussed for both the cases – with and without frame aggregation. The different performance metrics are linked to each other, but they are discussed separately for better clarity.

5.1. Real-Time applications

The Figure [5.1] below shows the simulation results for the real-time application categories - Voice and Video.

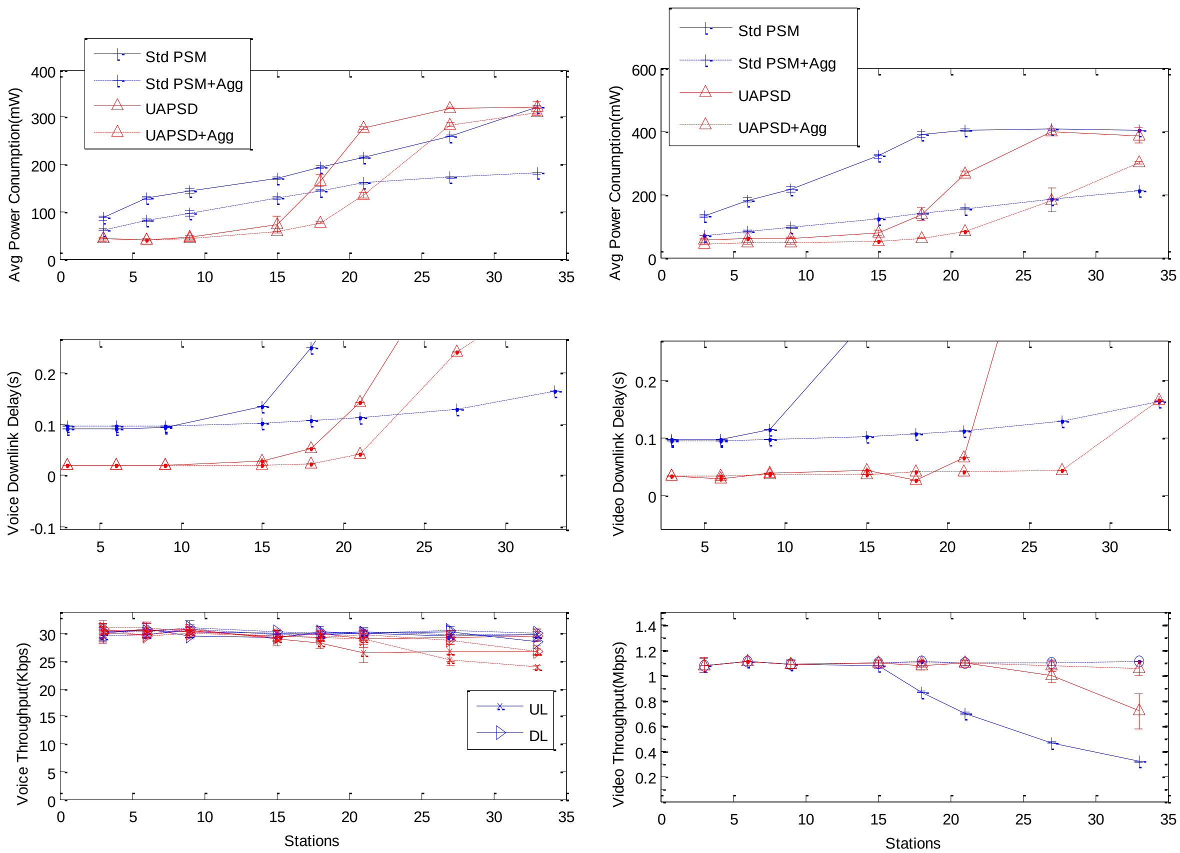

Figure [5.1]: Performance results of Voice and Video applications

Without frame aggregation

Power consumption: Every station using the Standard PSM wakes up during the Beacon arrival time in order to receive the Beacon frame from the AP and starts transmission of the PS-poll frame immediately if the station finds that there are frames buffered for the corresponding station in the AP. When all the stations try to transmit the PS-Poll frame at the same time, they have to wait more time in the Idle power state before it gets access to the channel, because of the contention process in the CSMA/CA protocol. Also, since the Beacon frame is synchronizing the stations, there is a higher collision probability of the transmitted frames. As the PS-Poll frames from all the stations are transmitted with the same access priority of AC_BE, the stations servicing different access categories in the network contents with the same priority to get access to the channel and the stations has to wait in the Idle state until the AP delivers the buffered frames for the corresponding station. The reason for using the low access priority of AC_BE for PS-Poll is that, it could be problematic to use a higher priority (smaller contention settings), because smaller contention settings means higher collision probability if there are many stations and when all stations get synchronized with the Beacon.

In UAPSD, the trigger frames (Data and QoS-Null) are transmitted by the stations periodically at different time instants and hence the contention for accessing the channel is minimized because of the distributed access to the channel. Also, the QoS-Null frames are transmitted with the same access priority as that of the application type and hence the real-time applications get higher priority in getting access to the channel. Also, the data frames in the uplink (from the station to the AP) also acts as trigger to download the buffered frames at the AP and this reduces the overhead of signaling frames unlike in Std PSM, especially for applications with bi-directional communication like Voice over Internet Protocol (VoIP). Because of these reasons the average power consumption is low for UAPSD compared with the Std PSM when the network is not congested with only a few numbers of stations (less than 15 stations as seen in the Figure [5.1]).

The more the number of stations trying to access the channel at the same time the more the time required to get access to the channel because of the contention process. Hence the stations will be waiting more time in the Idle power state to get access to the channel and also until the packets from the AP are delivered to the station. Also there will be more number of collisions of frames when the network is congested with more stations; and this is evident in the Figure [5.2] below. Another important reason for the increase of power consumption is also the queuing delay in the AP. The queue in the AP grows because of the increase in the downlink load with more number of stations and hence the time taken by the AP to deliver the downlink data for a station queued in the AP gets much longer. The average power consumption increases since the station has to be awake as long as AP delivers data.

Delay: As can be seen from the Figure [6.1] with few stations in the network, the lowest downlink delay in Std PSM will be 100ms because the frames buffered at the AP are only downloaded every Beacon interval which is configured to 100ms. Therefore Std PSM cannot fulfil the QoS requirements of the real-time applications like VoIP and Video. In UAPSD, the frames are downloaded from the AP whenever a trigger frame is sent from the station and for Voice stations the QoS-Null trigger frames are configured to be transmitted every 20ms and Video stations every 40ms, hence their corresponding minimum downlink delays are 20ms and 40ms respectively as can be seen in the results in Figure [5.1].

As explained before, when the network congestion increases with more number of stations, the queuing delay in the AP increases and this means that the Downlink delay will be increased as can be seen in the Figure [5.1].

Throughput: From the results in Figure [5.1], it can be seen that the voice throughput remains constant because the voice packets are small and there is also silence suppression, so it uses less channel bandwidth. But the video throughput reduces when frame aggregation is not used and the throughput reduces because some of the packets arriving from the upper layer are discarded without being transmitted. The packets could be discarded because of two reasons – when the retry limit of transmission attempt is reached after repeated collisions or if the buffer in the AP overflows. In Figure [5.1], the video downlink delay for Std PSM without frame aggregation increases rapidly after about 15 stations and at the same time the throughput reduces. This means that the buffer in the AP reached the maximum level and the new video frames arriving from the higher layer of the AP are discarded before transmission. The throughput also reduces with UAPSD, but happens with more number of stations than in Std PSM because of the differential treatment to the access categories in UAPSD and video has higher priority than the TCP traffic.

With frame aggregation

Power consumption: Using frame aggregation in Std PSM, all the frames buffered in the AP for a particular station are aggregated and transmitted on receiving a single PS-Poll frame from the station since the SP duration is configured to *All*. Hence the contention will be less among the stations to get access to the channel as the overhead in sending PS-Poll's per frame and ACK per frame is reduced and the packets buffered at the AP are delivered quickly. The effective time spent by the stations in the active state (Idle, Receive and Transmit) will be less and hence the average power consumption in Std PSM is less when frame aggregation is used as can be seen from the results.

In UAPSD, using the frame aggregation does not improve the power saving when the network is not congested, because the trigger frames from the stations are configured at a Service Interval which is at the same rate of packet generation rate of the respective application. This means that there will not be enough packets in the buffer of the AP at a time to use the frame aggregation efficiently. But when the network gets congested by increasing the number of stations (more than 15 stations), the number of packets in the AP buffer grows and using the frame aggregation more number of frames is fit into a single TXOP compared to non-aggregated case. Hence the stations have to wait for the packets from the AP for a comparatively less time than the non aggregated case. This is the reason for the lower power consumption when frame aggregation is used in UAPSD under congested network conditions.

Delay: When frame aggregation is used in Std PSM, the downlink delay is less compared to non-aggregated case because of lower contention due to efficient aggregation and also because of the reduced overhead in sending PS-Poll's per frame and ACK per frame. In UAPSD, when the network is not congested the packets are delivered to the station as soon as it sends a trigger to the AP after the packet had arrived from higher layers. Hence the worst case delay when the network is not congested will be the duration of a Service Interval. This highlights the fundamental trade-off between delay and aggregation capability, i.e. smaller the delay lesser the amount of aggregation.

Throughput: Because of efficient aggregation under congested network conditions, the throughput does not drop for Std PSM and UAPSD as can be seen in the Figure [5.1], unlike the non-aggregated case.

Impact of Signaling, Collisions and Capture effect

As can be seen from the Figure [5.1], with more number of stations (above 20 stations), the average power consumption with UAPSD and frame aggregation is more than that of Std PSM with frame aggregation and this behaviour was not expected. The reason for this can be explained using the signaling load graph in the Figure [5.2] where it is shown the percentage of channel usage by the signaling frames – link adaptation frames and PS-Polls in Std PSM; link adaptation frames and QoS-Null in UAPSD. The percentage of signaling and number of collisions shown in the graph is measured from all the stations servicing all access categories in the scenario but the number of stations on the x-axis represents the number of stations in each access category in order to make a comparison with the previous graph.

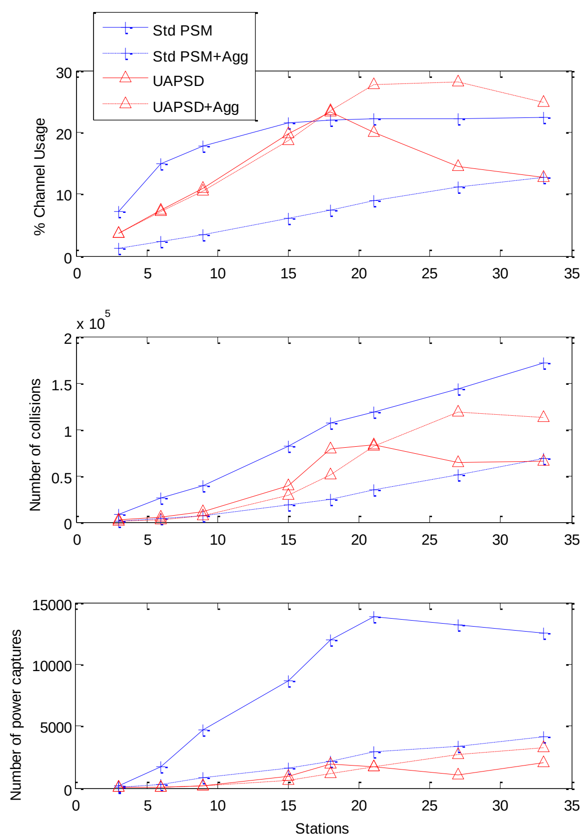

Figure [5.2]: Signaling load and Collisions

From the signaling load graph (% Channel Usage) it can be seen that with UAPSD the signaling increases rapidly as the number of station increases mainly because of the increase in the number of QoS-Null trigger frames sent in the uplink from all the stations and the trigger frames are sent at a much higher rate than the PS-Poll’s in Std PSM. Also note that the voice stations use silence suppression and during the silence periods also the trigger frames are sent in UAPSD but not in Std PSM. The signaling load graph saturates after about 20 stations and this means that the maximum network capacity has reached. With further increase in the number of stations, the signaling goes down because the SP gets longer and new triggers will not be sent until the previous SP is finished. The signaling for Std PSM without frame aggregation also increases because of the PS-Poll’s per frame. But the signaling load in Std PSM with aggregation does not increase rapidly because buffering the frames and the efficient use of aggregation results in less number of PS-poll’s providing more network capacity. This is also the reason why the delay and average power consumption in congested network conditions are better with Std PSM with frame aggregation compared to UAPSD and frame aggregation.

Another reason for expecting that the Std PSM should have worse performance than UAPSD because the high collision probability that results from synchronizing all STAs after the Beacon. However, to our surprise we found out that this synchronization does not always result in collisions because of the power capture effect. From the results in Figure [5.2] shown above, it can be seen that the capture effect is significant in Std PSM and its because of the high probability of more than one station trying to transmit at the same time. With distributed channel access as in UAPSD, the capture effect is less.

It can also be seen in the Figure [5.1] that the average power consumption of the Voice stations using UAPSD with aggregation crosses the one of Std PSM with aggregation at a comparatively lower number of stations than the video stations. This is because in UAPSD the trigger frames are sent also during the silence periods of the voice stations unlike in Std PSM and it results in a comparatively more signaling.

5.2. Non Real-Time applications

The Figure [5.3] consists of the performance results of the non real-time applications FTP and Web. Note that the FTP and Web stations use the UAPSD Continuous mode as explained in section 3.2.

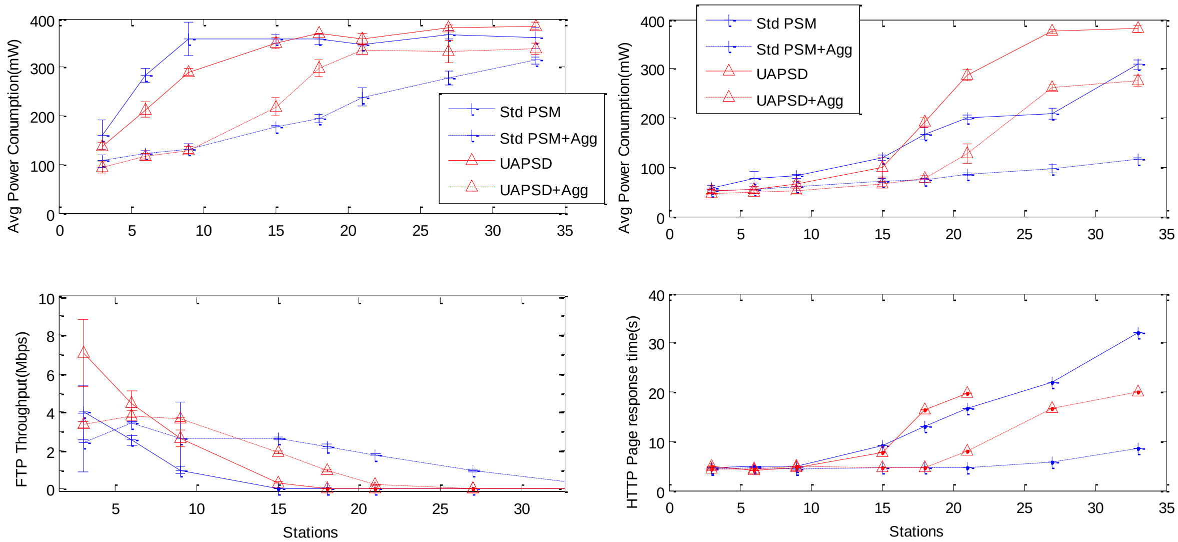

Figure [5.3]: Performance results of FTP and Web applications

Without frame aggregation

Power consumption: Without frame aggregation and when there is less congestion in the network, the average power consumption is lower using UAPSD continuous mode than Std PSM because of the single trigger frame in UAPSD compared to the one PS-Poll per frame in Std PSM. Although, the gain in power saving is not significant because of the access priorities in UAPSD where the FTP and Web applications belongs to the lower priority group compared to Voice and Video applications whereas in Std PSM, all the applications has equal priority to get access to the channel for transmission of signaling frames (PS-Poll) and to retrieve the frames buffered at the AP.

The average power consumption of FTP stations are more compared to the Web stations because of the download file size of 20MB in FTP stations compared to the page size of 10KB in Web stations. The FTP stations have to stay more time in the Idle power state waiting for the frames from the AP. When the number of stations is increased in the network, the average power consumption of the FTP stations saturates almost to the value of the Idle state power consumption of about 390mW and this means that the FTP stations are waiting almost all the time in the Idle state to receive the frames buffered at the AP because of the high priority Voice frames and Video frames getting through the channel utilizing the whole channel capacity..

When the network gets congested, the average power consumption of Web stations increases rapidly because the stations waiting in the active state for the page download gets longer It can be seen that the performance of Std PSM with frame aggregation is much better then UAPSD with frame aggregation when the network is congested. This is because of the fact that the Std PSM sends the PS-Poll with the same access priority as real-time applications and also because there is less signaling from high priority applications as seen in Figure [5.2].

HTTP Page response time: It can be seen from the page response time results in Figure [5.3] that the time taken to download a web page increases when the number of stations in the network increases. This is because there are many high priority Voice and Video stations after the Beacon accessing the channel that the Web stations have to wait a long time awake in the Idle state before exchanging their frames. Note that this could happen up to a certain degree and still the web page download time will be the same.

FTP Throughput: When the network is not congested (less than 5 stations) the throughput is higher with UAPSD than with Std PSM because PS-Poll per frame is not sent in UAPSD as in Std PSM. When the network congestion increases, the high priority applications utilize the entire channel capacity and the FTP throughput reduces to almost zero as seen in Figure [5.3] which indicates that the FTP frames buffered at the AP are almost not delivered to the FTP station.

With frame aggregation

Power consumption: As can be seen from the results in Figure [5.3], the average power consumption is less when frame aggregation is used. This is because of the efficient use of frame aggregation thereby reducing contention and the overhead in sending PS-Poll’s per frame (only in Std PSM) and ACK compared to the non-aggregated case. The power consumption of Std PSM is much lower than UAPSD when the network is congested as it gets advantage of sending PS-Poll’s with the same access priority as the other high priority applications to retrieve the frames buffered at the AP.

HTTP Page response time: As explained before frame aggregation works efficiently under congested network conditions and hence the page response time is lower for Web applications as seen in Figure [5.3]. Again, Std PSM has lower page response time than UAPSD because of the same access priority of PS-Poll’s for all application categories as explained before.

FTP Throughput: With only few stations in the network, frame aggregation does not give much throughput for the FTP applications which was not an expected behaviour and it was because of the effective Round Trip Time (RTT) increase in the TCP connection. This behaviour is explained in the Figure [5.4] where the frame exchange sequence between a station using the FTP application and the *FTP server* providing the TCP traffic is shown. As can be seen in the figure, when a TCP window of data arrives at the AP, it buffers the frames since the station is in the sleep state at that moment. During the Beacon arrival time, the station wakes up and starts receiving the frames buffered at the AP. If frame aggregation is used, the station downloads all the frames buffered at the AP efficiently using the frame aggregation, sends back the acknowledgement and then goes to the sleep state. When the TCP Acknowledgement reaches back the FTP server, it sends the next window of TCP data frames that arrives at the AP after the RTT. But the AP has to buffer these frames again as the station is already in the sleep state and it can deliver the frames only when the station wakes up at the next beacon interval. The FTP server does not sends a new window of TCP data as long as it does not receive back the acknowledgement for the previous window of data and this means that the effective RTT is increased for the duration equal to the Beacon inter-arrival period.

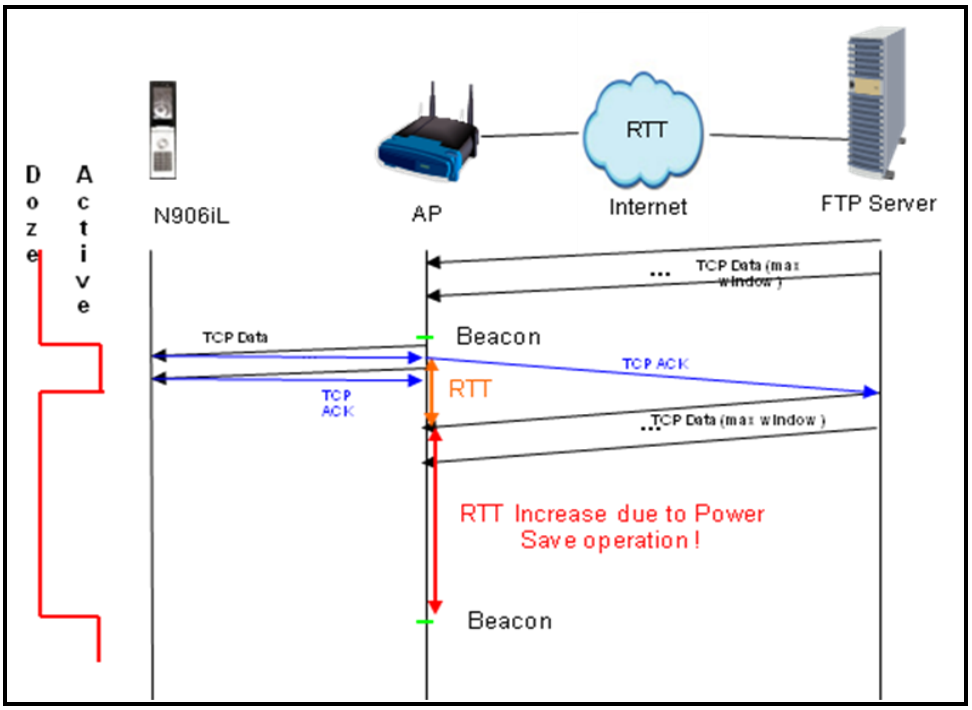

Figure [5.4]: Effective RTT increase for TCP applications

When frame aggregation is not used, the packets are downloaded one by one and the station can generate a TCP ACK as soon as it receives the first TCP data frame. The stations remains in the active state as long as the AP buffer is not empty and there is more probability that when the next window of packets arrive from the FTP server the stations are still awake and frames are delivered continuously making use of the WLAN bandwidth efficiently. This behaviour can be seen clearly from the statistics of the AP Power Save Buffer in Figure [5.5]. The statistics shows that without using frame aggregation, there is continuous incoming and outgoing traffic in the buffer, but with frame aggregation the traffic in the buffer remains constant most of the time except at Beacon intervals.

For example consider the buffer statistics when frame aggregation is used. At time 40.1s, all the packets in the buffer are emptied and this happens when all the packets are and delivered to the station in a single aggregated frame. Now the station finds that there is no more data buffered at the AP and hence it goes to the sleep state. After RTT delay, the new frames arrive at about 40.15s and they are buffered at the AP until the next Beacon interval at 40.2s. Since the FTP server did not get the TCP acknowledgement for the last window of data, it does not send any more data as can be seen from the statistics that the buffer contents remains constant from 40.15s to 40.2s. On the other hand, without frame aggregation it can be seen that the frames are delivered continuously to the station and also the buffer is always occupied with frames from the FTP server as it receives back the TCP Acknowledgements at regular intervals.

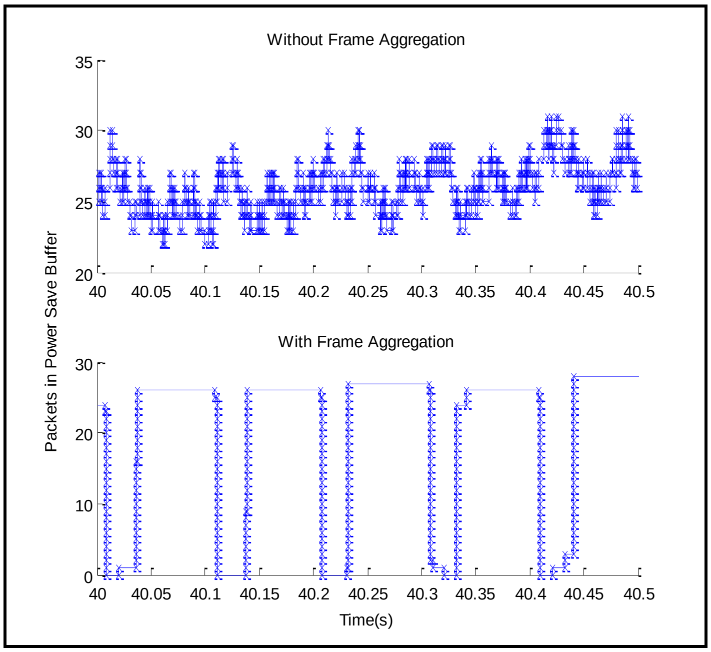

Figure [5.5]: Power Save Buffer Statistics

5.3. Summary of results

From the above discussions, it can be concluded that

- Aggregating frames gives better performance than without aggregation. The only exception is for TCP throughput when the network is not congested and it could happen that the throughput reduces because of effective increase in RTT.

- When the network is not congested, UAPSD with frame aggregation gives better performance and meets the QoS requirements of the real-time applications.

- When the network congestion increases, the Std PSM with frame aggregation gives better performance by buffering and efficient utilization of the frame aggregation feature. But UAPSD has a degraded performance because of the increase in signaling.

- From the above conclusions it is clear that a solution is required that provides the benefits of UAPSD when the network is not congested, and of StdPSM+agg when the network is congested.

6. An adaptive solution for congestion control and efficient frame aggregation in UAPSD

From the previous discussions we have found that the UAPSD with frame aggregation provides better performance when the network is not congested with only few stations in the network, but when the network gets congested with more stations, the performance results are not better because of the increase in the signaling load and the frame aggregation feature could not be utilized efficiently. Hence we propose here a method for dynamic congestion control and efficient frame aggregation for real-time applications depending on the network conditions. In this section, we discuss the proposed adaptive algorithm architecture, its functionality and performance analysis. The basic idea of the algorithm is selecting the optimum aggregation interval according to the level of congestion in the network and the algorithm is run by every station independently in a distributed way. The algorithm is standard compliant since 802.11n does not define how a device has to define its aggregations.

6.1. Algorithm Architecture

To control the network congestion dynamically, the stations perform a dynamic frame aggregation depending on the network conditions following the rules given below:

1) When the network congestion increases, the amount of aggregation is increased to reduce the level of congestion in the network.

2) When the network congestion decreases, the aggregation interval is reduced in order to minimize delay and utilizing the channel bandwidth efficiently.

The network congestion level is determined using the two *Exponentially Weighted Moving Average (EWMA)* values given below: (1) Average access delay and (2) Average SP duration

$$\text{Average access delay} = \alpha \times \text{Average access delay} + (1-\alpha) \times \text{Last access delay}$$

$$\text{Average SP duration} = \alpha \times \text{Average SP duration} + (1-\alpha) \times \text{Last SP duration}$$

Where α is a constant, and $0 \leq \alpha \leq 1$(default value of 0.8).

The *last access delay* is the difference between the time at which the packet arrives in the transmission queue from the higher layer for transmission and the time at which the packet is actually transmitted in to the air by the station. So the last access delay increases when the uplink congestion increases because of channel contention and hence the Average access delay measures the congestion level in the uplink

The *last SP duration* is the effective SP duration measured at the station i.e. the difference between the time at which a SP is started at by sending a trigger frame and the time at which the SP ends (EOSP bit set). Note that the SP is configured to '*All*' frames, and hence the effective SP duration can vary according to the congestion due to queuing in the AP in downlink. Hence the Average SP duration gives a measurement of the downlink congestion level in the network.

Note that the stations in power saving mode does not have to be awake for an additional time than it would normally do in order to execute these measurements of network congestion level. Once the network congestion level is measured, according to the level of congestion in the network the amount of aggregation is controlled by changing the Service Interval of the trigger frames in the uplink. By doing this the congestion level in the network can be reduced.

The Adaptive algorithm architecture is shown in Figure [6.1] below:

Variables used in the algorithm:

Teval: Period at which the station evaluates the network congestion level

Service Interval (SI): Time period of the transmission of the trigger frames

Minimum Service Interval (mSI): The starting default value of the SI

Maximum delay bound (D): Maximum allowed increase in the SI which corresponds to the maximum allowed downlink delay for an application

Up_count, Down_count: Variables to count the number of Teval times the network congestion level is above or below the upper and lower threshold levels respectively

Up_count _limit: Maximum number of attempts for which the station finds that the network congestion level is above the predefined upper threshold

Down_count_limit: Maximum number of attempts the STA finds that the network congestion level is below than the predefined lower threshold

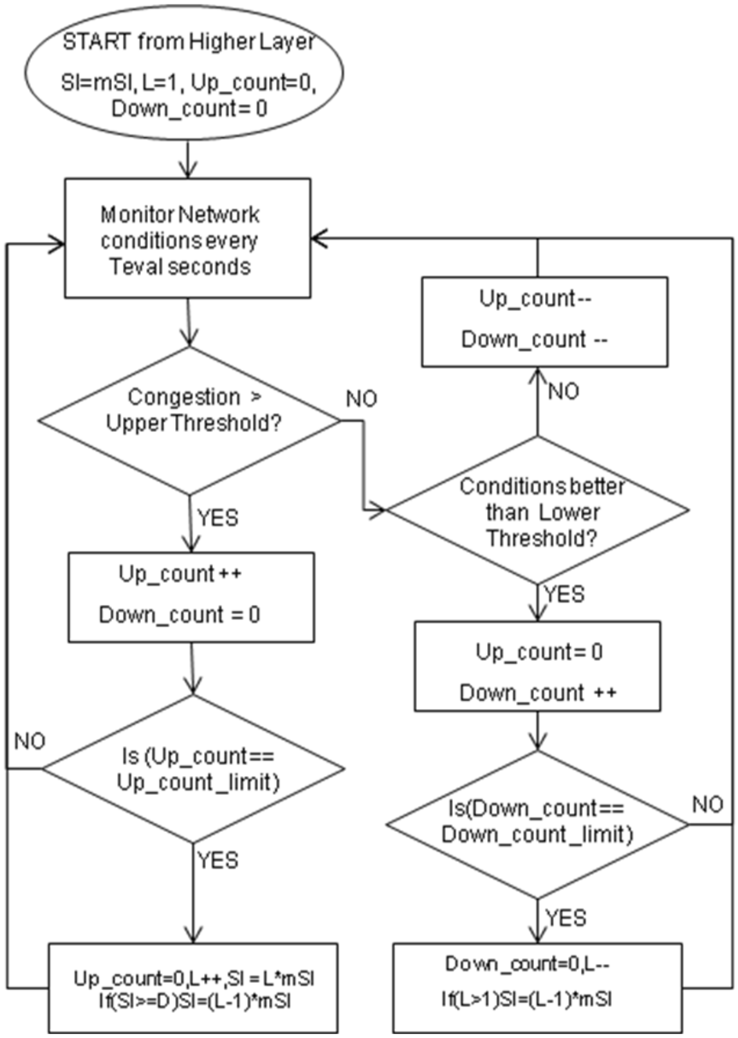

Figure [6.1]: Adaptive algorithm architecture

The Adaptive algorithm operates in the following manner:

1. The algorithm is triggered from the higher layer when a multimedia session starts (e.g. when a voice call starts)

2. The higher layer provides two parameters (i) mSI, usually the packet generation interval, (e.g. 20ms for G.711) and (ii) D (e.g. 100ms for voice call). The algorithm starts considering a SI equal to the mSI and this means a minimum delay in the beginning.

3. At monitoring times, i.e. tj = T_init + j*T_eval, the station evaluates the network conditions. Where T_init is the initial time offset after which the monitoring is started, T_eval is the evaluation period, the period at which the monitoring is repeated and j is a positive integer.

4. If network congestion level is high, determined by the upper threshold which is by default configured to a value of (1/3)*SI, the station increases the value of Up_count and when its value reaches the threshold value of Up_count_limit, the station increases its operating SI. This memory is introduced in order not to be affected by spurious increases in the measured congestion level. Increasing the SI will reduce the level of congestion by reducing the number of trigger frames. This is repeated until the congestion level falls below the upper threshold or if the selected SI has already reached the maximum allowed D, then the station then stops trying to increase the SI instead operates with the same SI at that moment.

5. When stations leave the network, the congestion level reduces and network conditions gets better, determined by the lower threshold which is by default configured to a value of (1/6)*SI, the value of Down_count is increased and when the Down_count reaches the down count limit, the current SI is reduced to the last operated SI. For the lower threshold, value much lower than the upper threshold is chosen in order to provide a hysteresis between the upper and lower threshold so as to avoid toggling in SI. The SI lowering process is continued as long as the congestion level is below the lower threshold limit or if the selected SI has already reached mSI.

6.2. Algorithm operation

To study the behaviour and to evaluate the performance of the algorithm, the simulations are initially performed in the same *hot-spot* scenario but with only voice stations and with the same configurations as in the previous experiment. The congestion in the network is increased by adding more stations to the scenario in the run time to study the behaviour of the dynamic behaviour of the adaptive algorithm.

To validate the functionality of the algorithm, the statistics of Average SP duration, Average access delay and SI of the stations are collected during the simulations. Figure [6.2] shows the change in the SI corresponding to the variation in the Average SP as observed in a random single station present in the network. The x-axis shows the simulation time instant at which the corresponding values of SI and the Average SP are measured. Note that the Average access delay is not shown in the graph because the measured values of Average access delay does not exceeded the threshold level’s to cause a change in SI.

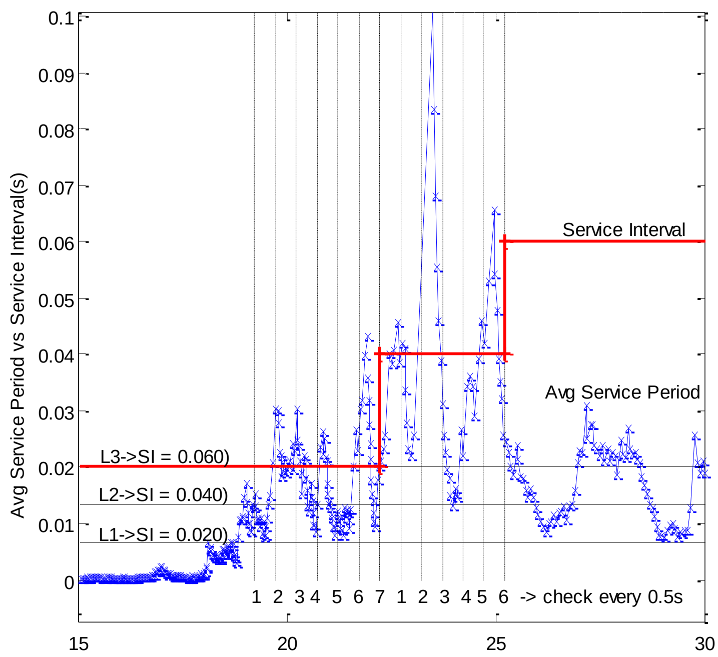

Figure [6.2]: Adaptive algorithm operation

It can be seen in the Figure [6.2] that the Average SP increases from 15s onwards and this is because of more stations joining the network and resulting in increase of congestion level. When the Average SP value reaches the level L1 as shown in the figure, which is 1/3 of the current SI of 20 ms and this indicates a congested network condition. Then the Up_count value is incremented by 1 and is repeated at every moitoring time instant, tj. When the Up_count value reaches the Up_count_limit, which is configured to 7, the SI is increased to 40 ms and now the trigger frames are sent every 40ms instead of 20ms.

After the SI is increased it can be seen that Average SP is still above the threshold level L2 that corresponds to the SI of 40 ms and this indicates that the network congestion is above the acceptable limit. The Up_count value is incremented again in steps by one during every monitoring time instant and when the value reaches the Up_count_limit of 6, the SI is again increased to 60 ms. The reason for dynamically reconfiguring the Up_count_limit to a random value every time is explained later in this section.

After the SI is increased to 60ms, it can be seen that the the Average SP remains below the threshold level L3 corresponding to the SI of 60ms and hence the SI is not increased further. The short oscillations seen in the Average SP before settling down does not change the SI because the algorithm only changes the SI if the network congestion level stays stable for a minimum duration depending on the configuration of Teval and the Up_count_limit/Down_count_limit values. Also note that frequent oscillations in the aggregation interval would not be desirable as it could introduce jitter in the application.

Our intention is that the aggregation interval is increased only in a minimum number of stations required to pull congestion down to an acceptable level. But since the level of downlink congestion seen by all stations is correlated, basically because the packets of all stations go through the same queue in the AP, there is possibility that the Average SP measured by the individual stations will be correlated and this could result in more number of stations change their SIs at the same time if the Up_count_limit and Down_count_limit are configured to the same value across all the stations in the network. This can lead to oscillations in the Average Service Period and toggling of the SI's. This behaviour is explained using the Figure [6.3] given below.

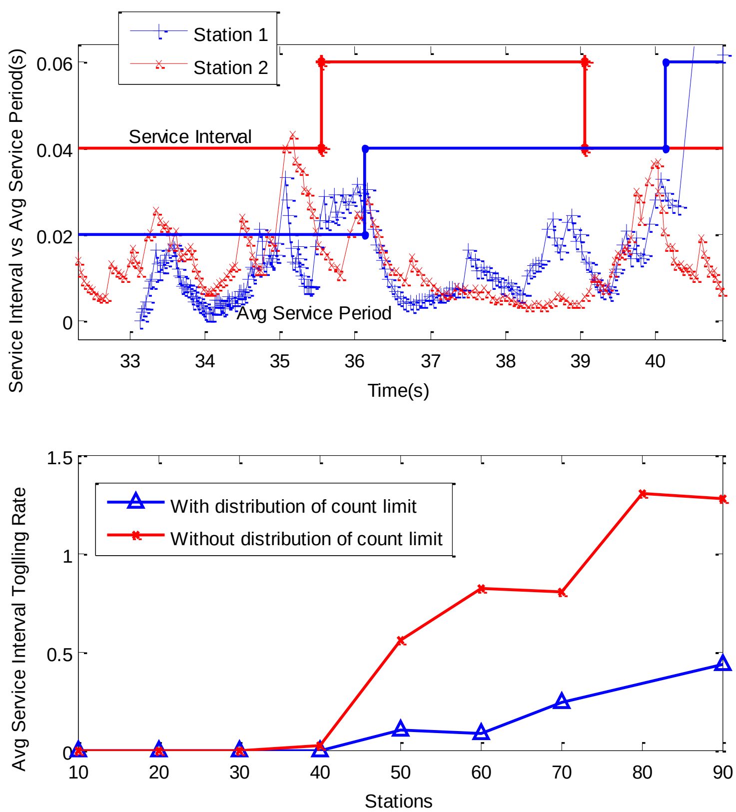

Figure [6.3]: Toggling of Service Intervals

The Figure [6.3] above shows the Average Service Period and the corresponding change in SI's measured from two random stations in the network, Station 1 and Station 2. It can be seen that the SI's of the two stations increases almost at the same time because of correlation in the Average Service Period values. When more stations increase their SI's at the same time, the Average Service Period goes lower than the lower threshold value of network congestion level and the stations reduce its SI in order to reduce the uplink delay and downlink delay (as well because the uplink frames are triggers for the downlink frames). This toggling of SI could cause jitter in the uplink delay of the frames delivered. To avoid the toggling of SI, the Up_count_limit and Down_count_limit is randomly distributed across different stations so that the probability of stations changing the SI at the same time is reduced. Since the stations with a lower value of Up_count_limit and Down_count_limit changes its SI more quicker than the stations with a higher value for Up_count_limit and Down_count_limit, the random value of Up_count_limit and Down_count_limit gets a new value every time the count is restarted so as to provide fairness to all the stations.

In order to prove the design assumption we show in Figure [6.3] parameter *Average Service Interval Toggling Rate* which gives a measure to find the impact of distributing the count thresholds on the toggling of the SI. The number of toggling in the SI is measured i.e. the number of times a station visits the old SI after switching to a new value. The average of this value is calculated from all the stations in the scenario to find the Average Service Interval Toggling Rate. It can be seen from the results that the Average Service Interval Toggling Rate is much higher when the count limits are configured to a constant value compared to the case where the count limit is randomly distributed across the stations.

6.3. Performance evaluation in ideal scenario

As a proof the concept, the performance metrics like average power consumption, average retransmissions, delay and throughput are measured in a scenario consisting only of voice stations. The simulations are also performed with different EDCA settings to study the impact of the EDCA settings on the performance when using the adaptive algorithm. Our expectation would be that, smaller EDCA settings result in higher collision probabilities and would hence benefit more from the algorithm that reduces congestion in the network. Note that in this section, the different EDCA settings are represented in the format EDCA <AIFSN, CWmin, CWmax>.

Figure [6.4] shows the average power consumption where it can be seen that, when the network congestion increases with more than 50 stations, the Adaptive algorithm maintains the average power consumption at the same level but at the same time the average power consumption increases without using the algorithm. When the network congestion level increases, the SI’s of the trigger frames are increased by the Adaptive algorithm to reduce the congestion in the network and by the efficient use of frame aggregation the frame exchange happens with less contention. In other words, the congestion level in the network is kept constant by the algorithm irrespective of the number of stations, but at the cost of the increase in the uplink delay which in turn also increases the downlink delay because the frames in the uplink is the trigger for downlink traffic.

The average number of re-transmissions in Figure [6.4] is computed by taking the average of the number of frame retransmissions because of collisions from all of the stations present in the network. It can be seen that the average number of retransmissions is comparatively less when the adaptive algorithm is used and this means a less number of collisions because of reduced congestion level in the network. Note that using the algorithm is especially more beneficial with stronger EDCA settings in which case there is more probability of collisions.

With relatively stronger EDCA settings (EDCA<2,3,7>) the contention is more. Hence the average power consumption and average number of retransmissions are more as expected. But when the Adaptive algorithm is used, the congestion level in the network is controlled dynamically as explained before and hence the different EDCA settings does not have an impact on the average power consumption.

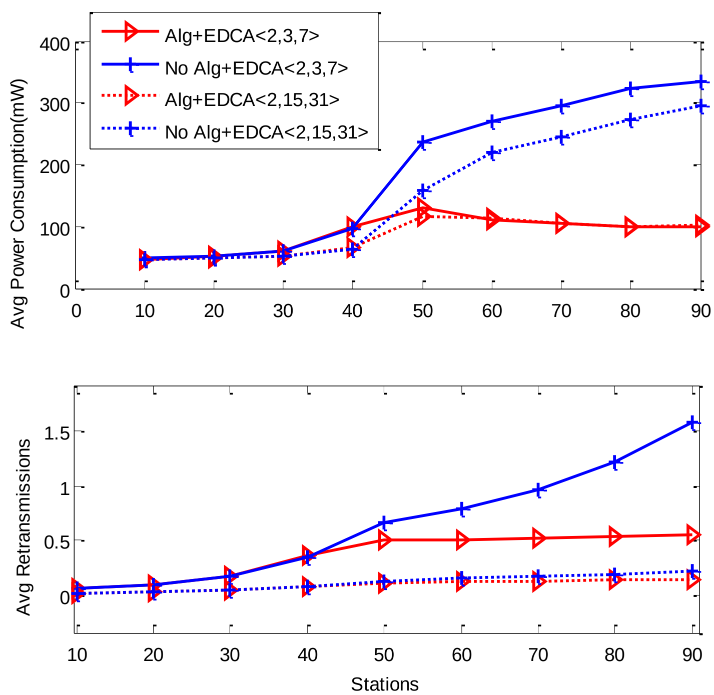

Figure [6.4]: Average Power Consumption and Retransmissions

The Figure [6.5] shows that the downlink delay is comparatively lower when using the adaptive algorithm and also the different EDCA settings do not have much impact on the uplink and downlink delays when algorithm is used. When using a relatively weaker EDCA settings (EDCA<2,15,31>), the uplink delay is a bit higher when algorithm is used and this is because of the buffering of the frames to change the aggregation interval for congestion control in the network. However notice that the benefits in downlink justify the increased delay in the uplink. For example, basically with the downlink delay without algorithm the voice application does not meet its QoS requirements.

When relatively stronger EDCA settings (EDCA<2,3,7>) are used, the contention between the stations to gain the access to the channel increases and the number of collisions also increases. When an aggregated A-MPDU packet collide, only the collided packets in the aggregated A-MPDU packet that are not successfully received at the receiving station are retransmitted. Since 802.11n guarantees in order delivery of the packets, the successful packets have to wait for the failed ones before being delivered to the higher layers and this waiting time increases the overall packet delay. This is the reason why the uplink delay increases rapidly with stronger EDCA settings when the Adaptive algorithm is not used as seen in Figure [6.5]. It is also clear that the different EDCA settings do not have an impact on the uplink and downlink delays.

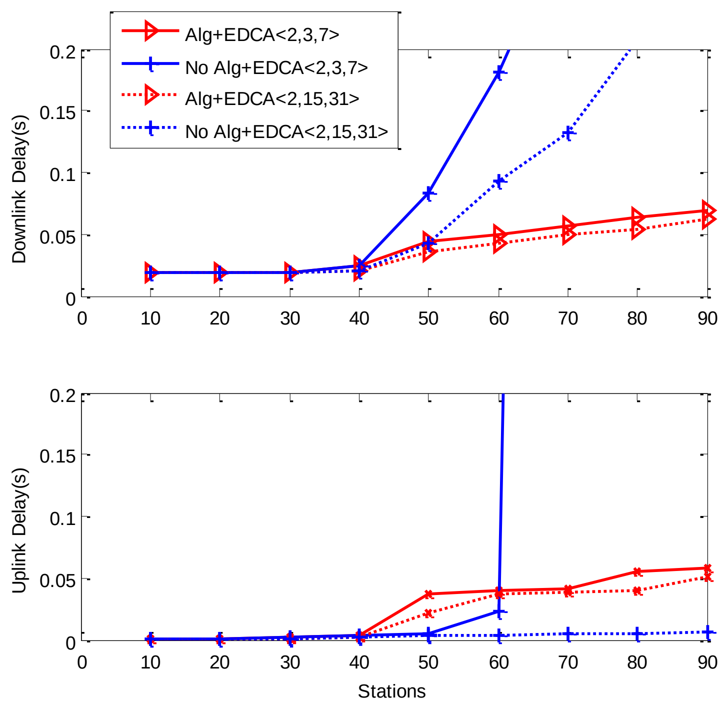

Figure [6.5]: Uplink and Downlink delay

Figure [6.6] shows the average uplink and downlink throughput where it can be seen that with relatively strong EDCA settings, the uplink throughput is maintained by using the adaptive algorithm. On the other hand, without using the algorithm the throughput drops down with more congestion in the network with stronger EDCA settings.

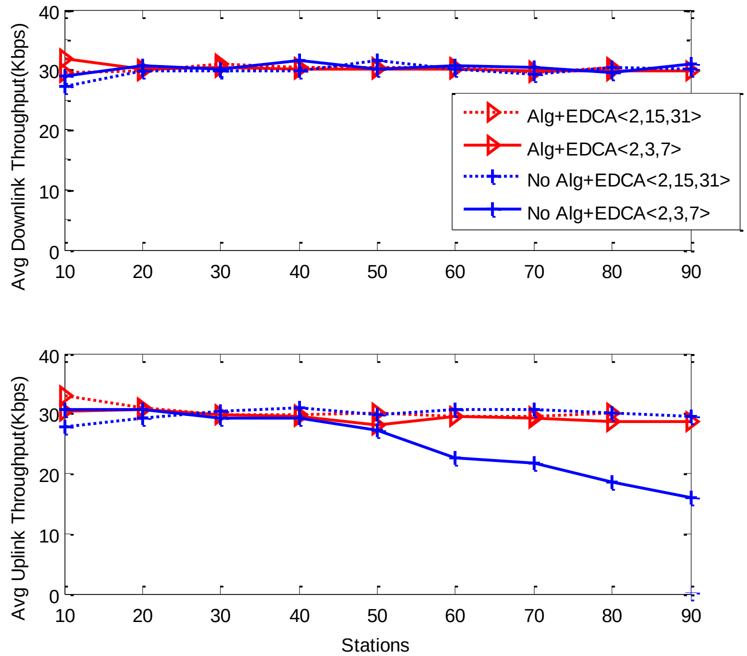

Figure [6.6]: Uplink and Downlink Throughput

The *Average Service Interval* shown in Figure[6.7] computed from the average of the SI's used by all the stations at different time instants is plotted to study the impact on the change in the SI's with respect to different EDCA settings. As can be seen from the results, the SI starts changing with above 50 stations, when the network gets congested. Also, it can be seen that the value of Average Service Interval corresponding to stronger EDCA settings (EDCA<2,3,7>) is higher compared to relatively weaker EDCA settings (EDCA<2,15,31>). This is because of the stronger contention with stronger EDCA settings that results in a more congestion level in the network compared to using relatively weaker EDCA settings.

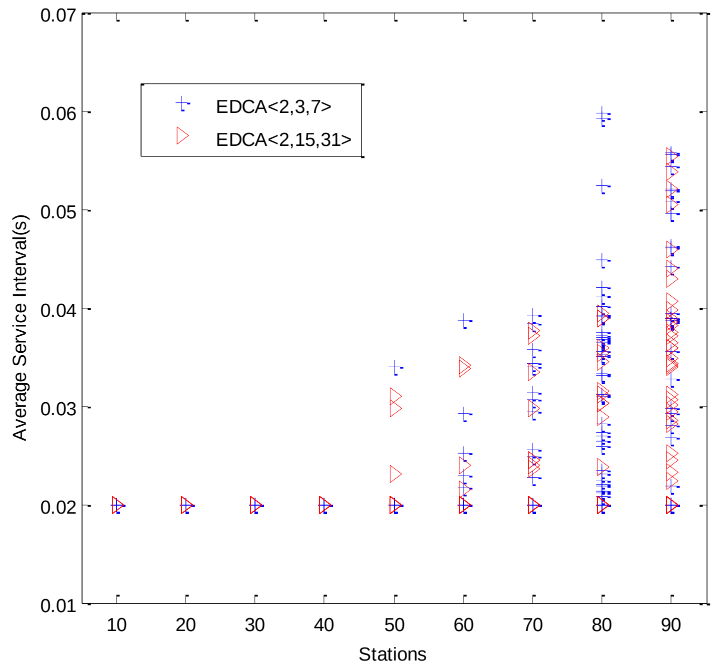

Figure [6.7]: Average Service Interval

6.4. Performance evaluation in real scenario

Finally, the simulations are performed in a scenario consisting of all access categories (same scenario used in the simulations in Chapter 5) to evaluate the performance of the adaptive algorithm in a real scenario and to compare with the performance results of Std PSM and UAPSD, when frame aggregation is used. Note that only the real-time applications use the adaptive algorithm and the non real-time applications use the UAPSD Continuous mode.

Real-time applications

The Figure [6.8] below shows the average power consumption and downlink delay performance results of Voice and Video stations for all the three cases - Std PSM with aggregation (Std PSM+Agg), UAPSD with frame aggregation (UAPSD+Agg) and UAPSD with adaptive algorithm and frame aggregation (UAPSD+Agg+Alg). From the results, it is clear that by using UAPSD with the Adaptive algorithm gives much better performance in both QoS and power consumption under congested network conditions.

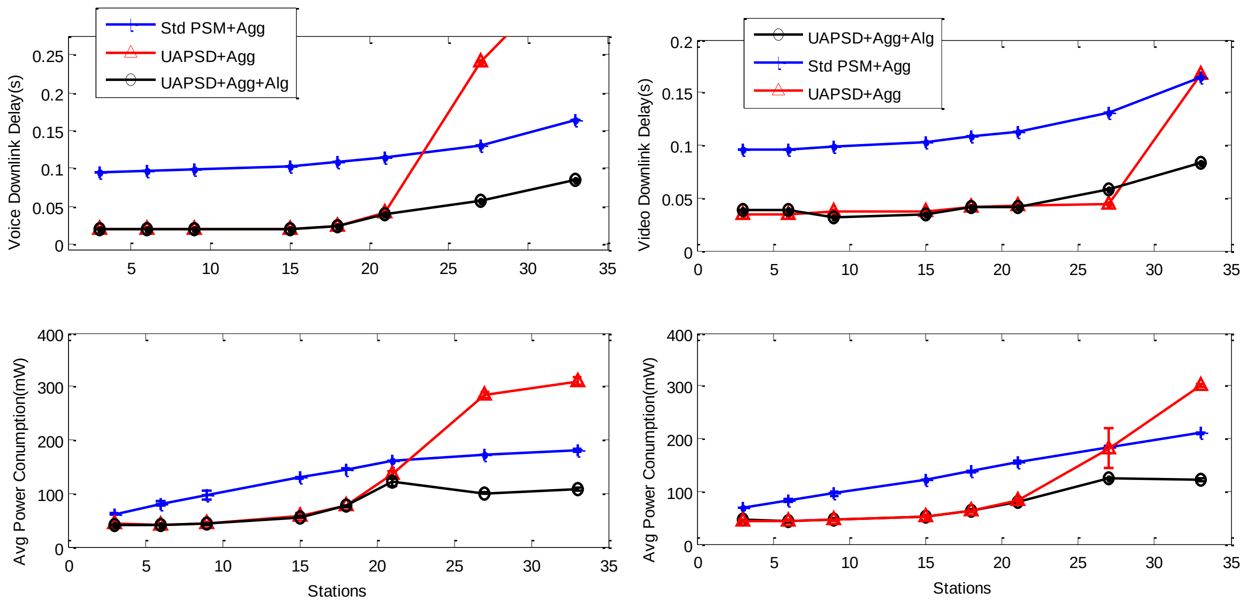

Figure [6.8]: Adaptive algorithm performance with real-time applications

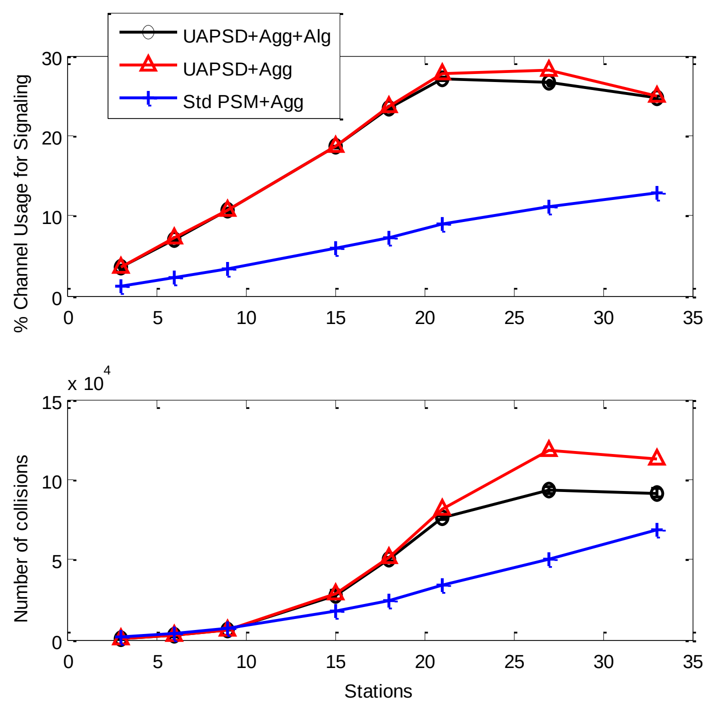

Figure [6.9]: Adaptive algorithm performance - Signaling load and collisions

The Adaptive algorithm controls the congestion level in the network dynamically so that the the network capacity is fully utilized to provide maximum QoS to the applications. When the network congestion level increases, only some of the stations in the network changes its SI to reduce the congestion level to the maximum acceptable level and hence there is not much significant change in the signaling load can be observed in the results shown in Figure [6.9]. The number of collisions are reduced with the use of the algorithm and this means that the congestion in the network is reduced using the algorithm. Also the average access delay will be less as the congestion will be reduced by using the algorithm.

The Figure [6.10] below shows the average SI used by the Voice and Video applications. The number of stations on the x-axis includes all the stations in the network including all access categories. It can be seen that when the network is not congested the average SI of voice is 20ms and video is 40ms which is the minimum SI. It can be seen that when the number of stations in the network increases the network congestion level also increases and the average SI used by both the applications starts increasing.

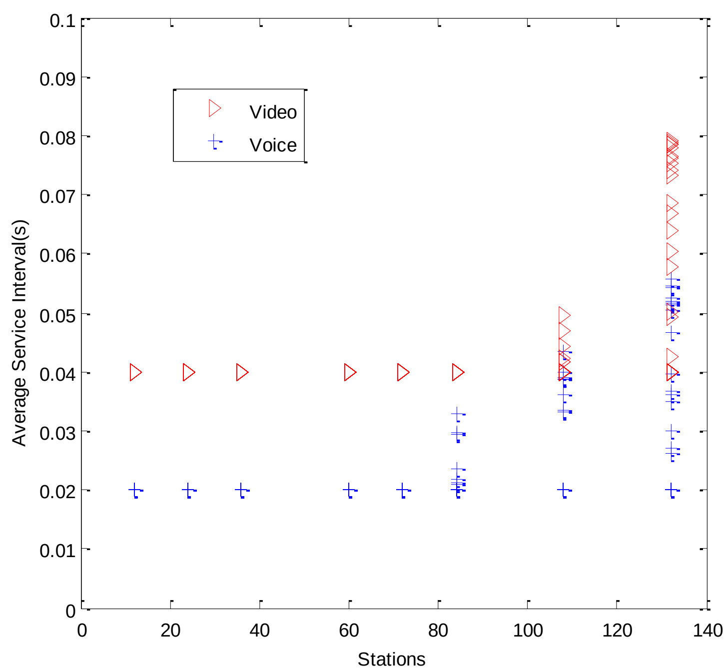

Figure [6.10]: Average Service Intervals used by the real-time applications

Non real-time applications

The Figure [6.11] below shows the performance result of the non-real time applications FTP and Web. The FTP and Web applications does not use the adaptive algorithm, but in this case UAPSD+Agg+Alg represents that these stations are present in the scenario consisting of the stations servicing the real-time applications using the adaptive algorithm. The performance results are almost similar to the results from first experiment and this is because the FTP and

Web applications 'collapse' much before the high priority stations start making use of the algorithm. In other words, before the algorithm starts functioning in the real-time applications, the congestion in the network due to the higher priority applications would have made the non real-time applications in a condition where it has almost stopped transfer of frames due to high levels of packet delays and losses.

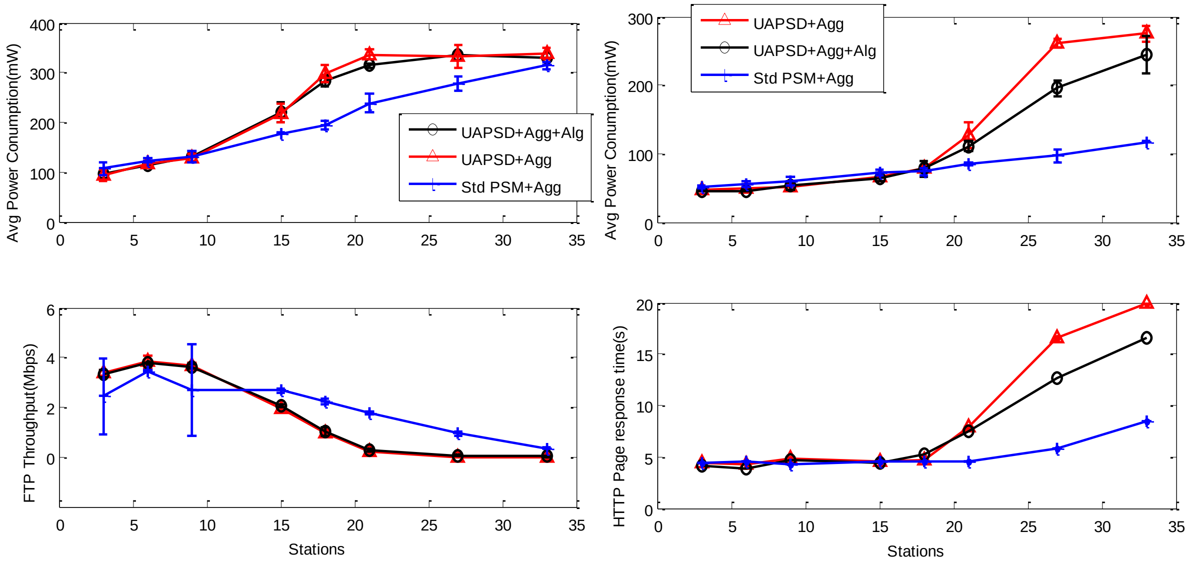

Figure [6.11]: Adaptive algorithm effect on non real-time applications

6.5. Summary of results

From the previous discussions, it can be concluded that:

- Under congested network conditions, the UAPSD with the Adaptive algorithm gives better performance by dynamic congestion control, efficiently utilizing the frame aggregation.
- When using relatively weaker EDCA settings, the uplink delay will be more with the adaptive algorithm because of buffering of the frames in the stations for dynamic aggregation. But with stronger EDCA settings, the uplink delay is better when the algorithm is used.
- Because of the dynamic congestion control by the Adaptive algorithm, the different EDCA settings do not have much impact on the power consumption and delay.
- The performance of non real-time applications is not affected when using the adaptive algorithm for the real-time applications.

7. Summary

7.1. Conclusion

A performance analysis of the power saving modes Std PSM and UAPSD has been performed using as basis OPNET's 802.11n simulation model. Based on the performance results it was possible to study the behaviour of the power saving modes in the IEEE 802.11n standard. The analysis focused first on the study of the impact of the frame aggregation feature introduced in the 802.11n standard in combination with different power saving modes. From the analysis results it can be concluded that

- Frame aggregation reduces the overhead in transmission of acknowledgement frames and trigger frames (PS-Poll in Std PSM and QoS Null in UAPSD). Hence, contention in the network is reduced and better power saving and QoS is achieved as compared to the case of not using frame aggregation.

- When the network is not congested, using frame aggregation results in FTP applications having an effective increase in RTT. The reason is that when aggregation is used frames are delivered faster and stations go to sleep before the next windows of packets arrive from the Ethernet server. Stations then wake up at the next Beacon arrival only and hence the throughput is less as compared to the case when frame aggregation is not used.

- When the network is not congested, UAPSD provides better QoS to real-time applications by providing a differential treatment to the different application categories. Also, UAPSD provides better power saving by providing a distributed access to the channel to the stations and hence, reduces the contention among the stations to get access to the channel. This is different to the Std PSM case where all the stations contend for the channel at the same time immediately after the Beacon.

- When the network is congested, Std PSM with frame aggregation is efficient because buffering and aggregating the frames significantly reduces the signaling overhead unlike in UAPSD where trigger frames are sent every Service Interval which increases the signaling load with more stations causing the poor utilization of the channel bandwidth for data exchange.

From the above conclusions, it is clear that the frame aggregation will perform better when the network is congested and UAPSD does not utilize the frame aggregation efficiently because of the network congestion caused by its signaling overhead. Therefore, we designed an adaptive algorithm for UAPSD which, based on the level of congestion in the network, reduces the amount of signaling and provides better QoS and power saving utilizing the frame aggregation efficiently.

The adaptive algorithm is initially evaluated under an ideal scenario consisting only of Voice stations to study its functionality. Simulations were performed with different EDCA settings to study the effect of EDCA settings on the performance of the algorithm. Once the functionality and performance of the algorithm was analyzed and verified, simulations were performed in a realistic scenario consisting of stations serving applications of all access categories types and compared with the performance results from the first experiment with Std PSM and UAPSD. From the results of the simulations it can be concluded that:

- The adaptive algorithm controls the level of congestion in the network by reducing the signaling in the network and hence it provides better power saving and QoS for real-time applications.

- Using different EDCA settings does not affect the performance as the adaptive algorithm dynamically reduces the congestion level by varying the amount of frame aggregation and thereby reducing contention.

- The performance of non real-time applications is not affected by using the adaptive algorithm for real-time applications.

7.2. Future work

The following areas are still need to be explored to further optimize the algorithm and make it suitable for all types of application categories:

- The Service Interval toggling rate in the adaptive algorithm can be optimized further and it can be done by reducing the monitoring time period and distributing the count limit values over a wide range across the different stations. In our experiment, the monitoring time period of the network conditions was configured to a constant high value and the count limit values were not distributed over a wide range.

- The value of the constant α, used to calculate the EWMA values could also be distributed across different stations instead of the count limits to optimize the Service Interval toggling rate

- The performance of the algorithm also has to be evaluated when stations leave the network, to determine how the algorithm adapts to the real dynamic network conditions.

- The Adaptive algorithm is only suitable for real-time applications, but for improving the performance of non real-time applications a better mechanism still has to be developed. A possible method could be sending trigger frames with a minimum Service Interval after the file transfer is started and increase the Service Interval on every QoS-Null received back which indicates there are no frames buffered at the AP. The Service Interval is increased and the trigger frames are sent as long as the Service Interval reaches the Beacon arrival interval and then the stations listens only to the Beacons.

- Another possible solution for non real-time applications could be to increase the aggregation interval of higher priority traffic based not on the congestion experienced by higher priority traffic, but on the congestion experienced by lower priority traffic. This would prevent the collapsing of the non real-time applications because of the higher priority applications.

List of figures

List of tables

Bibliography

[1] IEEE 802.11 WG, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEEStandard 802.11-2007, June 2007

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

[3] Thomas Paul, Tokunbo Ogunfunmi. Understanding the 802.11n Amendment, IEEE Circuits and Systems Magazine, First Quarter 2008

[4] OPNET, Modeling Concepts Reference Manual, Modeler/Release 11.0

[5] IEEE P802.11, Wireless LAN's Usage Models, doc.: IEEE 802.11-03/802r23, May 2004

[6] "Broadcom 4311AG 802.11a/b/g," http://www.broadcom.com

[7] Xavier Pe´rez-Costa, Daniel Camps-Mur, Albert Vidal, On distributed power saving mechanisms of wireless LANs802.11e U-APSD vs 802.11 power save mode, Computer Networks - The International Journal of Computer and Telecommunications Networking Volume 51, Issue 9, June 2007

[8] OPNET Technologies, Inc.," http://www.opnet.com

[9] Xavier Pe´rez-Costa, Daniel Camps-Mur, NLE 802.11n Simulation Capabilities, NEC Europe

[10] Xiaolong Li, Qing-An Zeng, Capture Effect in the IEEE 802.11 WLANs with Rayleigh Fading, Shadowing and Path Loss, Wireless and Mobile Computing, Networking and Communications, 2006 (WiMobapos 2006). IEEE International Conference on Volume, Issue, 19-21 June 2006 Page(s):110 - 115

[11] IEEE P802.11n™/D7.0, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, September 2008