# 國立交通大學

## 資訊學院

## 資訊工程學系

## 博士論文

# IEEE 802.11 無線區域網路與 IEEE 802.16e 無線都會網路上省電協定之設計

# Design of Power Saving Protocols for IEEE 802.11 WLANs and IEEE 802.16e WMANs


研　究　生：黃世昌

指 導 教 授：簡榮宏教授


中 華 民 國 九十七 年 六 月

# IEEE 802.11 無線區域網路與 IEEE 802.16e 無線都會網路上省電協定之設計

研究生：黃世昌　　　　指導教授：簡榮宏 博士

國立交通大學　資訊工程系

## 摘　　要

隨著無線網路的蓬勃發展，越來越多無線存取技術在不同的應用層面與需求下被提出來。在眾多的無線技術當中，IEEE 802.11 無線區域網路與 IEEE 802.16 無線都會網路是兩個最直接影響到使用者存取網際網路的技術。前者提供一個簡單、廉價的方案讓使用者建構自己的無線區域網路。後者提供一個無線存取方案來替代目前有線網路中以電纜或 ADSL 存取網際網路的方式。由於在無線網路中大部份的行動裝置都是由電池供應電力，節省電源以延長的運作時間是一個非常重要的議題。因此，在此篇論文當中，我們分別針對 IEEE 802.11 無線區域網路的基礎建置模式(infrastructure mode)、隨意網路模式(ad hoc mode)、與 IEEE 802.16e 無線都會網路設計相應的省電機制。

對於 IEEE 802.11 無線區域網路的基礎建置模式，我們提出一個新方法來安排休眠主機(sleep stations)醒來的時間，所提出的方法企圖讓每一個信標區間(beacon interval)中醒來的主機數量得以更加平均。藉由這個方法來降低封包相撞的機率，讓主機避免重傳資料來節省電力。此外，我們也考慮控制所要通知之醒來主機的數量以及他們之間的存取順序，藉以避免過多醒來的主機在發現擷取點(Access Point)上

有暫存的資料之時，同時發送 PS-Poll 訊框存取網路而造成嚴重的封包衝撞。在設計上我們提供三種不同的選取機制來控制所要通知的主機數，通知機制為分別為只選一台主機、或是根據聯結識別碼(AID)、或預存資料長度來選取主機數。模擬的結果證實我們提的方法能有效的節省電力。

對於 IEEE 802.11 無線區域網路的隨意網路模式，我們提出一個能自我組態的省電方法，稱為 SCPS。每一台主機如同基礎建置模式一樣，可以選取自己的休眠時間長度。當有一台主機進入或離開省電模式時，SCPS 會讓其他在省電模式下的主機去調整它們醒來的排程，這樣的調整可以平衡在每一個信標區間裡醒來的主機數量，使得能源因傳輸媒介的競爭與衝撞而耗損的問題可以進一步的改善。模擬的結果顯示 SCPS 成功的平衡了每一個信標區間裡醒來的主機數量、增加休眠時間的百分比、同時也降低主機彼此之間封包衝撞的機率。

對於 IEEE 802.16e 無線寬頻網路，我們提出幾個能源效率高的排程方法，這些排程方法考慮的是多台行動主機(MSS)，而非像目前大多數的研究僅考量單一行動主機的排程方式，所考量的資料流都是有服務品質延遲限制的常數速率資料流，所提出的方法同時考量能源使用的效益與頻寬利用率。我們提出的方法分為兩類，一是定期的自主休眠週期(PASC)，另一則為定期的相同休眠週期(PUSC)。在 PASC 中，所有行動主機的資料流所限定的服務品質限制直到它們加入前都未知，每一個行動主機都使用自己的休眠週期來節省電力。若是每一個行動主機的服務品質限制可以事先得知，則可以用 PUSC 方式讓行動主機的排程更有效率，因為在 PUSC 中所有的行動主機都使用相同的休眠週期，基地台可以很簡單的安排行動主機醒來的時間。模擬結果顯示 PUSC 與 PASC 皆有不錯的省電效能，特別是在 VOIP 的應用下，PUSC 還能讓頻寬利用率展現出更好的結果。

# Design of Power Saving Protocols for IEEE 802.11 WLANs and IEEE 802.16e WMANs

Student：Shih-Chang Huang Advisor：Dr. Rong-Hong Jan

Department of Computer Science,
National Chiao Tung University

## Abstract

Recently, wireless networks have widely developed. Many different wireless technologies have been proposed for different network aspects. Among all of the wireless technologies, the IEEE 802.11 WLANs and the IEEE 802.16 WMAN directly influence the internet access of end users. The former provides easy and low cost solution for people to build their own local area networks and the latter provides a wireless internet access solution to substitute the last-mile internet accessing in the wired networks. Because most of the mobile devices in the wireless networks are powered by battery, saving power to extend the operation time is a critical issue. Therefore, we design several power saving protocols for the infrastructure and ad hoc mode of IEEE 802.11 WLANs and for the IEEE 802.16e WMANs.

For the infrastructure mode of IEEE 802.11 WLANs, a novel method is presented to schedule the listening duration of sleeping stations and to balance the amount of wakeup stations in each beacon interval. This method saves stations' power by reducing the probability of collision. We also control the amount of wakeup stations which can send the PS-Poll frames to get back their buffered data to avoid contention. Three different mechanisms, single wakeup stations, the smallest association ID, and the smallest queue length, are proposed for to control the access order of wakeup stations. Our simulation results show that the proposed methods are effective in the power-saving.

For the ad hoc mode of IEEE 802.11 WLANs, a novel self-configuring power-saving protocol, called as SCPS, is proposed. Stations choose their listening intervals as the infrastructure mode. Besides, all stations in the PS mode can adjust their wakeup schedules whenever a station enters or exits the PS mode. The adjustment can balance the amount of wakeup stations in each beacon interval so that both the contention for transmission medium and the collisions in transmission can be ameliorated, which results in more efficient energy usage. Simulation results show that SCPS successfully balances the amount of wakeup stations in each beacon interval, increases the sleep ratio, and reduces the collision probability.

For the IEEE 802.16e broadband wireless networks, we proposed several energy efficient scheduling approaches. Instead of considering a single Mobile Subscriber Station (MSS) as most of the current researchers do, multiple MSSs are considered in our work. We consider constant bit rate traffic with QoS delay constraint. The proposed approaches address both energy efficiency and bandwidth utilization. Two classes of scheduling approaches are proposed, the periodical autonomic sleeping cycle (PASC) and the periodical uniform sleeping cycle (PUSC) approaches. In the PASC, the QoS information of all MSSs needs not to know beforehand. Each MSS uses its own sleeping cycle for power saving. While the possible QoS requirement of the MSSs can be known beforehand, the efficiency of the scheduling can be improved further. Thus, the PUSC approaches let all MSSs use the same length of sleeping cycle for their wakeup schedule. BS simply schedules the MSSs' wakeup time. Simulation results show that both PUSC and PASC can have higher power efficiency. Besides, the PUSC approaches are superior to the PASC approach on the bandwidth utilization under the application of VoIP.

# Acknowledgements

Special thanks goes to my advisor Professor Rong-Hong Jan for his guidance in my dissertation work and the instructions on writing articles. Thanks also to all members of Computer Network Lab for their assistance and kindly help in both the research and the daily life during these years. Finally, I will dedicate this dissertation to my families for their love and support.

# Contents

**Abstract (in Chinese)** ........ i

**Abstract (in English)** ........ iii

**Acknowledgements** ........ v

**Contents** ........ vi

**List of Tables** ........ viii

**List of Figures** ........ ix

**Chapter 1 Introduction** ........ 1

**Chapter 2 Reviews and Related Works** ........ 7

2.1 Power saving protocols for 802.11 WLANs ........ 7
2.1.1 Infrastructure mode ........ 7
2.1.2 Ad hoc mode ........ 10
2.2 Power saving protocols for 802.16e WMANs ........ 12

**Chapter 3 Power Saving Protocols for IEEE 802.11 Infrastructure Mode** ........ 15

3.1 Problem statement ........ 16
3.2 Load-aware wakeup scheduling ........ 17
3.3 Contention avoidance traffic scheduling ........ 19
3.3.1 Multiple wakeups single access ........ 20
3.3.2 Multiple wakeups multiple accesses ........ 22
3.4 Simulation and results ........ 25
3.4.1 Performance metrics and environment setup ........ 25
3.4.2 Results and discussion ........ 27
3.5 Summary ........ 29

**Chapter 4 Power Saving Protocols for IEEE 802.11 Ad Hoc Mode** ........ 31

4.1 Problem statement ........ 32
4.2 The SCPS approach ........ 33
4.2.1 The wakeup information of other stations ........ 33
4.2.2 Timer synchronization ........ 36
4.2.3 Maintaining a consistent wakeup table ........ 37
4.2.4 A station enters the PS mode ........ 40
4.2.5 A station exits the PS mode ........ 44
4.3 Simulation results ........ 46
4.3.1 Performance metrics and environment setup ........ 46
4.3.2 Experimental results ........ 48

4.4 Summary ... 56

**Chapter 5 Power Saving Protocols for IEEE 802.16e WMANs ... 57**

5.1 Autonomic sleep cycle ... 58
5.2 Uniform Sleeping Cycle ... 65
5.2.1 Multiple MSSs Power-Saving Scheduler (MMPS) ... 66
5.2.2 MMPS with Fragment Collection (MMPS-FC) ... 67
5.2.3 MMPS with Boundary Free (MMPS-BF) ... 69
5.2.4 Maximize the bandwidth utilization of MMPS-BF ... 70
5.3 Simulation Results ... 72
5.3.1 Environment setup ... 72
5.3.2 Numerical results ... 73
5.4 Summary ... 78

**Chapter 6 Conclusions and Future Works ... 79**

**Bibliography ... 80**

# List of Tables


TABLE 3-1 DETAIL SIMULATION CONFIGURATIONS ........................................................ 26

TABLE 4-1 THE WAKEUP INFORMATION TABLE ............................................................. 34

TABLE 4-2 THE NUMBER OF BEACON INTERVALS NEEDED OR THE CLOCK TO DRAFT FOR A DIFS ........................................................................................................ 36

TABLE 4-3 A SEQUENCE OF $W_I(T)$ AND $N(T)$ ..................................................... 38

TABLE 4-4 A SEQUENCE OF $W_I(T)$ AND $N(T)$ ..................................................... 41

TABLE 4-5 A SEQUENCE OF $W_I(T)$ AND $N(T)$ FOR STATION $J$ WITH FIRST WAKEUP TIME $T = 5$ . 41

TABLE 4-6 A SEQUENCE OF $W_I$(T) AND $N(T)$ FOR STATION C EXISTING THE PS MODE AT TIME $T$=5 ........................................................................................................ 45

TABLE 4-7 A SEQUENCE OF $W_I(T)$ AND $N(T)$ WHEN C EXISTS THE PS MODE AT TIME $T$=5....... 45

TABLE 5-1 THE AVERAGE REQUIRED BUFFERED SIZE FOR EACH MSS ................................ 78

# List of Figures


FIGURE 2-1 THE POWER SAVING MECHANISM OF IEEE 802.11 INFRASTRUCTURE MODE ...... 9

FIGURE 2-2 THE POWER SAVING MECHANISM OF IEEE 802.11 AD HOC MODE..................... 10

FIGURE 2-3 THE THREE POWER SAVING CLASSES OF IEEE 802.16E WMANS .................... 13

FIGURE 3-1 A SEQUENCE OF $W_I(T)$ AND N(T), $N(T)_J$ : THE $N(T)$ AFTER $J$ IS INCLUDED ............ 16

FIGURE 3-2 MULTIPLE WAKEUP SINGLE ACCESS ................................................................ 21

FIGURE 3-3 AN EXAMPLE OF SMALLEST AID FIRST METHOD ............................................ 22

FIGURE 3-4 AN EXAMPLE OF THE SMALLEST QUEUE LENGTH FIRST METHOD ...................... 24

FIGURE 3-5 THE AVERAGE SLEEPING TIME ......................................................................... 27

FIGURE 3-6 THE AVERAGE THROUGHPUT FOR EACH SCHEME ............................................ 28

FIGURE 3-7 THE LATENCY OF A SUCCESSFUL TRANSMISSION FOR EACH SCHEME ................ 29

FIGURE 3-8 THE IMPROVING RATE OF SLEEPING RATIO ...................................................... 29

FIGURE 4-1 THE POWER-SAVING OPERATION OF SCPS ....................................................... 32

FIGURE 4-2 THE FORMAT OF STATION WAKEUP INFORMATION (SWI) ................................. 35

FIGURE 4-3 THE AVERAGE AND THE MAXIMUM WAKEUP STATIONS IN AN ATIM WINDOW ... 49

FIGURE 4-4 THE RELATIONSHIP OF THE SLEEP RATIO AND THE TOTAL NUMBER OF STATIONS 50

FIGURE 4-5 THE RELATIONSHIP OF AVERAGE PACKET QUEUING DELAY AND THE TOTAL STATION NUMBER ........................................................................................................ 51

FIGURE 4-6 THE RELATIONSHIP OF PACKET DROP RATIO AND THE TOTAL NUMBER OF STATIONS ........................................................................................................ 52

FIGURE 4-7 THE RELATIONSHIP BETWEEN SLEEPING RATIO AND TRAFFIC GENERATION RATE (60 STATIONS; PACKET SIZE: 8000 BITS) ........................................................... 53

FIGURE 4-8 THE RELATIONSHIP BETWEEN SLEEPING RATIO AND TRAFFIC GENERATION RATE ........................................................................................................ 53

FIGURE 4-9 INFLUENCE OF PACKET SIZE (60 STATIONS; TRAFFIC GENERATION RATE: 8 PACKETS/S)........................................................................................................ 54

FIGURE 4-10 THE THROUGHPUT AND SLEEP RATIO AS FUNCTIONS OF THE LISTEN-INTERVAL IN SCPSB (60 STATION; TRAFFIC GENERATION RATE: 8 PACKETS/S) ............................... 55

FIGURE 4-11 THE AVERAGE WAITING TIME FOR A STATION TO SWITCH TO THE PS MODE IN

SCPSB ........ 55
FIGURE 5-1 EXAMPLE OF PERIODICAL AUTONOMIC SLEEPING CYCLE APPROACH ........ 60
FIGURE 5-2 AN EXAMPLE OF MMPS SCHEDULING ........ 67
FIGURE 5-3 EXAMPLE OF MMPS WITH FRAGMENT COLLECTION ........ 68
FIGURE 5-4 ALGORITHM OF MMPS-FC APPROACH ........ 69
FIGURE 5-5 A MMPS EXAMPLE WITH BOUNDARY FREE ........ 70
FIGURE 5-6 BANDWIDTH UTILIZATION ........ 73
FIGURE 5-7 THE NUMBER OF SERVICED MSSS ........ 74
FIGURE 5-8 COMPARISON OF SLEEPING RATIOS (16MBPS) ........ 76
FIGURE 5-9 THE BANDWIDTH UTILIZATION AND SLEEPING RATIO UNDER THE CASE OF 1.6MBPS RATE ........ 77

# Chapter 1

# Introduction

In the traditional wired network, the locations to attach networks are limited by the hubs which are the entrances to the backhaul networks connected by the physical electrical wires. This inconvenient network accessing mechanism results in the popularity of the mobile devices and the development of wireless communication technologies. Nowadays, wireless communication has become an indispensable ability for mobile devices to attach the networks. Among all of the current wireless communication technologies, the IEEE 802.11 WLANs [1] is the widest adapted technology to build the personal wireless local area network and the IEEE 802.16 WMANs [2] is the most expected technology to bridge the traffic between the local area networks and the public networks.

The IEEE 802.11 WLANs standard has been released by the IEEE LAN Standards Committee since 1997 for wireless local area network. This standard defines the medium access control (MAC) and physical (PHY) layers for a wireless LAN operating on 5 GHz and 2.4 GHz public spectrum bands. Because of the simple and low cost characteristics, the IEEE 802.11 WLAN has been widely deployed at the company, family, airport, and etc. Everyone can easily and freely deploy his own wireless LAN via the low cost facilities of this standard. Serial contents are specified in this standard such as the basic MAC level access control and data delivery services[1], the mobility of devices between multiple wireless LANs[3], the physical layer signaling techniques and interfaces[4-6], privacy and security of user data transferring[7], and etc.

The IEEE 802.16 WMANs standard which was first approved in December 2001 specifies point to multipoint broadband wireless transmission. It is developed to replace the current last-mile internet access mechanism which is relied on the cable or ADSL in the wired networks. It can also provide service to a large amount of Mobile Subscriber Stations (MSSs). In the standard of IEEE 802.16 MAC protocol, two main duplex modes, frequency division duplex (FDD) mode and time division duplex (TDD) mode, are specified for the communication between a Base Station (BS) and a MSS. In order to support various applications, such as Voice over IP (VoIP), video streaming, web browsing, etc., the IEEE 802.16e defines four kinds of service classes: Unsolicited Grant Service (UGS), Real-Time Polling Service (rtPS), Non-Real-Time Polling Service (nrtPs), and Best Effort (BE).

The IEEE 802.16 WMANs standard also standardizes the PHY and the MAC as the IEEE 802.11 WLANs. The specified contents in this standard include mobility and power saving [8], mobility management [9], bridging [10], multi-hop relay [11], and etc. More amendments of IEEE 802.16 WMANs are still under proceeding.

No matter the IEEE 801.11 WLANs or the IEEE 802.16 WMANs, the power source of the mobile devices usually comes from the equipped batteries. A mobile device will lose its availability when all equipped batteries exhaust energy. With the popularity of mobile devices, economizing the power of battery to retain availabilities of mobile devices gradually becomes an important issue that can not be neglected. Considering the slow progress in developing high capacity battery, making good use of the battery's power seems to be more practical way to prolong the operation time of mobile devices.

In a mobile device, the wireless communication component is one of the main power consumption sources. Designing the power saving (PS) communication protocols can efficiently extend the operation time of mobile devices. Thus, many efforts have been done to save the power consumed by message communication. In physical and MAC layers, the designed PS protocols can save power by using traffic-aware bit-rate selector [12-15], power control mechanisms [16-18], or adapting the activity periods of communication [19-25]. Besides, the power saving protocols can also be designed by considering frame aggregation [44] or reducing the possible retransmission [23]. Power saving protocols designed for network and TCP layers usually consider applying power-aware routing approaches [26-28] or dynamically changing the beacon interval size based on the TCP traffic [29]. Furthermore, the beacon interval size can also be adapted dynamically based on the users' behavior [25]. Among all of the power saving mechanism, the most efficient way to save power is to prevent mobile devices from idly wake up and let they stay in low power mode as long as possible. The lower power mode is also called as *sleep mode*.

The standard of IEEE 802.11 WLANs also specifies a basic sleep mode mechanism for both the infrastructure mode and the ad hoc mode. However, due to the power saving mechanism in the standard is inefficient, many researches have been proposed to amend them. For the infrastructure mode, some proposed a TDMA-like mechanism [41] [42] to reduce the power wasted on idly listening and retransmission; others let the stations flexibly adapt their listening interval length according to the traffic arrival time [17][43][45].

For the ad hoc mode, flexibly adapting the size of the beacon interval [20] and the size of ATIM window size are proposed [21-22] to save the power wasted on the idly listening and to increase the bandwidth utilization. All of these proposals focus on extending the sleep duration within a beacon interval. Stations still need to wake up in every beacon interval. For saving more energy, stations need to sleep more beacon intervals instead of one [30].

In this dissertation, we propose several power saving protocols for both the infrastructure mode and the ad hoc mode of IEEE 802.11 WLANs. For the infrastructure mode, we present a novel method to arrange wakeup schedule for sleeping stations. Our purpose is to balance the number of wakeup stations in each beacon interval. This method can reduce the probability of collision and thus the station can save more power. Next, we consider how to poll the wakeup stations to send the PS-Poll frame to get back their buffered data so that the contention can be avoided. Three different access scheduling mechanisms are proposed for the contention avoidance. In the first mechanism, only one of wakeup stations is scheduled to access the buffered data. The second and third mechanisms schedule a subset of wakeup stations to retrieve their buffered data within a beacon interval. The access sequences of the second and third mechanisms are based on the association ID (AID) and the queue length, respectively.

For the ad hoc mode, a self-configuration power-saving (SCPS) protocol for one-hop ad hoc networks is proposed. The proposed SCPS approach let stations sleep more than one beacon interval instead of waking up at every TBTT (target beacon transmission time). Furthermore, the SCPS protocol provides more flexibility for stations to select the length of listen-interval and each station can decide its own listen-interval as in the infrastructure mode. SCPS also attempts to evenly arrange stations to wake up in beacon intervals in order to reduce the medium-contention probability. When a station enters or exits the PS mode, stations in PS mode will arrange their wakeup schedule automatically. Its results are better then the Quorum based approach [30].

The IEEE 802.16 WMANs specifies its power saving mechanism in the standard of IEEE 802.16e [8]. Although the power saving mechanisms in IEEE 802.16e are designed for the central coordinated networks which is similar to the infrastructure mode of IEEE 802.11 WLANs, the existing power saving mechanisms for the IEEE 802.11 WLANs can not be applied to IEEE 802.16e directly. It results from that they have different medium access control mechanism (IEEE 802.11 WLANs is contention-based while the IEEE 802.16e is slot-based) and different QoS requirements (IEEE 802.11 WLANs are not emphasized as 802.16e).

Therefore, several researches have studied the power consumption in a power saving mode MSS [31-34]. New power-saving protocols are also proposed to determine the length of sleeping period [36-38]. Furthermore, a protocol that consider mixed mode of power saving classes is also proposed to improve the power saving efficiency [35].

In this dissertation, we consider the UGS traffic with QoS delay requirement and adapt the second power-saving class defined in IEEE 802.16e standard. Two kinds of power saving schedule approaches for multiple MSSs are proposed, the periodical autonomic sleeping cycle approach (PASC) and the periodical uniform sleeping cycle approach (PUSC). In the PASC, each MSS follows its own sleeping cycle to wake up. At the same time, the bandwidth utilization in each OFDM frame is also considered. The BS will try to schedule the wakeup time of newly joined MSSs to a light load OFDM frames without preempting those MSSs which have been scheduled. So, the traffic load in each OFDM frame can be balanced and BS can allocate bandwidth to new MSS easily.

Thus, for serving more MSSs, we proposed the PUSC approach. The PUSC assumes that the possible minimum delay constraint of MSSs is known beforehand. So, the shortest sleeping cycle within all MSSs can be known. The allocated OFDM frames for each MSS are well controlled and overlap-free. We condense the bandwidth of each available OFDM frame to increase the bandwidth utilization.

The rest of this dissertation is organized as follows. The reviews of the power saving protocols of IEEE 802.11 and IEEE 802.16e are given in chapter 2. In the chapter 3 and 4, we present our proposed power saving protocols for IEEE 802.11 infrastructure mode and ad hoc mode, respectively. The IEEE 802.16e power-saving protocols are presented in chapter 5. Finally, the conclusions and future works are shown in chapter 6.

# Chapter 2

# Reviews and Related Works

Power saving mechanism is important for the mobile devices. Using a power saving wireless communication protocol can economize the power spending on the communication. Many researches have been done and we will review some of those important works in this chapter.

## 2.1 Power saving protocols for 802.11 WLANs

The standard of IEEE 802.11 specifies two operation modes, the infrastructure mode and ad hoc mode. The infrastructure mode is a centrally controlled network system. A special station, called as Access Point (AP), coordinates the communication between the other stations in this mode. On the contrary, the ad hoc mode is a completely distributed network system. There is no central coordinator and stations cooperated with others distributedly. In the following subsections, we detail the power saving operations of these two modes and their related researches.

### 2.1.1 Infrastructure mode

In infrastructure mode, a station can enter the power-saving mode (PS mode) by shutting down its transceiver to save power. The station can inform AP its power management state. After AP has the power management state of every station which has associated with it. AP can either deliver the arrival frames to the station if it is waking up or buffered it if the station is in PS mode.

AP periodically broadcasts the information of buffered frames to indicate its serviced stations to get back their data via the traffic indication map (TIM) in beacon frames. The TIM is a virtual bitmap in which each bit corresponds to a particular AID where AID is assigned by AP when a station associates to the AP.

Power-saving stations have to wake up to listen for beacon frames and check the TIM. By this way, a mobile station can determine whether the AP has buffered frames for it. If the AP seldom buffers frames for the station, the station does not require waking up to check every beacon frame. Instead, it wakes up every *listen-interval* to check the beacon frame. A listen-interval is a number of beacon intervals for which the mobile station may choose to sleep. If the station finds that the AP has buffered data for it, it will send a PS-Poll control frame to retrieve the buffered frames. When multiple stations have buffered frames, all stations with buffered frames contend the medium for sending PS-Poll. After sending the PS-Poll, a station has to awake until the buffered frames are received or the bit in the TIM corresponding to its AID is no longer set.

For example, as shown in Figure 2-1, a station, denoted as STA, is wakeup in the first beacon interval and receives the beacon frame in which the TIM indicates buffered data for it. Then, STA contends the medium for sending a PS-Poll frame to inform the AP that it is wakeup and ready to get back the buffered data. After AP receives the PS-Poll, it transmits a buffered frame to the STA. The STA returns an ACK frame to inform AP that the frame is received completely.

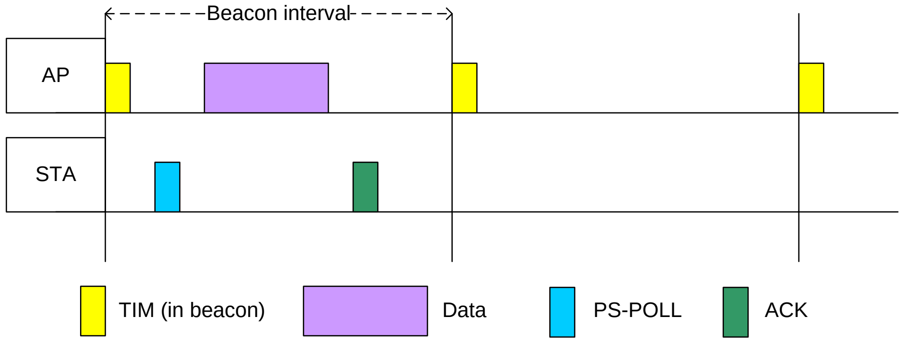



**Figure 2-1 the power saving mechanism of IEEE 802.11 infrastructure mode**

Idly listening reduces the possible sleeping time of stations. To improve the power saving mechanism of infrastructure mode in the standard, Ghazanfar *et al* proposed a MAC protocol which efficiently allocates the bandwidth of AP into multiple time slots[41] [42]. This protocol let the stations operate in the TDMA mode and AP coordinates the stations. Ghazanfar *et al* also provided more detailed analysis about the benefits that obtain from the TDMA operation mode in [42].

Although the TDMA like mechanism can efficiently save the energy wasted on idly waiting and on packet retransmission, it needs greatly modification to the standard. Thus, for minimizing the penalty on standard modification, Shuvo *et al* proposed an AMS protocol, which measures the average time between consecutive incoming packets under different VoIP applications for the voice traffic [43]. After then, the stations can conserve the maximum amount of energy by stay in the sleep state. Moreover it can also introduce as little additional delay as possible. It predicts exactly the time of the next packet transmission or packet arrival, and entering the awaken state just before this occurs. Similar mechanisms which adapt the listening interval length based on the traffic are also proposed in [17] and [45].

## 2.1.2 Ad hoc mode

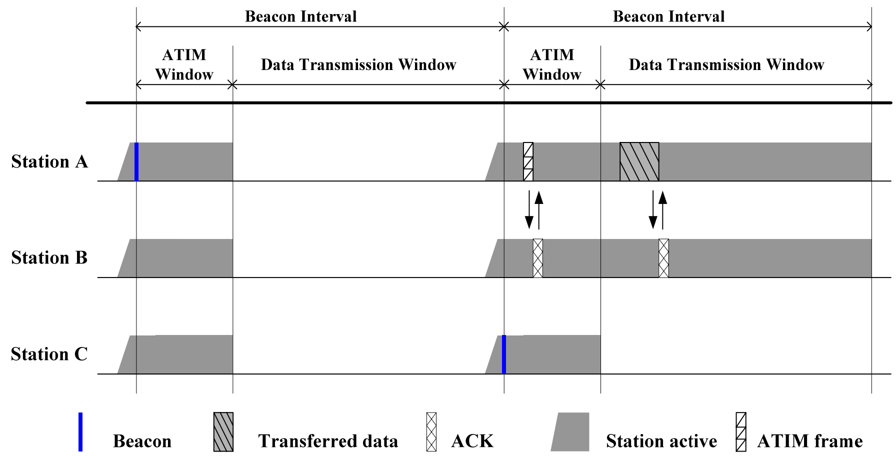



**Figure 2-2 the power saving mechanism of IEEE 802.11 ad hoc mode**

Design a power saving mechanism for ad hoc mode is more difficult than the infrastructure mode because there is no centralized coordinator. Stations must run the power saving mechanism distributedly. Figure 2-2 illustrates the power saving mechanism of the ad-hoc mode in the standard of IEEE 802.11. A beacon interval is divided into two parts, the ATIM window and the data transmission window. Every station has to wake up during the ATIM window. If a station has buffered data for another one, it sends an ATIM frame to notify the receiver. After successfully transmitting the ATIM frames, all stations compete for the transmission medium to send their buffered data during the data transmission window.

On the other hand, stations that have no data to send or receive go to sleep at the end of the ATIM window. It will not wake up until the next beacon interval. All other stations will wake up during the entire data transmission window. For example, in Figure 2-2, all three stations (A, B, and C) wake up in the first ATIM window. They do not send or receive any ATIM frames so they sleep at the end of the ATIM window.

All three stations wake up again in the second ATIM window. Assume that station A has buffered data for station B at this time. It sends an ATIM frame to station B during the second ATIM window and sends the buffered data during the second data transmission window. Because station C does not have data to send or receive, it goes to sleep at the end of the second ATIM window.

The fix-length beacon interval in the ad hoc mode limits the bandwidth flexibility while bandwidth demands of stations change randomly. Thus, to improve the power saving mechanism in the ad hoc mode of 802.11 standard, Liu *et al*. proposed a variable-length beacon-interval mechanism [20]. The length of the transmission window was determined by all the stations that have succeeded in ATIM frame transfer. So the length of a beacon interval can be dynamically adjusted according to the demands of stations.

Similar to the fix-length beacon interval, the duration of the ATIM window is fixed and is determined when the system starts up in the 802.11. A *fix-duration* ATIM window results in low bandwidth utilization. Flexibly adapting the ATIM window to reduce the power wasted on idle listening is proposed [21, 22].

In [21], if a station discovers that the channel has been idle for more than a predefined amount of time, that station will assume that all other stations are idle. At this time, the ATIM window ends and the data transmission window starts. In [22], Jung and Vaidya proposed a dynamic ATIM window adaptation mechanism, called the *dynamic power saving mode* (DPSM). DPSM dynamically adjusts the ATIM window based on the number of transmission requests. This improves the sleep time and bandwidth utilization. However, stations still have to compete for the medium for transmitting ATIM frames and data frames.

In order to soothe the contention, several distributed mechanisms have been proposed [25] [46] [47]. They suggest that stations first compete for transmitting ATIM frames. During the competition an order for transmitting data frames is established. The contention during the data transmission window is therefore avoided.

The above approaches only consider the transmission schedule within a beacon interval or adjust the ATIM window to fit the traffic load. Stations still have to wake up in *every* ATIM window even if they have no data to send or receive. In order to avoid waking up in every beacon interval, Chao *et al*. proposed the quorum-based energy conservation (QEC) protocol [30]. Before entering the power saving mode, each station creates an $n \times n$ grid. Each entry in this grid denotes a beacon interval. Each station randomly chooses one row and one column in this grid as its wakeup intervals. Therefore, any two stations will share at least two entries. They can communicate with each other during the beacon intervals denoted by the shared entries.

## 2.2 Power saving protocols for 802.16e WMANs

IEEE 802.16e standard [8] defines three power saving classes shown as figure 2-3. Each connection of a MSS can select a specific power-saving class. MSSs need to negotiate with the BS to decide the power-saving parameters such as listening period and sleeping period. A round of a sleeping period and a listening period is defined as one ***sleeping cycle***.

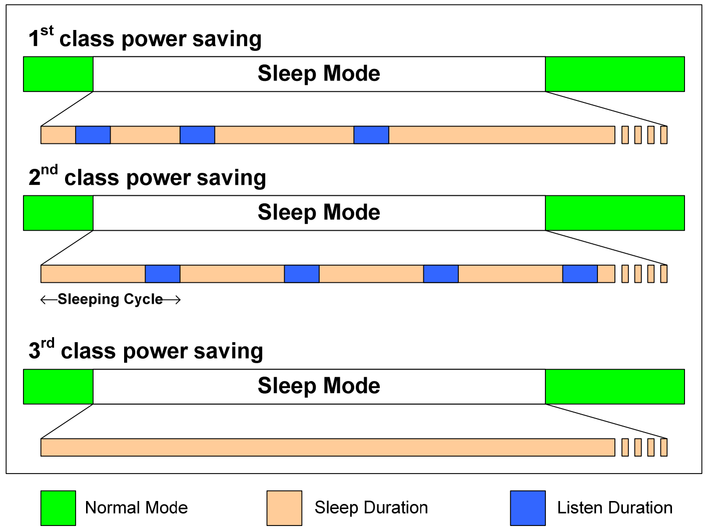



**Figure 2-3 the three power saving classes of IEEE 802.16e WMANs**

In the first power-saving class, each MSS sleeps for a period of time and then wakes up to listen. During the listening period, if no packets are sent or received, the MSS doubles the length of its next sleep cycle. This kind of power-saving is suitable for web browsing or data access services. In the second power saving class, a MSS is required to repeat the sleeping and listening periods in a round-robin fashion. The length of both the sleeping and listening periods in the sleeping cycle is fixed. This sleep mode works well for real-time applications that have packets to send or receive periodically such as VoIP and video streaming services. The third power-saving class requires a predefined sleep period length. The MSS simply sleeps for a predefined period of time and then returns to normal operation.

Several studies have been proposed to analyze the power consumption for IEEE 802.16e while a MSS operates in the power-saving mode [48-52]. There are some scheduling mechanisms proposed to determine the length of sleeping period [36-38]. In [36], the length of the sleeping period is varied according to the traffic type. However, the scenario is only valid under one MSS and the QoS delay constraint is not considered. In [37], although the QoS delay constraint is considered, the scenario can not be applied to multiple-MSS environment. In [38], a scheduling algorithm for multiple MSSs with QoS delay constraints is proposed. Authors classified the MSSs into two categories: primary and secondary MSSs. In order to save energy, the algorithm grants one primary MSS to use the bandwidth in burst mode. The other MSSs or we say secondary MSSs are only given the necessary bandwidth to meet the requirements of the QoS delay constraint. However, this algorithm works well only when all MSSs have light traffic load. In [39], an analysis is provided via semi-Markov Decision Processes (Semi-MDP) to find an optimal way to switch between Type I and II power saving modes while considering the mix modes operation. However, the discussion of bandwidth utilization is absent. It only considers how to select a suitable power saving class.

Thus, in this dissertation, we consider the UGS traffic which has QoS delay constraint and adapt the second power-saving class defined in IEEE 802.16e standard. We propose two kinds of power saving schedule approaches for multiple MSSs, the periodical autonomic sleeping cycle approach (PASC) and the periodical uniform sleeping cycle approach (PUSC). We will discuss them in the chapter 5.

# Chapter 3

# Power Saving Protocols for IEEE 802.11 Infrastructure Mode

This chapter presents a method to arrange the sleep schedule of stations in the infrastructure mode of IEEE 802.11 wireless local area networks (WLANs). The goal is to balance the number of wakeup stations in each beacon interval. This method reduces the probability of collision and thus the station can save more power. To avoid the contention, this method considers how to poll the wakeup stations to send the PS-Poll frame to get back their buffered data. Three different access scheduling mechanisms are proposed for the contention avoidance. In the first mechanism, only one wakeup station is scheduled to access the buffered data. The second and third mechanisms based on the smallest association ID (AID) first and the smallest queue length first, respectively, arrange a subset of wakeup stations to get back their buffered data within a beacon interval. Simulation results show that the proposed methods are effective in the power-saving.

The organization of this chapter is as follows. Problem statement is given in section 3.1. A new wakeup scheduling mechanism is considered in section 3.2 and the three contention avoidance mechanisms for polling wakeup stations are presented in section

3.3. In section 3.4, we give the simulation results to show the effectiveness of our proposed methods. Finally, a summary is given in section 3.5.

# 3.1 Problem statement

Considering the figure 3-1, there are six power-saving stations, A, B, C, D, E, and F, whose listen-intervals are 1, 2, 3, 6, 6, and 6. $w_i(t)$ is used to indicate the sleeping state of station $i$ at beacon interval t. If $w_i(t)$ is 1, station is active, otherwise station is in sleep mode. $n(t)$ is the total number of wakeup stations in beacon t.

| ***t*** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $w_A(t)$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| $w_B(t)$ | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| $w_C(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| $w_D(t)$ | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| $w_E(t)$ | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| $w_F(t)$ | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| $w_J(t)$ | - | - | - | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $n(t)$ | 3 | 2 | 1 | 3 | 2 | 3 | 3 | 2 | 1 | 3 | 2 | 3 | 3 | 2 | 1 | 3 | 2 | 3 |
| $n(t)_J$ | 3 | 2 | 1 | 3 | 3 | 3 | 3 | 3 | 1 | 3 | 3 | 3 | 3 | 3 | 1 | 3 | 3 | 3 |

**Figure 3-1 a sequence of $w_i(t)$ and n(t), $n(t)_J$ : the $n(t)$ after $J$ is included**

In order to trace the wakeup time of each station, AP needs to maintain a wakeup counter, denoted as $c_i(t)$, for each sleeping station i. $c_i(t)$ indicates the remaining beacon interval that station $i$ will wake up. Initially, AP sets $c_i(t) = \ell_i - 1$, where $\ell_i$ is the listen-interval of station i. Station $i$ wakes up if $c_i(t)$ become 0 and the counter will reset to $c_i(t) = \ell_i - 1$ for further counting down. AP sets counter $c_i(t+1)$ for beacon interval $t$+1 as follows:

$$c_i(t+1)=\begin{cases} c_i(t)-1 & if \quad c_i(t)\neq 0 \\ \ell_i-1 & if \quad c_i(t)=0 \end{cases}$$

Thus, AP can find $w_i(t+k)$ for beacon interval t+k by

$$w_i(t+k)=\begin{cases} 1 & if \quad k \bmod \ell_i = 0 \\ 0 & otherwise \end{cases}$$

The wakeup scheduling problem (WSP) can be formulated as follows: Given a set of sleeping stations S at beacon interval t. Consisting m stations and each of the station *i* has wakeup count $c_i(t)$ and listen-interval $\ell_i$. For a new sleeping station *j*, we assign an initial value to $c_j(t)$ such that the maximum value of $n(t+k)=\sum_{i\in S\cup\{j\}} w_i(t+k)$ , *k* = 1, 2, 3 . . . is minimized. In the next section, we present the mechanism of load-aware wakeup scheduling.

# 3.2 Load-aware wakeup scheduling

By observing the sequence of *n(t)* in Figure 3-1, we find that a pattern repeats every six beacon intervals, e.g., (*n(1), n(2), …, n(6)*) = (*n(7), n(8), …, n(12)*) = (*n(13), n(14), …, n(18)*) = (3, 2, 1, 3, 2, 3). The length of this repeating pattern, ***r***, can be found by computing the least common multiple (lcm) of listen-interval $\ell_i$, $i \in S$. For example, the listen-intervals of stations, A, B, C, D, E, and F are 1, 2, 3, 6, 6, and 6, respectively, in Figure 3-1. Then

$$r = lcm\ \{1, 2, 3, 6, 6, 6\} = 6$$

Thus, $n^* = \max\{n(t+1),\ n(t+2),\ \ldots,\ n(t+r)\} = \max\{n(t+k)\ |\ k=1,\ 2,\ldots\}$. Now, we want to add a new sleeping station $j$ with listen-interval $\ell_j$ to the sleeping station set $S$ with repeating pattern size $r$ and assign an initial value to $c_j(t)$. A stepwise solving method for the WSP problem is given as follows.

1. Find $r = lcm\{\ell_i, r\}$ and a sequence of total number of wakeup stations $(n(t+1), n(t+2), \ldots, n(t+r))$ for the first $r$ intervals for sleeping station set $S \cup \{j\}$.

2. For $i = \ell_j - 1, \ldots, 1, 0$, perform the following operations:

(a) Set $c_j(t) = i$ and find $(w_j(t+1), w_j(t+2), \ldots, w_j(t+r))$;

(b) Set $(n(t+1), n(t+2), \ldots, n(t+r)) = (n(t+1), n(t+2), \ldots, n(t+r)) + (w_j(t+1), w_j(t+2), \ldots, w_j(t+r))$;

(c) Find $n_i = \max\{n(t+1), n(t+2), \ldots, n(t+r)\}$.

3. Find $n^* = \min\{n_i \mid i = \ell_j - 1, \ell_j - 2, \ldots, 0\}$, say $n^* = n_k$, and thus set $c_j(t) = k$.

For example, six stations with $r = 6$ as given in Figure 3-1, station $J$ with $\ell_J = 3$ enters the sleeping mode. The AP applies the above solving method to determine the initial value of counter $c_J(t)$ for station $J$ as follows.

1. $r = lcm\{3,6\} = 6$ and $(n(t+1), n(t+2), \ldots, n(t+6)) = (3,2,1,3,2,3)$

2. $i = 2$:

   (a) Set $c_j(t) = 2$ and find $(w_j(t+1), w_j(t+2), \ldots, w_j(t+6)) = (0,0,1,0,0,1)$;

   (b) Set $(n(t+1), n(t+2), \ldots, n(t+6)) = (3,2,1,3,2,3) + (0,0,1,0,0,1) = (3,2,2,3,2,4)$;

   (c) Find $n_2 = \max\{3,2,2,3,2,4\} = 4$.

$i = 1$:

(a) Set $c_j(t) = 1$ and find $(w_j(t + 1), w_j(t + 2), \ldots, w_j(t + 6)) = (0,1,0,0,1,0)$;

(b) Set $(n(t + 1), n(t + 2), \ldots, n(t + 6)) = (3,2,1,3,2,3) + (0,1,0,0,1,0) = (3,3,1,3,3,3)$;

(c) Find $n_1 = \max\{3,3,1,3,3,3\} = 3$.

$i = 0$:

(a) Set $c_j(t) = 0$ and find $(w_j(t + 1), w_j(t + 2), \ldots, w_j(t + 6)) = (1,0,0,1,0,0)$;

(b) Set $(n(t + 1), n(t + 2), \ldots, n(t + 6)) = (3,2,1,3,2,3) + (1,0,0,1,0,0) = (4,2,1,4,2,3)$;

(c) Find $n_0 = \max\{4,2,1,4,2,3\} = 4$.

3. Find $n^* = \min\{4,3,4\} = 3$, i.e., $n^* = n_1$, and thus set $c_j(t) = 1$.

Note that according to IEEE 802.11 standard, if mobile station $j$ has no data to send, it can send a Null data frame with Power Management bit set to 1. The AP begins to buffer frames and sends an ACK frame to the station after receiving the Null data frame. We can just modify this step to incorporate our wakeup scheduling in IEEE 802.11 standard as follows: The AP begins to buffer frames, determines $c_j(t)$ and sends an ACK frame with $c_j(t)$ value to station $j$ after receiving the Null data frame. Then, the station $j$ sets its wakeup counter to $c_j(t)$ and enters the sleeping mode.

## 3.3 Contention avoidance traffic scheduling

In the previous section, we arrange stations' wakeup beacon intervals so that the number of wakeup stations in each beacon interval is balanced. In this section, we consider how to inform stations that frames are buffered such that the contention is avoided. Three different access scheduling mechanisms are proposed for the contention avoidance problem. In the first mechanism, only one wakeup station is scheduled to access the buffered data in a beacon interval by marking one bit in TIM. The second and third mechanisms schedule multiple wakeup stations to get back their buffered data within a beacon interval. The access sequence within the beacon interval is according to their AIDs and the length of queuing data.

### 3.3.1 Multiple wakeups single access

One of the simple ways to avoid contention is that we only inform a station that AP has its buffered frames at each beacon interval. So there is no contention problem of sending PS-Poll frame to get back its buffered data. Let $S_w(t)$ be the set including all stations waking up at beacon interval $t$. That is,

$$S_w(t)=\{i|i\in S, c_i(t)=0\}$$

where $S$ is the set including all sleeping stations. Let $S_b(t)$ be the set including all stations that frames are buffered in AP at beacon interval $t$. Thus, we can choose a station, say station $v$, from set $S_w(t) \cap S_b(t)$ with a largest listen-interval $\ell_v$ to inform that the AP has buffered frames for it.

It is possible that some stations in set $S_w(t) \cap S_b(t)$ use small listen-interval and they are never chosen by AP. To avoid such a case, we associate each station $v$ in $S_w(t) \cap S_b(t)$ with an age, denoted as $a_v$. Initially, the age of each station is set to zero. For each beacon interval, if a station in set $S_w(t) \cap S_b(t)$ is not selected to inform, AP increases its age by one; otherwise, AP sets its age to zero. Thus, AP can choose a station, say station $v$, from set $S_w(t) \cap S_b(t)$ with a largest value of $\ell_v + a_v$ to inform. Here we denote $\ell_v + a_v$ as $p_v$.

Figure 3-2 shows an example of this mechanism. Consider that there are four stations, $A$, $B$, $C$, and $D$, with listen-interval $(\ell_A, \ell_B, \ell_C, \ell_D) = (2, 2, 3, 1)$. Suppose there are 1,1,1,1 packets send to station A, B, C, and D in every beacon interval. Packet arrival rate of each station is one frame per beacon interval. In beacon interval $t$, stations $A$, $C$, and $D$ wake up in which station $C$, has maximum $p_C = \ell_C + a_C = 3 + 0$, is indicated in TIM to inform it that the AP has buffered its data. Stations $A$ and $D$ are deferred to their next wakeup beacon intervals. The AP sets ages $a_A = a_A + 1$ and $a_D = a_D + 1$. In the beacon interval $t + 1$, stations $B$ and $D$ wake up. Because $\ell_B + a_B = \ell_C + a_C = 2$, the AP selects station $B$, arbitrarily, to inform it has buffered frames. Similarly, station $A$ is chosen to inform in beacon interval $t + 2$. At beacon interval $t + 3$, $\ell_B + a_B < \ell_C + a_C < \ell_D + a_D$ and thus station $D$ is chosen to inform.

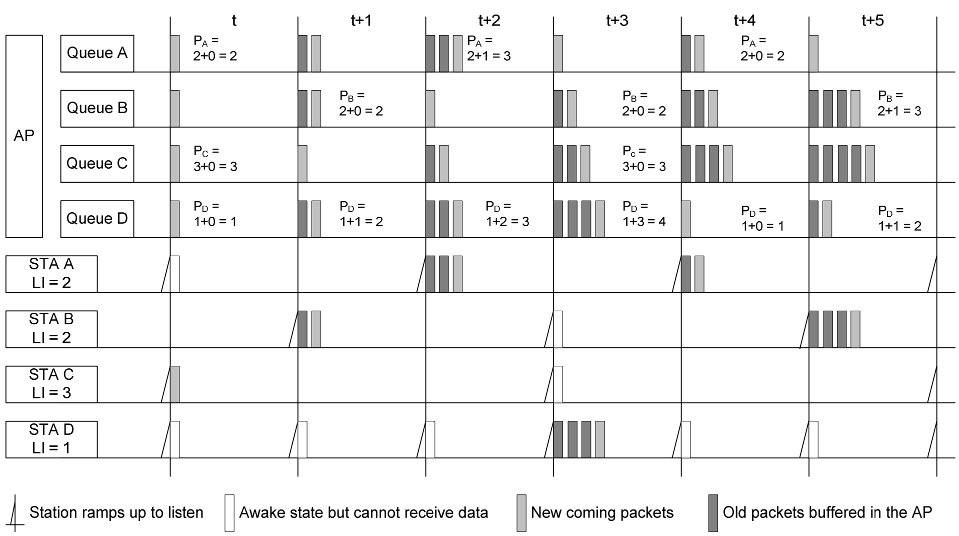



**Figure 3-2 multiple wakeup single access**

## 3.3.2 Multiple wakeups multiple accesses

Although the multiple wakeups single access mechanism avoids the contention among stations, it may lower the bandwidth utilization and increase the transmission delay. However, the AP knows how many frames it has buffered in queue, transmission rate and the length of beacon interval. Thus, the AP can determine how many frames it can transmit in a beacon interval and schedule the buffered frames by means of announcing the TIM. In the following, we give two methods to arrange the access sequence of stations.

### 3.3.2.1 The smallest AID first


**Figure 3-3 an example of smallest AID first method**

In order to control the traffic load in a beacon interval, AP selects a set of stations with an appropriate size from $S_w(t) \cap S_b(t)$ to inform them to retrieve the data. That is the total amount of buffered frames of selected stations should be less than the capacity of a listen-interval. This can avoid a station that awakes within whole beacon interval but can not get back its buffered data. Next, we modify the power management scheme of IEEE 802.11 WLANs such that a station retrieves the buffered frame according to the sequence of AID marked in TIM. That is, the station with smallest AID among the selected stations sends PS-Poll frame to retrieve buffered data first.

Figure 3-3 shows an example of the AID sequence method. There are four stations, *A*, *B*, *C*, and *D* with listen-interval $(\ell_A, \ell_B, \ell_C, \ell_D) = (2, 1, 3, 2)$ with under the service of an AP. Suppose there are 2, 2, 1, 2 packets send to A, B, C, D in each beacon interval. Their corresponding AIDs are 1, 2, 3, and 4 for stations *A*, *B*, *C*, and *D*, respectively. Suppose packet arrival rates of stations *A*, *B*, *C*, and *D* are 2, 2, 1, and 2 per beacon interval. The maximum number of frames that AP can transfer to stations in a beacon interval is 8. In beacon interval *t*, all of these four stations wake up. Because the number of buffered frames is $2 + 2 + 1 + 2 = 7$ ($7 < 8$), the AP marks AIDs 1, 2, 3, and 4 in the TIM. The stations check the TIM in beacon frame. They learn that 4 stations will send PS-Poll to retrieve their buffered frames and every station knows which station precedes it in access sequence. For example, station *C* has to wait stations *A* and *B* finishing their access. In beacon interval $t + 2$, $S_w(t + 2) \cap S_b(t + 2) = \{A, B, D\}$ and the number of frames buffered for stations, *A*, *B*, and *D* is 10 ($10 > 8$). Thus, based on the values of $p_A$ and $p_D$, the AP selects stations *A* and *D* to inform them to retrieve the buffered data.

### 3.3.2.2 The smallest queue length first

Instead of the smallest AID first, the AP can arrange the access sequence for the selected stations according to their associated queue lengths. The station with smallest queue length receives a highest precedence and thus it can have a longer sleeping time. In this method, we need to add an information element, describes the access sequence, as a component of the beacon frame. The station checks this information element for the access sequence.

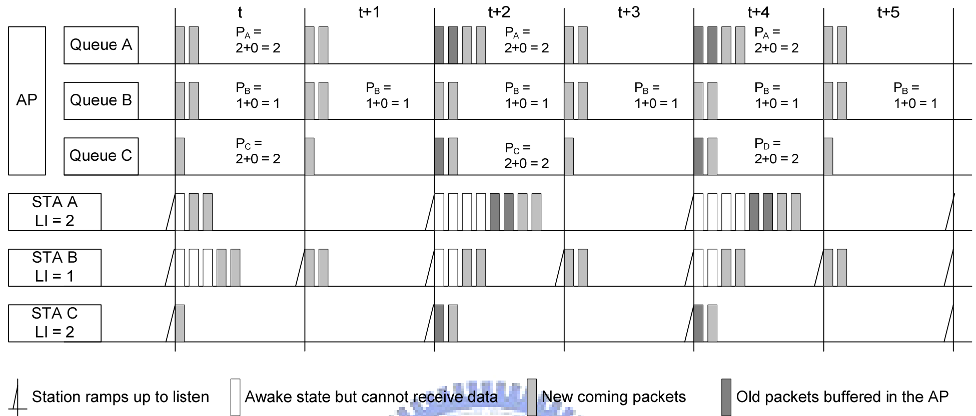



**Figure 3-4 an example of the smallest queue length first method**

Figure 3-4 shows an example of the smallest queue length first method. There are three stations, *A*, *B*, and *C* with listen-interval ($\ell_A$, $\ell_B$, $\ell_C$) = (2, 1, 2) with under the service of an AP. Suppose packet arrival rates of stations *A*, *B*, and *C* are 2, 2, and 1 frames per beacon interval. The maximum frame size that AP can transfer to stations in a beacon interval is assumed to be 8. In beacon interval *t*, all of the three stations wake up. Because the number of buffered frames are 2+2+1=5 (5<8), the AP marks AIDs 1, 2, and 3 in the TIM and adds the access sequence *C*, *A*, and *B* in the beacon frame. In beacon interval *t*+2, the access sequence is *C*, *B*, and *A*. Note that if two stations have same queue length, AP uses their $p_v$ values to break the tie.

# 3.4 Simulation and results

## 3.4.1 Performance metrics and environment setup

In this section, we show the performance analysis for the proposed schemes:

1. Load-aware wakeup scheduling (LAWS);

2. LAWS with multiple wakeups single access (LAWS+MWSA);

3. LAWS with multiple wakeups multiple access and the smallest AID first (LAWS+SAF);

4. LAWS with multiple wakeups multiple access and the smallest queue length first (LAWS+SQLF).

Note that all of these four schemes are enhanced from the PS mode of 802.11. The LAWS arranges station's wakeup time. The MWSA, SAF, and SQLF schemes can be used by AP to schedule the access sequence by marking the bits in TIM. We compare their performances against pure IEEE 802.11 PS mode by simulation. The performance metrics are given as follows:

1. *Average sleeping time of the station:* This measure is the duration that a station stays in the sleeping mode. If a scheme can make stations stay more time in sleeping, then stations will save more power.

2. *Average throughput:* This value shows the total amount of data successfully transmitting per second. If AP can efficiently schedule and distribute the access of its serving stations, it will have higher data throughput.

3. *Average latency of a successful transmission:* The latency is defined as the time duration starting while a packet is issued and buffered at AP and ending when the target station returns the acknowledge. An AP with a good scheduling scheme will make the latency as small as possible. Thus, the resources required for buffering data can be reduced.

**Table 3-1 detail simulation configurations**

| Data rate | 11Mbps |
|---|---|
| MAC header | 28 bytes |
| IP header | 20 bytes |
| UDP header | 20 bytes |
| Beacon frame | 28 bytes |
| ACK frame | 14 bytes |
| PS-POLL frame | 14 bytes |
| SIFS | 0.00001 sec |
| DIFS | 0.00005 sec |
| Slot time | 0.00002 sec |
| Beacon interval | 0.1 sec |

Our simulation uses an IEEE 802.11b wireless communication module with 11 Mbps data rate. An AP can serve at most 30 stations. Each station will randomly set 1 to 5 beacon intervals as its listen-interval size and its packet arrival rate is 3 packets per beacon interval. Packet size in our simulation is fixed and set to 1 Kbytes. Communication channel assumes to be clear and symmetric. The total simulation time is 3 minutes. The details of other simulation configurations such as header length, and inter-frame spaces (IFS) are listed in Table 3-1. Simulation results will compare the IEEE 802.11 PS mode with the proposed LAWS, LAWS+MWSA, LAWS+SAF, and LAWS+SQLF schemes.

### 3.4.2 Results and discussion

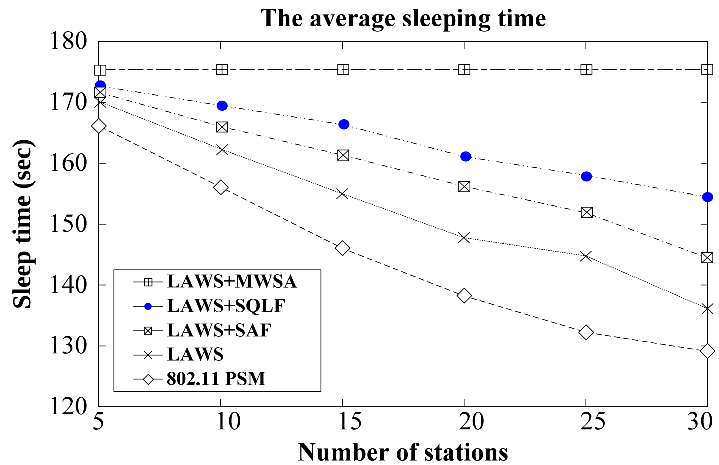



**Figure 3-5 the average sleeping time**

Figure 3-5 shows the relation between average sleeping time and number of stations. Considering contention-based schemes, LAWS can have more sleeping time than IEEE 802.11 PS mode in any size of stations. By using LAWS+MWSA, LAWS+SAF, and LAWS+SQLF schemes to reduce the contention within a beacon interval, stations can have more time on staying in sleeping than LAWS and IEEE 802.11 PS mode. In this figure, it seems that LAWS+MWSA has better sleeping time than LAWS+SAF and LAWS+SQLF. However, we will find in figure 3-7 that it trades the transmission latency with the sleeping time.

Figure 3-6 shows the average throughput for each scheme. From this figure, we can explicitly find that the throughput of IEEE 802.11 PS mode falls down when station number is greater than 20. However, our proposed schemes, LAWS, LAWS+MWSA, LAWS+SAF, and LAWS+SQLF, are not influenced as number of station increases. This is because our schemes can efficiently avoid the data collision between stations.

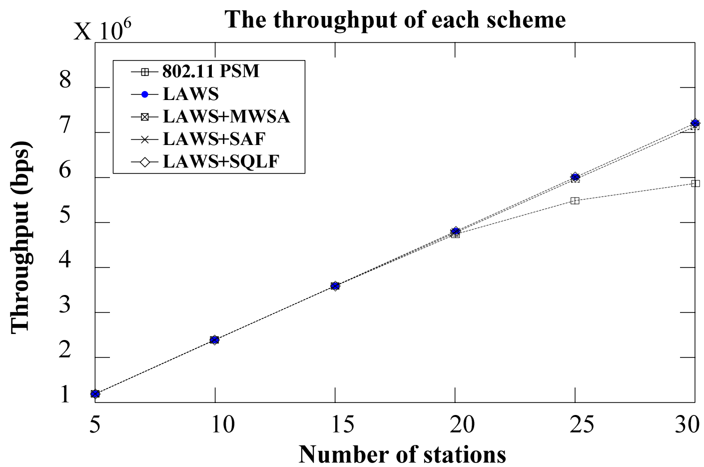



**Figure 3-6 the average throughput for each scheme**

In Figure 3-7, we show the average latency of a successful transmission for each scheme. For LAWS, LAWS+SAF, and LAWS+SQLF, all of their latency is smaller than 0.3 sec and increase slowly as number of stations grows. Because only one station is indicated within a beacon interval, the latency of LAWS+MWSA scheme is longer than the other proposed schemes. The pure IEEE 802.11 PS mode, however, will suffer the worst latency while number of stations increases.

Finally, Figure 3-8 shows the improving rate of sleeping time for each proposed scheme (compared to pure IEEE 802.11 PS mode). The improving rate $R_i$ of scheme $i$ is defined as $R_i = \frac{S_i - S_0}{S_0} \times 100\%$, where $S_0$ and $S_i$ are the average sleeping times for pure IEEE 802.11 PS mode and the proposed scheme $i$, respectively. By efficiently scheduling the wakeup time of sleeping stations, the sleeping duration of LAWS, LAWS+MWSA, LAWS+SAF, and LAWS+SQLF schemes can be improved significantly.

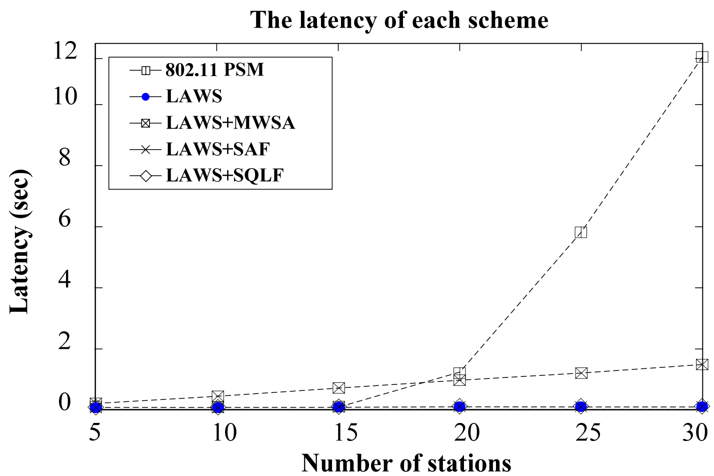



**Figure 3-7 the latency of a successful transmission for each scheme**

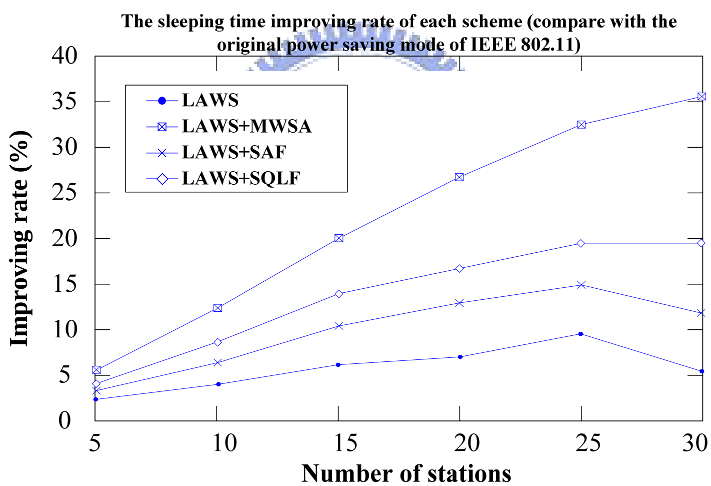



**Figure 3-8 the improving rate of sleeping ratio**

# 3.5 Summary

In this chapter, we propose a load-aware wakeup schedule scheme for infrastructure mode of IEEE 802.11 WLANs. The LAWS scheme balances the number of wakeup stations in each beacon interval to reduce the amount of contention stations. For avoiding the contention, MWSA, SAF, and SQLF scheme are used to arrange the access sequence of the wakeup stations within a beacon interval. Simulation results show that comparing to IEEE 802.11 PS-mode, the proposed LAWS, MWSA, SAF, and SQLF schemes can efficiently improve the sleeping duration of each station, average throughput, and transmission delay.

The following two issues should be considered in the implementation of the proposed schemes:

1. An *aging function* should be implemented in the AP to determine when buffered frames are old enough to be discarded.

2. If the mobile station misses the beacon, it should remain awake until it receives the next beacon. The mobile station checks the beacon frame. If the bit corresponding to its AID is set to zero in the TIM, or else it has retrieved all buffered frames, the mobile station can resume the sleeping mode by asking AP for a new wakeup counter $c_j(t)$. In the LAWS+SAF and LAWS+SQLF schemes, the mobile station misses the beacon can not show up to retrieve the buffered data in its turn. The next station in the access sequence can send PS-Poll frames to get back its buffered data if it finds that the medium has been idle for longer than the distributed coordination function inter-frame space (DIFS).

# Chapter 4

# Power Saving Protocols for IEEE 802.11 Ad Hoc Mode

In this chapter, we give a novel self-configuring power-saving protocol for wireless one-hop ad hoc networks, the SCPS. According to IEEE 802.11 WLANs standard, a station may enter a special power-saving (PS) mode. SCPS allows all stations in the PS mode to adjust their wakeup schedules whenever a station enters or exits the PS mode. The adjustment can balance the number of wakeup stations in each beacon interval so that the contention for transmission medium and the collisions in transmission will be ameliorated, which results in more efficient energy usage. Simulation results show that SCPS successfully balances the number of stations that wake up in each beacon interval, increases the sleep ratio, and reduces the collision probability. The combined effect reduces total energy consumption.

This chapter is organized as follows. In Section 4.1, we state our research problem of power-saving in the ad hoc network. The proposed SCPS approach is presented in Sections 4.2. In Section 4.3, we present the simulation results of SCPS and the summary is in section 4.4.

# 4.1 Problem statement

The QEC [30] approach which mentioned in chapter 2 does not balance the number of wakeup stations in each beacon interval, it cannot avoid high packet collision rate. In order to balance the number of wakeup stations in beacon intervals, we need another mechanism approach which can allow every station not only to decide its own listen-interval as in the 802.11 infrastructure mode but also to choose a suitable wakeup schedule. Each station wakes up once per listen-interval, rather than per beacon interval. Thus, a station wakes up less frequently and sleeps longer. Besides, by efficiently balance the numbers of wakeup stations during beacon intervals, the collision (and hence re-transmission) of packets can be reduced.

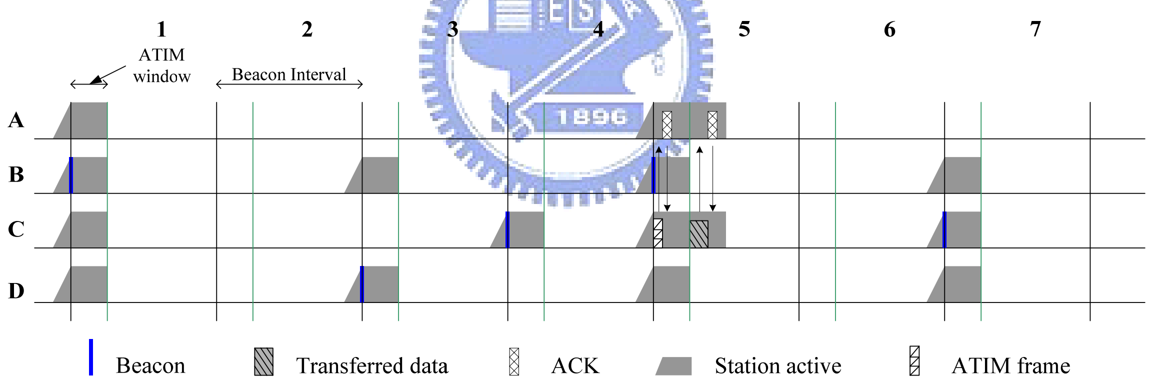



**Figure 4-1 the power-saving operation of SCPS**

As the Figure 4-1 shows, assume that there are four stations, A, B, C, and D with listen-intervals 4, 2, 3, and 2, respectively. That is, station A wakes up every 4 beacon intervals, B wakes up every 2 beacon intervals, and so on. Assume that, in the 4th beacon interval, C wishes to send data to A. Because A does not wake up in this beacon interval, C cannot transmit the data to A at this time. C needs to wait until the 5$^{th}$ beacon interval in which A will wake up. If C knows A's wakeup schedule, C can sleep until the 5th beacon interval instead of waking up at every beacon interval to check A's availability.

To achieve the mechanism of Figure 4-1, every station has a copy of the wakeup schedule of every other station that is in the PS mode. Before we can use this mechanism, we need to deal with three issues. The first is synchronizing the timers. Because stations do not wake up in every beacon interval, stations' timers may become out of synchronization. The second is maintaining a consistent wakeup table (which is the collection of the wakeup schedules of all stations in the PS mode) among all stations. Because there is not a central coordinator in an ad hoc network, a distributed mechanism is needed to keep all the copies of the wakeup table consistent. The third is evenly arranging stations to wake up in beacon intervals in order to alleviate the contention.

# 4.2 The SCPS approach

The motivation of SCPS comes from the observation of Figure 4-1. In the following subsections, we discuss how to conquer the three problems. Timer synchronizing is in section 4.2.2. Maintaining a consistent wakeup table (which is the collection of the wakeup schedules of all stations in the PS mode) among all stations is in section 4.2.3. And finally problem that evenly arranges stations to wake up in beacon intervals is in 4.2.4.

## 4.2.1 The wakeup information of other stations

In SCPS, every station needs to maintain the wakeup information of all the other stations, which is kept in the Wakeup Information Table (WIT). Table 4-1 is an example of WIT. Each entry comprises four fields: station ID (SID), MAC address, listen-interval

($\ell$), and wakeup count (WC). SID is a unique value chosen by the station to identify itself. The range of a SID is 0–127. When a station enters the PS mode, it announces its SID to the network. Other stations can bind this SID to the announcer's MAC address. The listen-interval, denote as $\ell$, of a station is the number of beacon intervals for which the station sleeps between two adjacent wakeups. The wakeup count (WC) is the number of beacon intervals for which the station will sleep before the next wakeup. Note that WC cycles through from $\ell - 1$ down to 0.

**Table 4-1 the wakeup information table**

| SID | MAC address | $\ell$ | WC |
|---|---|---|---|
| 2 | 00-06-AE-EF-56-21 | 5 | 0 |
| 4 | 00-02-3B-D3-A6-21 | 4 | 3 |
| 8 | 00-06-4E-B7-65-67 | 4 | 0 |
| 5 | 00-03-E9-03-24-5F | 4 | 2 |
| 9 | 00-01-D2-03-71-21 | 4 | 1 |

Since an ad hoc network lacks a central coordinator, WIT maintenance is done during the beacon process. We add a new piece of information, called the Station Wakeup Information (SWI), in the beacon frames (Figure 4-2). The station that wins the right to send a beacon frame broadcasts the SWI information, which contains the wakeup schedules of all stations in the PS mode. Thus, newly joined stations can obtain the wakeup schedules of other stations that are in the PS mode. We call the station that sends the beacon frame as the beacon sender. The SWI includes the following fields:

- PS (power-saving) status: This one-byte field indicates the state of the beacon sender. If the PS status is JOIN or LEAVE, it means that the beacon sender is going to enter or exit the PS mode, respectively; otherwise, its status is NORMAL.

- SID bitmap: The SID bitmap is similar to the virtual bitmap in the traffic indication map (TIM) in the 802.11 infrastructure mode. It consists of 128 bits. Each bit is tied to a station ID. When a SID is occupied by a station (which must be in the PS mode), the bit tied to the SID is 1; otherwise, the bit is 0. In practice, a one-hop ad hoc network comprises 25–40 stations. Thus, 128 bits should be enough.

- Wakeup Parameter Set (WPS): This set includes the listen-interval and wakeup counter of every station that is in the PS mode. The information is listed in the ascending order of the SIDs.

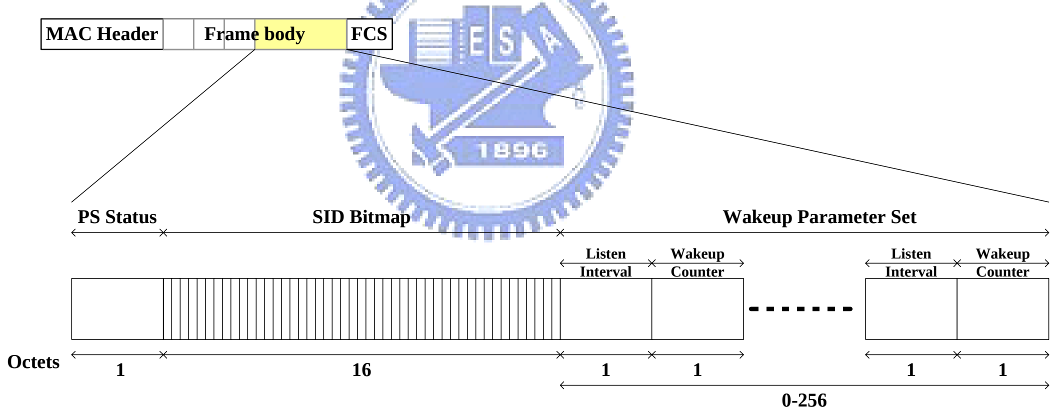



**Figure 4-2 the format of station wakeup information (SWI)**

In order to allow a station with new SWI information (such as those who want to enter or exit the PS mode) to preempt other stations when competing for the transmission medium, the rules for selecting a backoff window size are modified in SCPS. The station that wants to enter or exit the PS mode chooses its backoff window size randomly between 0 and $\rho/2$, where $\rho$ is two times the minimum backoff window size specified in IEEE 802.11. Other stations choose their backoff window sizes randomly between $\rho$ /2 and $\rho$ .

Obviously, the WIT in each beacon frame consumes bandwidth. The size of a WIT is 1 + 16 + 2*u* bytes, where *u* is the number of stations in the PS mode. If the physical transmission rate is 1 Mbps, the extra time for transmitting the WIT is less than 2.2 ms when *u* =128 stations, which is about 2% of the bandwidth if a beacon interval lasts for 100 ms. For a common network (containing 25-40 stations) the overhead is 0.54-0.78% of the bandwidth.

## 4.2.2 Timer synchronization

In order to synchronize stations' timers, we define the Timer Synchronization Beacon Interval (TSBI) as a beacon interval in which all stations will wake up and compete to serve as the beacon sender. All stations will synchronize their timers with that of the beacon sender. The beacon of TSBI is called as the Timer Synchronization Beacon. The number of beacon intervals between two consecutive Timer Synchronization Beacons is called the Timer Synchronization Period (TSP).

**Table 4-2 the number of beacon intervals needed or the clock to draft for a DIFS**

| Accuracy | 0.01% | 0.005% | 0.003% | 0.002% | 0.0016% | 0.001% |
|---|---|---|---|---|---|---|
| Beacon intervals | 2.50 | 5.00 | 8.33 | 12.50 | 15.00 | 25.00 |

TSP affects the efficiency of power-saving operations and timer synchronization. Long TSP allows stations to sleep longer but also increases the failure probability on timer synchronization. Table 4-2 shows the number of beacon intervals needed for the clock to draft for more than the duration of a DIFS with various clock accuracy ratios ranged from 0.001% to 0.01% in an 802.11b Direct Sequence Spread Spectrum (DSSS) physical layer. The worst clock accuracy ratio specified in 802.11 is 0.01%. Because a station has to wait for a DIFS period before it counts down the (random) backoff time, we require the clock skew should not exceed a DIFS period. For instance, if the clock accuracy ratio is 0.001%, all the clocks should be synchronized every 25 (or less) beacon intervals. Determining a suitable TSP needs experiments. In our simulation study, the clock accuracy ratio is assumed to be 0.001%. Thus, TSP is 25 beacon intervals. The reason will be given in the simulation section. Note that the TSBI is for the sole purpose of timer synchronization. The stations will not send any frames except the Timer Synchronization Beacon frame during a TSBI. After that, the stations go to sleep.

## 4.2.3 Maintaining a consistent wakeup table

In this section, we show how to maintain a consistent wakeup table while stations enter or exit the PS mode. Before a new station, say $S_j$, enters the PS mode, it needs to be a beacon sender to send SWI to all other stations. Then, $S_j$ stays wakeup until it receives a beacon frame with the NORMAL PS status. At that time, $S_j$ knows that no station is entering or exiting the PS mode. It then builds a WIT table based on the SWI fields of the received beacon frame. In the next beacon interval, $S_j$ chooses a SID, sets the PS status to JOIN, and prepares the SWI fields for the beacon frame. $S_j$ uses [0, $\rho$/2] as the backoff window size to compete for the medium for transmitting the beacon frame. Once $S_j$ successfully sends out its beacon frame, it will serve as the beacon sender for the next k beacon intervals (starting from the current beacon interval). Note that k should be large enough to ensure that all stations in the PS mode will wake up and receive $S_j$'s beacon frame at least once during these k beacon intervals. *k* can be determined as follows:

$$k = \min(\ell_{\max}, \omega)$$

where $\ell_{\max}$ is the maximum listen-interval in the current WIT table and $\boldsymbol{\omega}$ is the number of beacon intervals from the current beacon interval to the next TSBI. After k beacon intervals, all stations in the PS mode are notified and they can update their WIT tables. Other stations that also wish to be the beacon senders but fail to win the transmission medium need to wait for the next beacon frame with the NORMAL PS status.

Similarly, when a station, say $S_e$, wants to exit the PS mode, it removes its wakeup schedule from the beacon frame, sets its PS status to LEAVE, and competes for the transmission medium with a backoff window size randomly chosen from [0, $\rho$/2].

**Table 4-3 a sequence of $w_i(t)$ and $n(t)$**

| ***t*** | ***1*** | ***2*** | ***3*** | ***4*** | ***5**** | ***6*** | ***7*** | ***8*** | ***9*** | ***10*** | ***11*** | ***12*** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $w_A(t)$ | 0 | $1^N$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| $w_B(t)$ | 0 | 0 | 1 | 0 | 0 | 1 | $1^L$ | $1^L$ | $1^L$ | $1^L$ | - | - |
| $w_C(t)$ | 0 | 0 | $1^N$ | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | $1^N$ |
| $w_D(t)$ | 0 | 0 | 1 | 0 | 0 | $1^N$ | 0 | 0 | 1 | 0 | 0 | 1 |
| $w_E(t)$ | - | - | 1 | $1^J$ | $1^J$ | 0 | 1 | 0 | 0 | 0 | $1^N$ | 0 |
| $n(t)$ | 0 | 1 | 4 | 1 | 1 | 4 | 2 | 1 | 3 | 2 | 1 | 2 |

After $S_e$ sends its beacon frame, it continues to serve as the beacon sender for the next k beacon intervals to notify all other stations in the PS mode, where k is given in equation (1) above. When a station wakes up and receives a beacon frame with a JOIN or LEAVE PS status, it uses the SWI information in the beacon frame to update its own WIT table. Because the MAC address of the beacon sender is included in the beacon frame, other stations can bind the beacon sender SID to its MAC address.

Table 4-3 shows an example of a station entering and exiting the PS mode. We use English letters A, B, etc. to represent SIDs. In Table 2, $w_i(t)$ = 1 if station $i$ wakes up at beacon interval t; $w_i(t)$ = 0, otherwise. For each beacon interval, we mark the beacon sender with its PS status as a superscript. That is, $w_i(t) = 1^x$ indicates that, during interval $t$, station $i$ is the beacon sender and the PS status is NORMAL (N), JOIN (J), or LEAVE (L).

Let $n(t)$ denote the number of stations waking up at beacon interval $t$. Then, $n(t)$ can be found by :

$$n(t) = \sum_{i \in S} w_i(t)$$

where $S$ is the set of all stations in PS mode. There are 5 stations in this example. Assume that the 5th beacon interval is a TSBI. During the 1st beacon interval, 4 stations A, B, C, and D are in the PS mode. Their listen-intervals are 4, 3, 3, and 3, respectively. Assume that, at the 3rd beacon interval, station E wants to enter the PS mode with its listen-interval set to 4. Station E receives a beacon frame issued by station C with the NORMAL PS status. So it uses the SWI information in this beacon frame to update its WIT table. During the 4th beacon interval, station E adds its wakeup schedule to the beacon frame and sets the PS status to JOIN. It employs a short delay to win the opportunity to serve as the beacon sender. It continues to serve as the beacon sender until the 5th beacon interval, which is the TSBI. Note that in this case $\ell_{max} = 4$, $\boldsymbol{\omega} = 2$; and that $k = \min(\ell_{max}, \boldsymbol{\omega}) = 2$.

Assume that, during the 5th beacon interval, station B wants to exit the PS mode. Because the received beacon frame is not in the NORMAL status, B keeps on monitoring the medium. During the 6th beacon interval, B detects that the PS status of the beacon frame is NORMAL. It competes for the medium during the 7th beacon interval. Station B continues to serve as the beacon sender until the 10th beacon interval. Stations that wake up and receive the beacon frames from B make use of the SWI information to update their respective WIT tables. It is obvious that the number of wakeup stations in each beacon interval is not the same. If stations' wakeup schedules are properly arranged so that the number of wakeup stations in each beacon interval is roughly the same, the contention for the transmission medium will be reduced. Therefore, we hope to balance the number of wakeup stations in each interval.

To balance the number of wakeup stations in each beacon interval, we need to adjust the WC of the sleeping stations when there is a station entering or exiting the PS mode. In the following subsections, we show how SCPS balances the number of stations.

### 4.2.4 A station enters the PS mode

Consider a wireless ad hoc network with six stations, A, B, C, D, E, and F. Assume they all are sleeping initially. Their listen-intervals are 4, 3, 3, 3, 3, and 4, respectively. We will use the sequence of $w_i(t)$'s and the total number of wakeup stations $n(t)$ for t =1, 2, . . . , 18, in Table 4-4 as our example.

Assume that the 12th beacon interval is a TSBI and the stations' first wake up at the 4th, 3rd, 2nd, 1st, 1st, and 3rd beacon intervals, respectively. Assume that station *J* wants to enter the PS mode at beacon interval 4. Furthermore, assume station *J*'s listen-interval is 3 and *J* randomly chooses a wakeup schedule (1, 0, 0) (this schedule means that *J* wakes up once every three beacon intervals). The row of $n(t)_{bf}$ shows the number of wakeup stations in each beacon interval before station *J* enters the PS mode and the row of $n(t)_{af}$ shows the numbers after station *J* enters the PS mode.

**Table 4-4 a sequence of $w_i(t)$ and $n(t)$**

| $t$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12* | 13 | 14 | 15 | 16 | 17 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $w_A(t)$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| $w_B(t)$ | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| $w_C(t)$ | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $w_D(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| $w_E(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| $w_F(t)$ | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| $w_J(t)$ | - | - | - | $1^J$ | $1^J$ | $1^J$ | $1^J$ | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| $n(t)_{bf}$ | 2 | 1 | 2 | 3 | 1 | 1 | 3 | 2 | 1 | 2 | 2 | 6 | 2 | 1 | 2 | 3 | 1 | 1 |
| $n(t)_{af}$ | 2 | 1 | 2 | 4 | 2 | 2 | 4 | 2 | 1 | 3 | 2 | 7 | 3 | 1 | 2 | 4 | 1 | 1 |

Before station *J* enters the PS mode, the maximum number of $n(t)$ in a beacon interval is 3, which occurs at beacon intervals $4^{th}$, $7^{th}$, and $16^{th}$, respectively. With the unfortunate choice of the wakeup schedule (1, 0, 0), the maximum number of $n(t)$ becomes 4 after station *J* enters the PS mode. It is because that its wakeup time includes the $4^{th}$, $7^{th}$, and $16^{th}$ beacon intervals.

**Table 4-5 a sequence of $w_i(t)$ and $n(t)$ for station *J* with first wakeup time $t = 5$**

| $t$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $w_A(t)$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| $w_B(t)$ | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| $w_C(t)$ | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $w_D(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| $w_E(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| $w_F(t)$ | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| $w_J(t)$ | - | - | - | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $n(t)_{bf}$ | 2 | 1 | 2 | 3 | 1 | 1 | 3 | 2 | 1 | 2 | 2 | 2 | 2 | 1 | 2 | 3 | 1 | 1 |
| $n(t)$ | 2 | 1 | 2 | 3 | 2 | 1 | 3 | 3 | 1 | 2 | 3 | 3 | 2 | 2 | 2 | 3 | 2 | 1 |

On the other hand, if *J* chooses the wakeup schedule (0, 1, 0), the maximum of $n(t)$ is still 3, as shown in Table 4-5 (Only the stations' periodical wakeup schedules are show in it). Therefore, by choosing an appropriate wakeup schedule for *J*, the maximum number of competing stations at any interval can be minimized.

By observing the sequence of $n(t)_{bf}$ in Table 4-5, we may find that a pattern repeats every 12 beacon intervals, e.g., ($n(4), \ldots ; n(15)$) = (3, 1, 1, 3, 2, 1, 2, 2, 2, 2, 1, 2)

The length of this repeating pattern, *r*, must be a factor of the least common multiple (*lcm*) of the listen-interval $\ell_i$, where $i \in S$. For example, the listen-intervals of stations, A, B, C, D, E, F and *J* are 4, 3, 3, 3, 3, 4 and 3, respectively, in Table 4-5.

Then r =*lcm*{4; 3; 3; 3; 3; 4; 3} = 12

In addition, for a station with listen-interval $\ell$ there are exactly $\ell$ wakeup patterns:

(…0, 0, 1) , (…0, 1, 0), (…1, 0, 0),. . .

Consider station *J* of Table 4-5, whose listen-interval is 3. *J* will carry out the following computations to find out the maximum *n(t)* for each pattern p, which is denoted as $n_p^*$:

Case 1. Assume ($w_J(4), w_J(5), \ldots, w_J(15)$) = (0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1). Then,

($n(4), n(5), \ldots, n(15)$) = (0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1)+(3, 1, 1, 3, 2, 1, 2, 2, 2, 2, 1, 2)

= (3, 1, 2, 3, 2, 2, 2, 2, 3, 2, 1, 3) and $n_1^*$ = max{$n(4), n(5), \ldots, n(15)$} = 3

Case 2. Assume ($w_J(4), w_J(5), \ldots, w_J(15)$) = (0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0). Then,

($n(4), n(5), \ldots, n(15)$) = (0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0) + (3, 1, 1, 3, 2, 1, 2, 2, 2, 2, 1, 2)

= (3, 2, 1, 3, 3, 1, 2, 3, 2, 2, 2, 2) and $n_2^*$ = max{$n(4), n(5), \ldots, n(15)$} = 3

Case 3. Assume ($w_J(4), w_J(5), \ldots, w_J(15)$) = (1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0). Then,

$(n(4), n(5), \ldots, n(15))$ = (1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0) + (3, 1, 1, 3, 2, 1, 2, 2, 2, 2, 1, 2) = (4, 1, 1, 4, 2, 1, 3, 2, 2, 3, 1, 2) and $n_3^* = \max\{n(4), n(5), \ldots, n(15)\} = 4$

Finally, station $J$ chooses a pattern with the minimum $n_x^*$ as its schedule. In this example, $n_2^*$ =min{ $n_1^*$, $n_2^*$, $n_3^*$ } = 3. J chooses the first case as its schedule.

Formally, we define the wakeup scheduling problem (WSP) as follows: Assume station $J$ wants to enter the PS mode. Given a set of m sleeping stations at beacon interval $t$, assume each station i has the wakeup schedule $\langle w_i(t+1), w_i(t+2), w_i(t+3), \ldots, w_i(t+\ell_i) \rangle$. We want to assign a wakeup schedule $\langle w_J(t+1), w_J(t+2), w_J(t+3), \ldots \rangle$ to station $J$ such that max $\{n(t+k) | k = 1,2,3\ldots\}$ is minimized, where $n(t+k)$ is defined as $\sum_{i \in S \cup \{J\}} w_i(t+k)$, for k =1, 2, ….

The WSP problem may be solved with the following method:

1. Determine the length of the repeating pattern:

$$r = lcm\{\ell_i \mid i \in S \cup \{J\}\}$$

2. Generate the set of all possible wakeup patterns:

$$W = \{\langle w_J^a(t+1), w_J^a(t+2), \ldots, w_J^a(t+r) \rangle \mid a = 1,2,\ldots,\ell_J\}$$

3. For each wakeup pattern $\langle w_J^a(t+1), w_J^a(t+2), \ldots, w_J^a(t+r) \rangle$ where $a = 1,2,\ldots,\ell_J$, compute $n^{(a)} = \max\{n_k = \sum_{i \in S} w_i(t+k) + w_J^a(t+k), \quad k = 1,2,\ldots,r\}$

4. Choose a wakeup pattern $\langle w_J^*(t+1), w_J^*(t+2), \ldots, w_J^*(t+r) \rangle$ such that $n^* = \min\{n^{(a)} \mid a = 1,2\ldots,\ell_J\}$ . That is, the wakeup pattern

$\left\langle w_J^*(t+1), w_J^*(t+2),..., w_J^*(t+r)\right\rangle$ balances the total number of wakeup stations *n(s)*, for *s = t+1, t+2, …*

The time complexity for computing a minimum sequence depends on the least common multiple of the listen-intervals. Assume that *R* is the current least common multiple and the listen-interval of a newly joined station is *L*. The time complexity for the new station to examine all possible patterns is *O(R*L)* and that for finding out the minimum (that is, $n^*$) is *O(L)*. Thus, the total time complexity is *O(R*L) + O(L) = O(R*L)*.

## 4.2.5 A station exits the PS mode

When a station, says $S_e$, exits the PS mode, the number of wakeup stations in each beacon interval may become unbalanced. Table 4-6 illustrates such a scenario. The number of wakeup stations, shown in the row of *n(t)* in Table 4-6, is not balanced after station C exits the PS mode at the 5th beacon interval. This is because stations D and E wake up at the same beacon interval. If one of them re-arranges its wakeup sequence, the number of wakeup stations in each beacon interval can become balanced again. The unbalanced phenomenon may become more severe if many stations exit the PS mode but no station enters the PS mode in a period of time. Thus, it is necessary to re-balance the number of wakeup stations after some station exits the PS mode.

Note that all stations have to wake up in the TSBI to synchronize their timers. Thus, TSBI is the right time to re-balance the number of wakeup stations. Thus, we embed the re-balance mechanism in TSBI.

Table 4-6 a sequence of $w_i$(t) and *n(t)* for station C existing the PS mode at time *t*=5

| *t* | *1* | *2* | *3* | *4* | *5* | *6* | *7* | *8* | *9* | *10* | *11* | *12** | *13* | *14* | *15* | *16* | *17* | *18* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $w_A(t)$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| $w_B(t)$ | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| $w_C(t)$ | 0 | 1 | 0 | 0 | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $w_D(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| $w_E(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| $w_F(t)$ | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| $n(t)$ | 2 | 1 | 2 | 3 | 0 | 1 | 3 | 1 | 1 | 2 | 1 | 2 | 2 | 0 | 2 | 3 | 0 | 1 |

At TSBI, say beacon interval $t + 1$, each station i solves the WSP problem as if it wants to enter the PS mode again and calculates the optimal value $n_i^*$. Each station i can also calculate the optimal values $n_j^*$ (where $j \in$ S-{i}) for all other stations *j*. Comparing $n_i^*$ with other $n_j^*$s, station *i* can learn if $n_i^*$ is the smallest. The station with the least $n_j^*$ wins the right to serve as the beacon sender. It can re-balance the number of wakeup stations in a beacon interval as well as synchronize all stations' timers. In this case, there are more than one station with the same least *n**, the one with the smallest station ID.

Table 4-7 a sequence of $w_i(t)$ and *n(t)* when C exists the PS mode at time *t*=5

| *t* | *1* | *2* | *3* | *4* | *5* | *6* | *7* | *8* | *9* | *10* | *11* | *12** | *13* | *14* | *15* | *16* | *17* | *18* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $w_A(t)$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| $w_B(t)$ | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| $w_C(t)$ | 0 | 1 | 0 | 0 | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $w_D(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | **0** | **0** | **1** | **0** | **0** | **1** | **0** |
| $w_E(t)$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| $w_F(t)$ | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| $n(t)$ | 2 | 1 | 2 | 3 | 0 | 1 | 3 | 1 | 1 | 2 | 1 | **2** | **1** | **1** | **2** | **2** | **1** | **1** |

Re-balance occurs at $t$ = 12

Table 4-7 shows an example of the re-balance mechanism. Assume that at the 5<sup>th</sup> beacon interval, station C exits the PS mode. All stations in the PS mode learn C exited the PS mode after the 8<sup>th</sup> beacon interval. The 12<sup>th</sup> beacon interval is a TSBI. All stations wake up and solve the WSP problems. Then, every station calculates $n_A^* = 3$, $n_B^* = 3$, $n_D^* = 2$, $n_E^* = 2$, and $n_F^* = 3$. Station D wins the right to serve as the beacon sender in the 12th beacon interval. Its wakeup schedule ($w_D(12)$, $w_D(13)$, $w_D(14)$, . . .) is set to (0, 0, 1, . . .). Note that the maximum number of wakeup stations in each beacon interval after the 12th becomes 2 rather than 3.

# 4.3 Simulation results

## 4.3.1 Performance metrics and environment setup

The following four metrics are studied in the simulation of SCPS:

1. The number of wakeup stations in the ATIM window of a beacon interval. Generally, the probability of a collision during the competition for the transmission medium is influenced by the number of wakeup stations that contend during the ATIM window. If we can carefully control the number of wakeup stations, the collision probability can be reduced.

2. Sleep ratio. Sleep ratio is a common index for evaluating power-saving mechanisms. The sleep ratio is defined as the ratio of the amount of time that a station sleeps to the total amount of time. The longer a station sleeps the less energy it consumes. The sleep ratio is influenced by the traffic generation rate. Thus, we adjust the traffic generation rate in our simulation to observe the variance of the sleep ratio.

3. The average queuing delay. Usually, increasing the sleep time also increases the queuing delay. A longer listen-interval will result in more efficient energy usage but it also increases the packets' queuing time. The average queuing delay for different

listen-intervals can be shown by this metric. The delay is calculated from the time when a packet is put into the queue until it is sent out.

4. The average packet drop ratio. A station drops its queued packets if the packets cannot be transmitted before the expiration time or they collide with other stations' packets while they are transmitted. Dropped packets need to be re-sent. This wastes energy. Hence, the average packet drop ratio is a good index of energy efficiency.

We compare these four metrics in SCPS, 802.11 PS mode (denoted as PSM in the following figures), DPSM [22], and QEC [30] with simulation. The beacon interval is fixed at 100 ms. The ATIM window size for DPSM ranges from 2 to 26 ms. For the other two protocols, the ATIM window size is 25 ms. Two different grid sizes are implemented for QEC: QEC2 (with a 2×2 grid) and QEC4 (with a 4×4 grid). In QEC2, the average listen-interval is 4/3 beacon intervals, that is, a station wakes up three times every four beacon intervals. Similarly, the average listen- interval is 16/7 beacon intervals for QEC4.In SCPS, the length of each station's listen-interval is randomly chosen from 1 to 4 beacon intervals. For a fair comparison, we choose two configurations for SCPS:

SCPSa Half of the stations wake up every beacon interval. The remaining stations wake up once every two beacon intervals. The average listen-interval is, thus, 4/3 beacon intervals. SCPSa will be compared against QEC2.

SCPSb Three fourths stations wake up once every two beacon intervals. The remaining one fourth stations wake up once every four beacon intervals. The average listen-interval is, thus, 16/7 beacon intervals. SCPSb will be compared against QEC4.

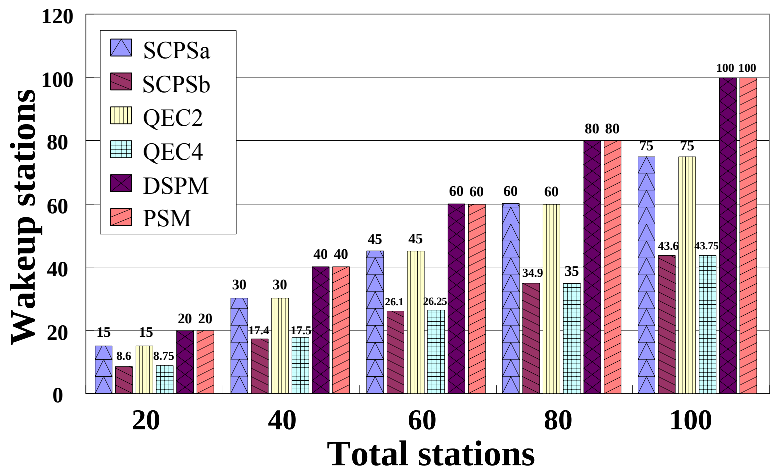
The remaining simulation parameters are as follows: The physical data transmission rate is 11 Mbps. The constant bit rate (CBR), 4-packets/sec, is used for traffic generation. The size of a packet is 8000 bits. A packet is dropped if its queuing time is more than 1.6 s, which is 4 times the maximum listen-interval. The maximum buffer size of each station is 20 k-bytes. The total simulation time was 1800s. The performance metrics are averaged over 1000 runs.

The clock accuracy ratio in our simulation is assumed to be 0.001%. In QEC4, two stations will synchronize their clocks when they wake up at the same beacon interval. The difference between two consecutive clock synchronizations of two stations is at most 15 beacon intervals. According to Table 1, we have to use clocks with accuracy ratio no worse than 0.0016%. In our simulation, clock accuracy ratio is fixed at 0.001% and, hence, TSP is 25 beacon intervals.

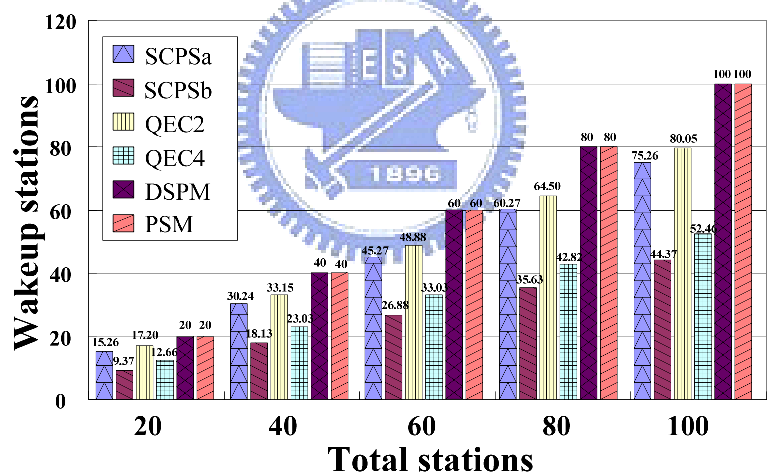
### 4.3.2 Experimental results

Figure 4-3 (a) shows the average number of wakeup stations in an ATIM window. Because in both 802.11 PSM and DPSM, all stations in the PS state need to wake up in every beacon interval, the number of wakeup stations is equal to the total number of stations in the PS state. On the other hand, for QEC and SCPS, the average numbers of wakeup stations in a beacon interval obtained from the experiments are identical. This is consistent with the probability distribution.

Figure 4-3(b) shows the maximum number of wakeup stations in an ATIM window. Consider the case of 100 total stations. The maximum number of wakeup stations in QEC4 is 52.46 while that in SCPSb is 44.37. It is obvious that SCPSb incurs lighter contention than QEC4 in the worst case. Similarly for QEC2 vs. SCPSa.




(a) Average number of wakeup stations in an ATIM window




(b) Maximum number of wakeup stations in an ATIM window

**Figure 4-3 the average and the maximum wakeup stations in an ATIM window**

Figure 4-4 shows the relationship of the sleep ratio and the total number of stations. As expected, the sleep ratio in every protocol decreases as the number of stations grows. The original IEEE 802.11 protocol (PSM) has the worst (that is, lowest) sleep ratio. Its sleep ratio is about 45%. In QEC and SCPS, the sleep ratios are more than 72%. The sleep ratio in SCPSa is almost the same as that in QEC2. Similarly, the sleep ratios in SCPSb and QEC4 are also almost identical. For DPSM, because the initial ATIM window is short (2 ms), DPSM has the best sleep ratio. However, as the number of stations increases, the sleep ratio in DPSM decreases rapidly. This is due to the fact that the narrow bandwidth provided by short ATIM windows cannot satisfy the transmission requests as there are more and more stations. Every station has to increase its ATIM window size and, hence, the sleep ratio decreases.

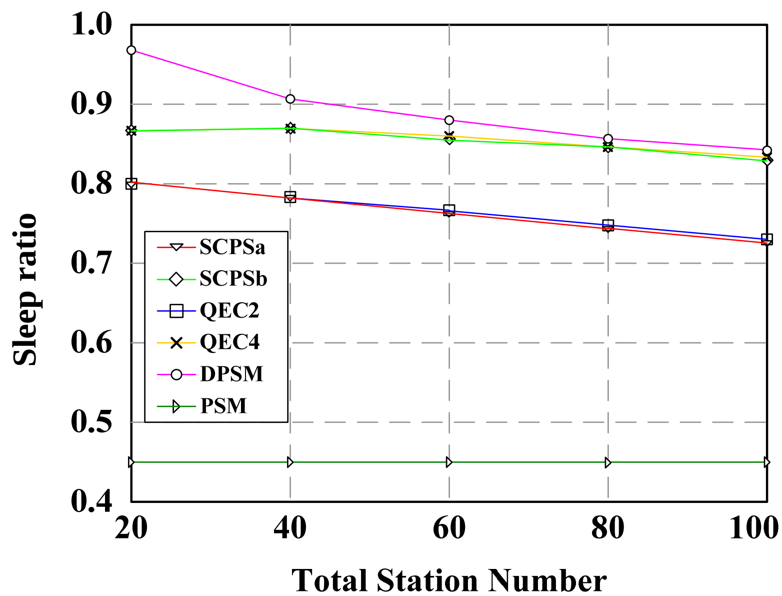



**Figure 4-4 the relationship of the sleep ratio and the total number of stations**

Figure 4-5 shows the average queuing delay of a packet. When there are more stations in the network, the competition for the medium gets hotter. Thus, the queuing delay becomes longer. Among the tested protocols, PSM has the shortest delay because stations wake up more often. In DPSM, because of the slow adaptation of the ATIM window size, the queuing delay grows rapidly as the number of stations exceeds 40. This results from the short initial ATIM window, which limits the number of ATIM frames that can be announced and causes the packets congested in the buffer. Consequently, the queuing delay becomes longer. SCPSa has almost the same queuing delay as QEC2 (their curves completely overlap and are hard to distinguish in Figure 4-5). The delay in SCPSb is shorter than that in QEC4 by 30ms. SCPSb reduces 20% of the queuing delay of QEC4 but its sleep ratio is almost the same as that in QEC4.

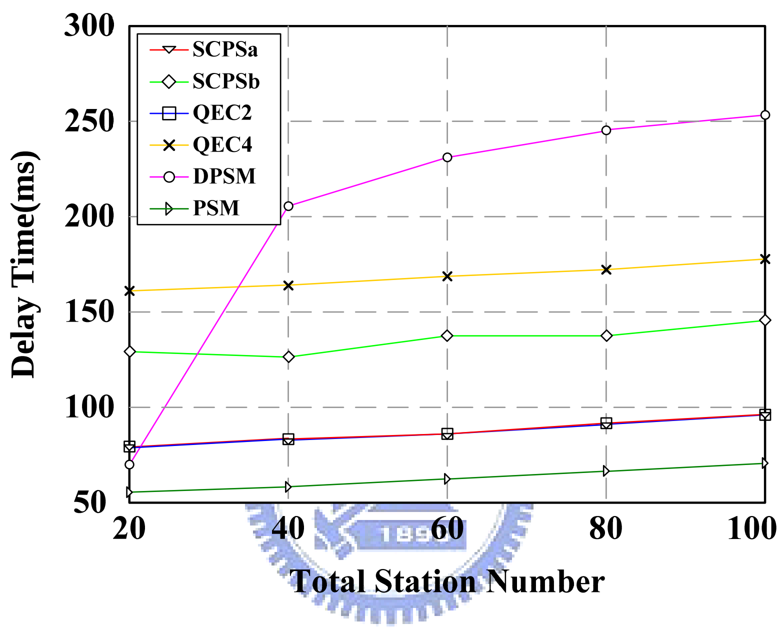



**Figure 4-5 the relationship of average packet queuing delay and the total station number**

Figure 4-6 shows the number of dropped packets. With the constant traffic generation rate of 4 packets per second, the total number of transmitted packets per station is 7200 packets (4 packets/s x 1800 s). Because of the short initial ATIM window in DPSM, the number of ATIM frames that can be announced is very limited. Thus, many packets are dropped even though there is still available bandwidth in the data transmission window. Furthermore, all stations have to wake up in every beacon interval in DPSM. This causes higher probability of collision. So, DPSM drops more packets.

DPSM has the highest packet drop ratio. The packet drop ratios of SCPSa, QEC2, and PSM are almost the same. The drop ratio in SCPSb is less than that in QEC4 because the number of wakeup stations in SCPSb is more balanced. By minimizing the maximum number of wakeup stations in every beacon interval, SCPSb reduces the number of packet collisions by alleviating the medium contention.

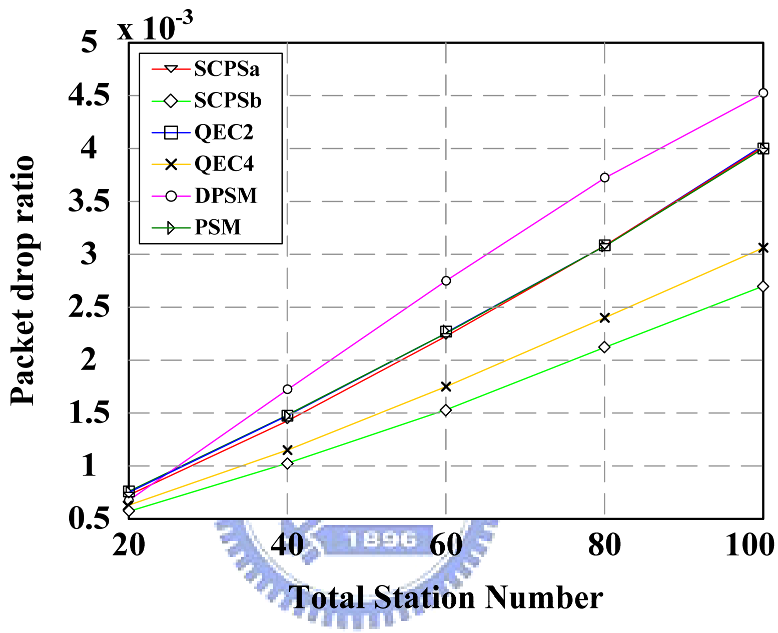



**Figure 4-6 the relationship of packet drop ratio and the total number of stations**

In Figure 4-7, sleep ratios of all tested protocols decrease as the traffic generation rate increases. Because stations which transmit or receive data in a data transmission window need to stay awake during the whole beacon interval, 802.11 PSM has the worst sleep ratio. The stations almost cannot sleep when the traffic generation rate is higher than 10-packets/s.

Figure 4-8 gives the ratio of dropped packets in each approach. Without distributing the wakeup stations evenly among all beacon intervals, DPSM and 802.11 PSM drop more packets. Furthermore, because of the slow adaptation of the ATIM window size, DPSM drops even more packets than 802.11 PSM while traffic rate is less than 6-packets/s.

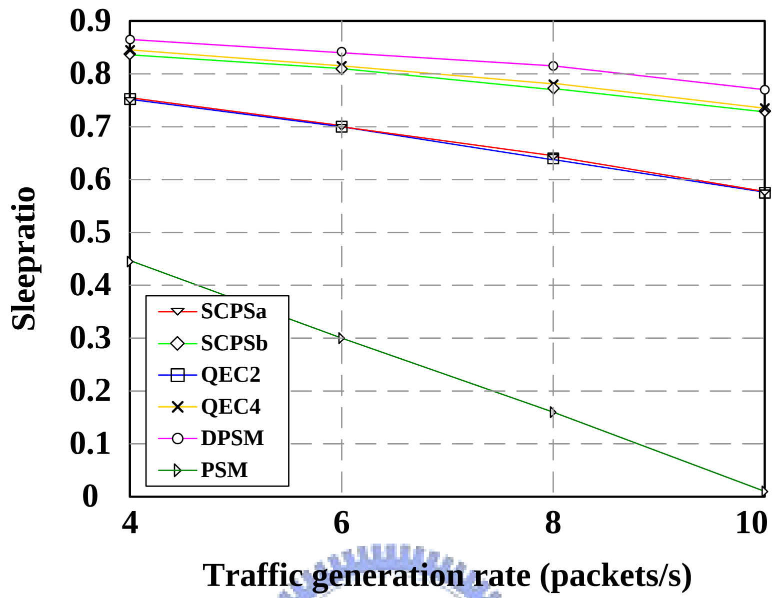



**Figure 4-7 the relationship between sleeping ratio and traffic generation rate (60 stations; packet size: 8000 bits)**

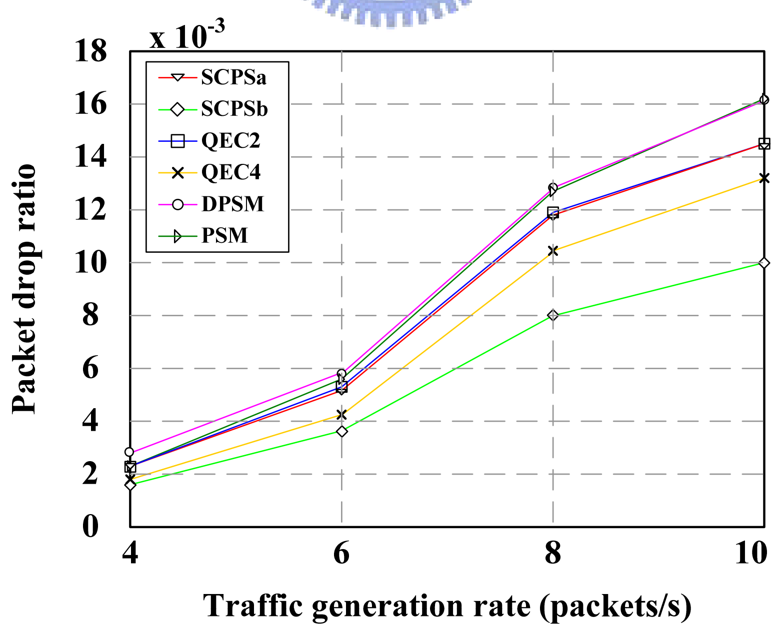



**Figure 4-8 the relationship between sleeping ratio and traffic generation rate**

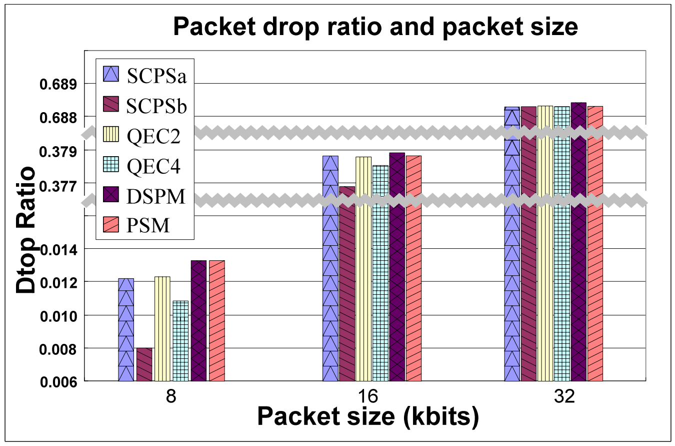



**Figure 4-9 influence of packet size (60 stations; traffic generation rate: 8 packets/s)**

Figure 4-9 shows the impact of packet sizes. While the packet size is 8000 bits, the bandwidth capacity can still satisfy the requests of all stations. Packets are dropped due to packet collision. As the packet size increases to 16,000 bits or more, the bandwidth capacity cannot satisfy all requests. In this case, packets are dropped because packets wait too long in the queue. The number of dropped packets is the same for all tested protocols (because we use the same traffic generation rate and the same bandwidth).

The throughput and sleep ratio for SCPSb are given in Figure 4-10. The packet size is 16,000 bits. As expected, the sleep ratio increases and the throughput ratio decreases while a listen-interval lasts longer. For stations with listen-interval equal to 1 beacon interval, the sleep ratio is only 30% but the throughput ratio is higher than 95%. On the other hand, for stations with listen-interval equal to 4, the sleep ratio approximates 80%, but the throughput ratio only 55%.

Finally, figure 4-11 shows the average waiting time that a station spends on switching to the PS mode in SCPSb. The X-axis is the number of stations that switch to the PS mode every 40 beacon intervals.

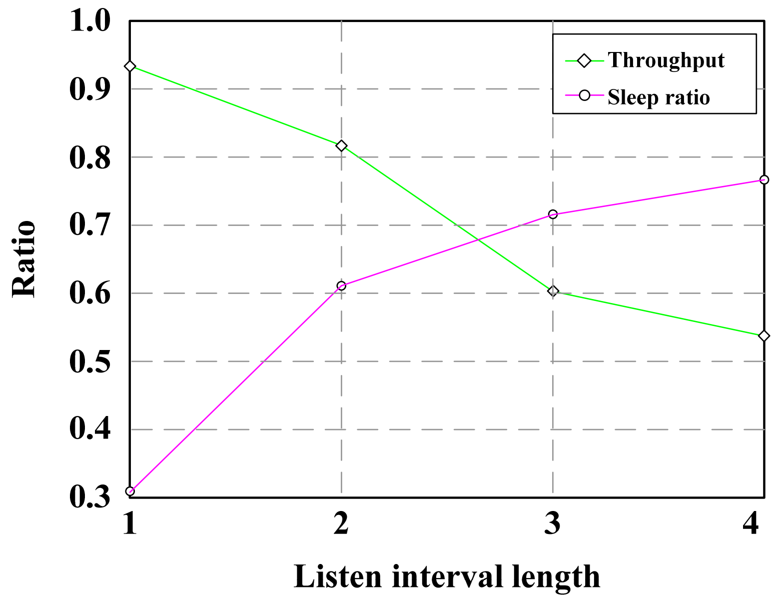



**Figure 4-10 the throughput and sleep ratio as functions of the listen-interval in SCPSb (60 station; traffic generation rate: 8 packets/s)**

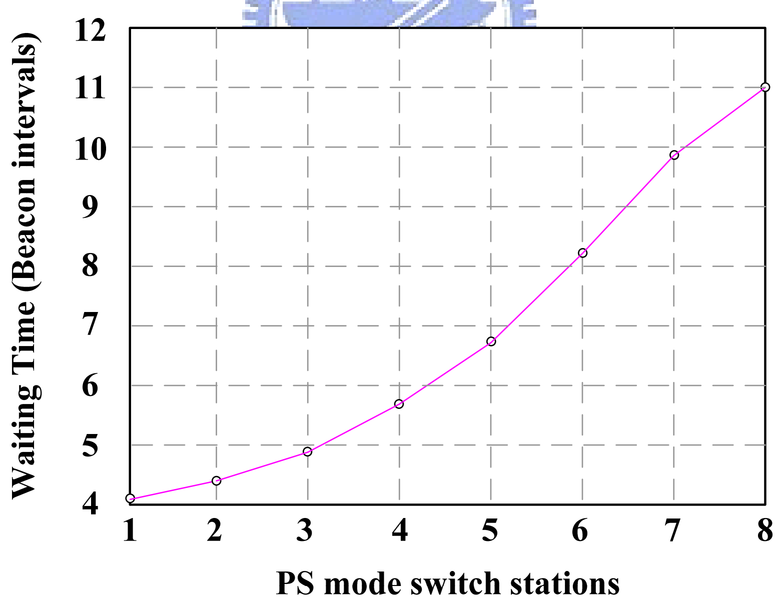



**Figure 4-11 the average waiting time for a station to switch to the PS mode in SCPSb**

Because a station switching to the PS mode needs to serve as the beacon sender for a fixed number of listen-intervals and because a NORMAL beacon frame must be detected before another station may switch to the PS mode, at most 8 stations may switch to the PS mode every 40 beacon intervals. From figure 4-11, we can see that the waiting time increases as the number of switching stations grows. In the worst case, stations have to wait three times the maximum listen-interval.

## 4.4 Summary

We proposed a flexible power-saving scheduling protocol, SCPS, for ad hoc networks. By providing each station a flexible mechanism for adjusting its listen-interval, SCPS can balance the number of wakeup stations during each beacon interval without the need of a coordinator. Simulation results show that, when the number of wakeup stations is balanced, packets dropped due to collision are effectively reduced.

# Chapter 5

# Power Saving Protocols for IEEE 802.16e WMANs

In this chapter, we consider the UGS traffic which has QoS delay constraint and adapt the second power-saving class defined in IEEE 802.16e standard. We propose two kinds of power saving schedule approaches for multiple MSSs, the periodical autonomic sleeping cycle approach (PASC) and the periodical uniform sleeping cycle approach (PUSC). In the PASC, each MSS uses its own sleeping cycle for power saving. In addition to the sleeping time of MSSs, the traffic load in each OFDM frame is considered. The BS will try to schedule the wakeup time of newly joined MSSs to a light load OFDM frames without preempting those MSSs which have been scheduled.

Although PASC lets BS schedule a newly joined MSS without preempting, it lowers the scheduling flexibility and reduces the amount of serviced MSSs. Thus, for serving more MSSs, we proposed the PUSC approach. The PUSC assumes that the possible minimum delay constraint of MSSs is known beforehand. By the way the minimum length of the sleeping cycle for all MSSs can be predicted. The allocated OFDM frames of different MSSs are well controlled to make them overlap-free and we condense the bandwidth of each available OFDM frame to increase the bandwidth utilization.

The organization of this chapter is as follows. Our proposed periodical autonomic sleeping cycle approach is given in section 5.1 and the periodical uniform sleeping cycle approaches are in section 5.2. Section 5.3 gives the simulation results. The summary is in section 5.4.

# 5.1 Autonomic sleep cycle

The periodical autonomic sleeping cycle power-saving scheduling approach considers the second power saving class of IEEE 802.16e. To minimize the power consumption of a MSS with multiple real-time connections, a MSS is required to repeat the sleeping and listening states in a round-robin fashion. Suppose that there are n MSSs in a network and each of them has at least one downlink connection. Each connection provides the Unsolicited Grant Service (UGS), which generates fixed-size data packets on a periodic basis. Besides, each connection of MSSs has its specified QoS delay requirement in which the packets need to be delivered before their expiry times.

For providing QoS guarantee to the connections of a MSS, both delay and bandwidth constraints specified by all connections must be satisfied. To satisfy the delay constraint, the sleeping MSS must be woken up before the minimum delay of a packet expires. We define the sleeping cycle, $\boldsymbol{C_i}$, (in units of OFDM frames) as the duration that a MSS $\boldsymbol{S_i}$ alternates in one round of the sleeping and listening states. If we assume QoS delay constraint, $\varphi$, associated with each connection, since the length of the sleeping cycle must not exceed delay constraints of all connections, the sleeping cycle, $\boldsymbol{C_i}$, for each MSS $\boldsymbol{S_i}$ can be determined by

$$C_i = \min\{\boldsymbol{\varphi}_1^i, \boldsymbol{\varphi}_2^i, \boldsymbol{\varphi}_3^i, ..., \boldsymbol{\varphi}_m^i\} \tag{5.1}$$

Where $m$ is the number of connections in $S_i$, and $\varphi_j^i$, $j = 1,\ldots, m$, is the QoS delay constraint of each connection in $S_i$ (in units of OFDM frames).

To satisfy the bandwidth constraint, the second step is allocating bandwidth to the listening state of each MSS $S_i$, denoted as $\boldsymbol{\omega}_i$. The allocated bandwidth must be large enough to transfer the packets generated within the sleeping cycle of MSS $S_i$, $\boldsymbol{C}_i$.

Equation (5.2) calculates the minimum required bandwidth of each MSS $S_i$ during its sleeping cycle.

$$\boldsymbol{\omega}_i = \frac{\boldsymbol{\tau}_i \times C_i}{\Omega} \tag{5.2}$$

Where $\tau_i$ is the total traffic generation rate of a MSS $S_i$ and $\Omega$ is the capacity for downlink traffic in an OFDM frame. $\boldsymbol{\omega}_i$ is the ratio of OFDM frames.

Finally, we schedule the multiple MSSs. Because the length of the sleeping cycle, $C_i$, of each MSS $\boldsymbol{S}_i$ is different, scheduling their sleeping cycles directly may make their listening time overlap severely with each other. The result of overlapping will cause not only clustering bandwidth allocation but also reducing number of MSSs supported by a BS (bandwidth utilization). Therefore, how to interleave the listening periods between MSSs to reduce the overlapping is the key factor to increasing the number of supported MSSs and improving the bandwidth utilization.

Now, we use the example in figure 5-1 to exhibit the overlapping listening period between MSSs. There are 4 MSSs A, B, C, and D. We simply let their scheduled order be A to D. Their $C_i$ are 2, 3, 6, and 6 and their required bandwidth, $\boldsymbol{\omega}_i$, are 0.2, 0.2, 1.1, and 1.4 OFDM frame capacity. For representing the bandwidth of each OFDM frame, we use N(t) as the accumulated bandwidth at the $t^{th}$ OFDM frame.

| OFDM frame | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 |
| B | 0 | 0 | 0.2 | 0 | 0 | 0.2 | 0 | 0 | 0.2 | 0 | 0 | 0.2 |
| C | 0 | 0 | 0 | 0 | 1 | 0.1 | 0 | 0 | 0 | 0 | 1 | 0.1 |
| D | 0 | 0 | 0 | 0 | 1 | 0.4 | 0 | 0 | 0 | 0 | 1 | 0.4 |
| N(t) | 0 | 0.2 | 0.2 | 0.2 | 2 | 0.9 | 0 | 0.2 | 0.2 | 0.2 | 2 | 0.9 |

**(a)**

| OFDM frame | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 |
| B | 0 | 0 | 0.2 | 0 | 0 | 0.2 | 0 | 0 | 0.2 | 0 | 0 | 0.2 |
| C | 0 | 0 | 0 | 0 | 1 | 0.1 | 0 | 0 | 0 | 0 | 1 | 0.1 |
| D | 1 | 0.4 | 0 | 0 | 0 | 0 | 1 | 0.4 | 0 | 0 | 0 | 0 |
| N(t) | 1 | 0.6 | 0.2 | 0.2 | 1 | 0.5 | 1 | 0.6 | 0.2 | 0.2 | 1 | 0.5 |

**(b)**

| OFDM frame | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 | 0 | 0.2 |
| B | 0 | 0 | 0.2 | 0 | 0 | 0.2 | 0 | 0 | 0.2 | 0 | 0 | 0.2 |
| C | 0 | 0 | 0 | 0 | 1 | 0.1 | 0 | 0 | 0 | 0 | 1 | 0.1 |
| D | 1 | 0.4 | 0 | 0 | 0 | 0 | 1 | 0.4 | 0 | 0 | 0 | 0 |
| J | 0 | 0 | 0.5 | 0 | 0 | 0.5 | 0 | 0 | 0.5 | 0 | 0 | 0.5 |
| N(t) | 1 | 0.6 | 0.7 | 0.2 | 1 | 1 | 1 | 0.6 | 0.7 | 0.2 | 1 | 1 |

**(c)**

**Figure 5-1 example of periodical autonomic sleeping cycle approach**

In figure 5-1(a), the BS schedules the MSSs directly according to their sleeping cycle. The N(t) shows that the capacity of the 5th and the 11th OFDM frames overflow if all of the four MSSs are scheduled. BS cannot support all of the MSSs, if MSSs are scheduled in this way. With further investigation, we find that while MSS D is scheduled, the capacities of the 5th and the 11th OFDM frames have been exhausted. Thus, only MSS A, B, and C can be scheduled.

However, if BS can dispatch the bandwidth which is required by the listening state of the new scheduled MSS to the light load OFDM frames, more MSSs will be serviced. For example, as the Figure 5-1 (b) shows, the maximum allocated bandwidth in an OFDM frame after MSS A, B, and C being scheduled is the same as the result of Figure 5-1 (a). When MSS D is scheduled, BS allocates the listening OFDM frames of MSS D to $1^{st}$, $2^{nd}$ and $7^{th}$ and $8^{th}$ OFDM frames instead of $5^{th}$, $6^{th}$ and $11^{th}$, $12^{th}$. The maximum allocated bandwidth ratio in an OFDM frame is still 1 instead of 2. In this schedule, BS can completely schedule all of these four MSSs without incurring overflow at any OFDM frame.

Our proposed PASC approach bases on the observation of Figure 5-1 (b). The PASC approach greedily allocates the bandwidth of new scheduled MSS to low load OFDM frames. It tries to minimize the maximal allocated bandwidth in any OFDM frame while a new MSS is scheduled. Therefore, the listening OFDM frames and required bandwidth of each MSS need to be traced for computing the traffic load in each OFDM frame.

To trace the listening OFDM frames of each MSS, BS maintains an active counter, denoted as $c_i(t)$, for each MSS $S_i$. This $c_i(t)$ indicates remaining OFDM frames that the listening OFDM frame of a MSS $S_i$ will arrive at.

Initially, BS sets $c_i(t) = C_i - 1$ for MSS $S_i$ wherever MSS $S_i$ is scheduled. The counter $c_i(t)$ will be decreased by one every OFDM frame. The MSS $S_i$ wakes up if its $c_i(t)$ becomes less than $\lceil \boldsymbol{\omega_i} \rceil$. After $c_i(t)$ becomes zero, the counter will reset to $c_i(t) = C_i - 1$ for further counting down. Thus, after one OFDM frame passes, BS sets counter $c_i(t+1)$ for the $(t+1)^{th}$ OFDM frame as follows:

$$c_i(t+1) = \begin{cases} c_i(t)-1, \; if \; c_i(t) \neq 0 \\ C_i - 1, \quad if \; c_i(t) = 0 \end{cases}$$

When BS has active counter $c_i(t)$ and sleeping cycle $C_i$ at the $t^{th}$ OFDM frame of MSS $S_i$, BS can find the contributed traffic load of MSS $S_i$ for the OFDM frame t + k, k= 1, 2, . . ., denote as $w_i(t + k)$.

$$w_i(t+k) = \begin{cases} 1 & c_i(t) < (k \bmod C_i) \le c_i(t) + \lceil \boldsymbol{\omega}_i \rceil \\ \boldsymbol{\omega}_i - \lfloor \boldsymbol{\omega}_i \rfloor & k \bmod C_i = c_i(t) \\ 0 & otherwise \end{cases}$$

$\boldsymbol{\omega_i}$ is the required bandwidth of MSS $S_i$ obtained from equation 2.2. Then, N(t+k) can be found by computing N(t+k) = $\sum_{i \in S} w_i(t+k)$ for k = 1, 2, . . . where S is a set including all scheduled MSSs.

Hence, the power-saving scheduling problem (PSP) considered in PASC can be stated formally as follows: Given a set of power saving MSSs S (IEEE 802.16e Type II PS mode) at the OFDM frame t, each of the MSS $S_i$ has it sleeping cycle $C_i$. For a new MSS j assigns an initial value to its $c_j(t)$ such that the maximum value of N(t + k) = $\sum_{i \in S \cup \{j\}} w_i(t+k)$, k = 1, 2, . . . is minimized.

By observing figure 5-1 (b), we find that a pattern repeats every six OFDM frames, eg. the (N(1), N(2), … N(6)) = (N(7), N(8), … N(12)). The length of this repeating pattern, *r*, can be found by computing the least common multiple (*lcm*) of sleeping period $C_i$ For example, the r in the example of figure 5-1 is lcm{2, 3, 6, 6} = 6.

Thus, $N^*$ = max{N(t + 1), N (t + 2), . . . , N(t + r)} = max{N(t + k)|k = 1, 2, . . .}. Now, we want to schedule a new MSS *j* with sleeping period $C_j$ to the set S with repeating pattern size $r$ and assign an initial value to $c_j(t)$. A stepwise solving method for PSP problem is given as follows.

1. Find r = lcm{$C_j$, $r$} and a sequence of total number of wakeup stations (N(t + 1), N(t + 2), . . . , N(t + $r$)) for the first $r$ intervals for sleeping station set S ∪ {j}.

2. For i = $C_j$ − 1, . . . , 1, 0, perform the following operations:

(a) Set $c_j$(t) = i and find ($w_j$(t + 1), $w_j$(t + 2), . . ., $w_j$(t + r));

(b) Set (N(t+1), N(t+2), . . . , N(t+$r$)) = (N(t+1), N(t+2), . . . , N(t+r))+($w_j$(t + 1), $w_j$(t + 2), . . ., $w_j$(t + $r$));

(c) Find $N_i$ = max{ N(t + 1), N(t + 2), . . . , N(t + $r$)}.

3. Find $N^*$ = min{$N_i$|i = $C_j$ − 1, $C_j$ − 2, . . . , 0}, say $N^*$ = $N_k$, and thus set $c_j(t)$ = k. If $N_k \le$ 1, MSS j can be scheduled.

For example, in figure 5-1 (c) the 5 MSSs have r = 6. MSS J, whose $C_J$ = 3 and $\boldsymbol{\omega_J}$ = 0.5, is the next scheduled MSS. The BS applies the above solving method to determine the initial value of counter $c_J(t)$ for MSS $J$ as follows.

1. $r$ = lcm{3, 6} = 6 and (N(t + 1), N(t + 2), . . . , N(t + 6)) = (1, 0.6, 0.2, 0.2, 1, 0.5)

2. i = 2:

(a) Set $c_J(t)$ = 2 and find ($w_J$(t+1), $w_J$(t+2), . . ., $w_J$(t+6)) = (0, 0, 0.5, 0, 0, 0.5);

(b) Set (N(t+1), N(t+2), . . . , N(t+6)) = (1, 0.6, 0.2, 0.2, 1, 0.5) + (0, 0, 0.5, 0, 0, 0.5) = (1, 0.6, 0.7, 0.2, 1, 1);

(c) Find $N_2$=max{1, 0.6, 0.7, 0.2, 1, 1} = 1.

i = 1:

(a) Set $c_J(t)$ = 1 and find ($w_J$(t+1), $w_J$(t+2), . . ., $w_J$(t+6)) = (0, 0.5, 0, 0.5, 0);

(b) Set (N(t+1), N(t+2), . . . , N(t+6)) = (1, 0.6, 0.2, 0.2, 1, 0.5) + (0, 0.5, 0, 0.5, 0) =(1, 1.1, 0.2, 0.2, 1.5, 0.5);

(c) Find $N_1$=max{1, 1.1, 0.2, 0.2, 1.5, 0.5} = 1.5. (Overflow)

i = 0:

(a) Set $c_J(t)$ = 0 and find ($w_J$(t+1), $w_J$(t+2), . . ., $w_J$(t+6)) = (0.5, 0, 0, 0.5, 0, 0);

(b) Set (N(t+1), N(t+2), . . . , N(t+6)) = (1, 0.6, 0.2, 0.2, 1, 0.5) + (0.5, 0, 0, 0.5, 0, 0) = (1.5, 0.6, 0.2, 0.7, 1, 0.5);

(c) Find $N_0$ = max{1.5, 0.6, 0.2, 0.7, 1, 0.5} = 1.5.

3. Find $N^*$ = min{1, 1.5, 1.5} = 1.0, i.e., $N^*$ = $N_2$, and thus set $c_J(t)$ =2.

The time complexity of this approach is dominated by both the length of this repeating pattern $r$ and the sleeping cycle $C_J$ of new added MSS $J$. While a MSS is scheduled, BS has to find the maximum loading OFDM frame during the period $r$ for each $c_J(t)$ of MSS $J$. The amount of possible $c_J(t)$ is equal to the number of OFDM frames in the sleeping cycle $C_J$. Thus, the complexity to find out all $N_i$ is O($r$×$C_J$). After then, BS has to compute the minimum $N^*$ from all $N_i$ which is O($C_J$). So, total time is O($r$×$C_J$) + O($C_J$).

The advantage of the PASC approach is that it can simply schedule a new joined MSS without preempting the current schedule of those MSSs that have been scheduled. Newly joined MSSs only have to follow their own sleeping cycles and be allocated to the unused bandwidth space. However, the advantage of freely selecting the sleeping cycle also greatly increases the difficult to free the listening time of multiple MSSs from overlapping with each other. The penalty is reducing the number of serviced MSSs.

Thus, to improve the number of serviced MSSs, we need more limitations on the wakeup schedules of MSSs. If the QoS requirement of each MSS can be known beforehand, we can use the periodical uniform sleeping cycle approaches proposed in the next section.

## 5.2 Uniform Sleeping Cycle

In the PASC, using different length of sleeping cycle is the prim factor that causes the listening period of MSSs overlapping with each other. If all of the MSSs use a uniform length of sleeping cycle, BS can schedule the new MSS easily by accumulating its needed bandwidth to the scheduled MSS. If the bandwidth during the uniform sleeping cycle doest not exhaust, this MSS can be scheduled. We denote this uniform sleeping cycle as $C_{com}$ (in units of OFDM frames).

Because the sleeping cycle is determined by the QoS delay constraint of MSS, different MSSs may use different length of $C_{com}$. To determine this uniform $C_{com}$, the QoS delay constraint of all MSSs must not be violated. Therefore, the minimum sleeping cycle over all serviced MSSs must be used, shown as the equation (5.3). So, The MSSs with longer sleeping cycles can shrink to a smaller sleeping cycle to match the $C_{com}$, Their QoS delay constraint can still be satisfied.

$$\boldsymbol{C_{com} = \min\{ C_1, C_2, C_3, \ldots, C_n\}} \tag{5.3}$$

In the following, we give three schedule approaches for PUSC.

## 5.2.1 Multiple MSSs Power-Saving Scheduler (MMPS)

The first approach in the PUSC is called Multiple MSSs Power-Saving Scheduler (MMPS). First of all, we use equation (5.3) to finds the uniform sleeping cycle, $C_{com}$, for all MSSs. After then, while a new MSS included, BS has to determine the, $\boldsymbol{\omega_i}$, the allocated bandwidth during $C_{com}$, for this MSS. In the MMPS approach, BS simply schedules the needed bandwidth of new MSS to a completely free OFDM frame within $C_{com}$. By using a uniform length sleeping cycle, the equation (5.2) ,which calculates the bandwidth of each MSS $S_i$ is changed as following.

$$\omega_i = \left\lceil \frac{\tau_i \times C_{com}}{\Omega} \right\rceil \tag{5.4}$$

Once the needed bandwidth of the new MSS is obtained, BS will determine whether this new MSS can be scheduled or not by the following equation.

$$\begin{cases} 1 & W + \omega_i \leq C_{com} \\ 0 & otherwise \end{cases} \tag{5.5}$$

Where $\boldsymbol{W}$ is the number of OFDM frames during $C_{com}$ that have been allocated to scheduled MSSs before this new MSS is included. If the result of (5.5) is not zero, BS will service this new MSS. Thus, the maximum amount of admitted MSSs, denoted as $\delta$, which can be served by the BS, is show in equation (5.6).

$$\delta = \arg \max_j \{\sum_{i=0}^{j} \omega_i < C_{com}\} \tag{5.6}$$

Figure 5-2 shows an MMPS example in which 4 MSSs in the example of figure 5-1 are served. Their $C_{com}$ is 2. First, the BS computes the number of OFDM frames needed for its serviced MSSs according to equation 2.4. The required bandwidth of these four MSSs within the $C_{com}$ is 0.2, 0.4/3, 1.1/3, and 1.4/3 OFDM frame, respectively. All of these MSSs need one OFDM frame to transfer their packets. The BS schedules MSS A in the 1$^{st}$ OFDM frame, MSSs B to the 2$^{nd}$. Because all OFDM frames within the $C_{com}$ have been exhausted, the MSS C and MSS D can not be scheduled.

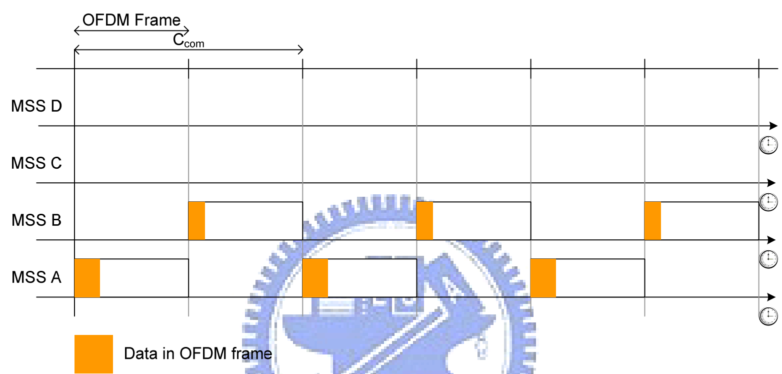



**Figure 5-2 an example of MMPS scheduling**

From this example, we can observe that the allocated bandwidth for MSS A, and B is not fully utilized. Only partial bandwidth of the allocated OFDM frame is used. Too much bandwidth is allocated to MSS A and B. We name this kind of OFDM frame a *fragment* OFDM frame. For preventing allocating too much bandwidth and improving the bandwidth utilization, we have the following two approaches.

## 5.2.2 MMPS with Fragment Collection (MMPS-FC)

The MMPS is simple for implementation but its bandwidth utilization is too bad, especially when there are some MSSs with very low data rate. The admission region of MMPS can be very small, which significantly reduces the bandwidth utilization.

We start major idea of our proposed multiple-MSS power-saving scheduler with fragment collection (MMPS-FC) via the example in the figure 5-3. If the packets of MSS B and MSS C in the MMPS are scheduled in the fragment OFDM frame left by MSS A, the MSS D is scheduled in the second OFDM frame of the $C_{com}$, all of these four MSSs can be scheduled.

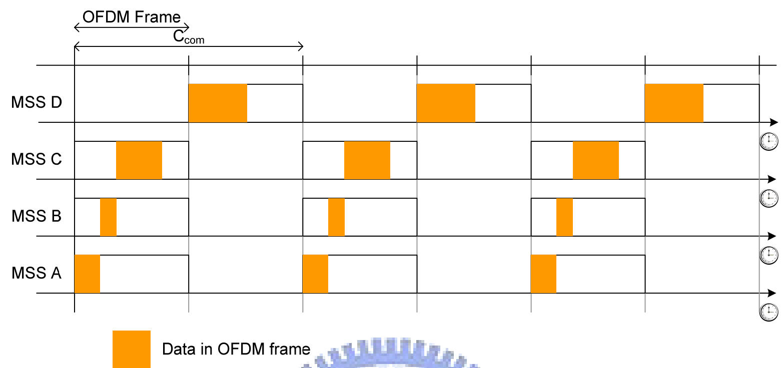



**Figure 5-3 example of MMPS with fragment collection**

Thus, in the MMPS-FC approach, when BS prepares to include a new MSS, BS has to compute not only the needed bandwidth of this new MSS by using equation (5.2) but also the fragment OFDM frame that may be created by the packets of this new MSS. If the remaining bandwidth of the fragment OFDM frame left by the previously scheduled MSS is large enough to fit its fragment data, the packets are scheduled into the fragment OFDM frame; otherwise, a new OFDM frame is used. Therefore, the unused bandwidth in the fragment OFDM frames can be minimized further. Figure 5-4 is the algorithm to allocate the bandwidth for the new MSS in the MMPS-FC approach.

$\boldsymbol{O_k}$: The $k^{th}$ OFDM frame which is the last allocated OFDM frame during $\boldsymbol{C_{com}}$
$\boldsymbol{\Omega_{frag}}$: The remaining bandwidth of $\boldsymbol{O_k}$.
$\boldsymbol{\omega_{frag}}$: The bandwidth of fragment OFDM frame created by the new MSS $i$.

$\omega_{frag} = \omega_i - \lfloor \omega_i \rfloor$
*if* $\omega_{frag} < \Omega_{frag}$
    $O_k \leftarrow \omega_{frag}$
    $O_{(k+1)..(k+\lfloor \omega i \rfloor)} \leftarrow \lfloor \omega_i \rfloor$

**Figure 5-4 algorithm of MMPS-FC approach**

## 5.2.3 MMPS with Boundary Free (MMPS-BF)

Although the MMPS-FC improves the bandwidth utilization of MMPS, it still wastes some bandwidth if the fragment OFDM frame can not be completely fulfilled by the next MSS. For example, in the figure 5-3, there is still some unused bandwidth in the first OFDM frame of the $C_{com}$. Thus, our third approach, called MMPS with Boundary Free (MMPS-BF), does not limit bandwidth sharing by the boundary of OFDM frame. The scheduling can allocate bandwidth sharing across the boundary of two OFDM frames. Thus, except the last scheduled OFDM frame, no fragments are left.

This MMPS-BF increases the bandwidth utilization compared with the other two approaches. However, a MSS may pay more wakeup time if its data is smaller than the capacity of one OFDM frame but is allocated to cross over the boundary of two OFDM frames. Therefore, the MSS needs to wake up twice, instead of waking up once. We give an example of MMPS-BF in figure 5-5.

As figure 5-5 shows, after MSS C is assigned, the remaining bandwidth left by MSS C can not completely satisfy the needed of MSS D. Thus, MSS D is assigned to the $1^{st}$

and 2nd OFDM frames within a sleep cycle $C_{com}$. Although the required bandwidth of MSS D is not larger than an OFDM frame, it has to wakeup for two OFDM frames.

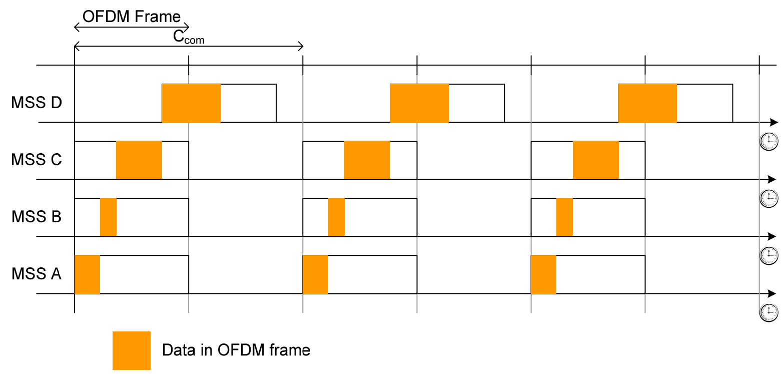



**Figure 5-5 a MMPS example with boundary free**

Because the minimum delay constraint of MSSs can be known beforehand, the time complexities of these periodical uniform sleeping cycle approaches are dominated by the length of sleeping cycle $C_{com}$. While a MSS $S_i$ is scheduled, BS has to check whether the remaining bandwidth during sleeping cycle $C_{com}$ can satisfy the bandwidth requirement of $S_i$. Thus, the time complexity of PURC approaches is O($C_{com}$).

Among all of the proposed PUSC approaches, the BS obtains the needed bandwidth of a MSS when it is included for scheduling. If the total number of MSSs and their needed bandwidth are known beforehand, the MMPS-BF can choose suitable serviced MSSs to improve its bandwidth utilization.

### 5.2.4 Maximize the bandwidth utilization of MMPS-BF

To maximum the bandwidth utilization of MMPS-BF, we transform the MMPS-BF schedule problem to the 0-1 knap-sack problem. Assume that total number of MSSs is n. Each of MSSs is treated as an item in the 0-1 knap-sack problem. The needed bandwidth of each MSS is its weight, $\boldsymbol{\omega_i}$. The total bandwidth during the $C_{com}$ is the capacity of the knapsack. The value function of each item, denote as $v_i$, is defined as:

$$v_i = \boldsymbol{\omega_i} \tag{5.7}$$

So, we can obtain the solution with optimal bandwidth utilization. However, if we also consider to higher the number of serviced MSS under the solution of optimal bandwidth utilization, the value function of each item will change as follows:

$$v_i = \boldsymbol{\omega_i} + f \tag{5.8}$$

Where $f$ is a small constant bias added to the value of each item. When the value function in (5.7) is used and there are more than two sets of items whose total value is the same but amount of items is different, the constant bias $f$ can help BS to select the set which contains maximum items.

Finally, the PUSC schedule problem is formulated to 0-1-knapsack problem as following:

$$\text{Max} \ \sum_{i=1}^{n} v_i x_i \quad x_i = 0 \ or \ 1 \qquad\qquad \text{Subject to} \ \sum_{i=1}^{n} \boldsymbol{\omega}_i x_i \le C_{com} \qquad x_i = 0 \ or \ 1$$

However, the 0-1 knapsack problem is NP-complete or "hard" combinatorial optimization for which, for which, to date, no technically sound algorithms are available [40]. We can not use greedy algorithm such as the PUSC or PASC to find an optimal solution. Thus, we need a heuristic algorithm to find out a better solution.

Fortunately, the 0-1 knapsack problem has the optimal sub-problem. While we consider an item to include, we must compare the solution to the sub-problem in which the item is included before we can make the choice. We can apply the dynamic programming algorithm to find out an optimal solution. The schedule results which obtain from dynamic programming will also be given in our simulation.

# 5.3 Simulation Results

In this section, we give the simulation results to evaluate our scheduling approaches, the PASC and the PUSC approaches. We also implement the KNAP approach for static group of MSSs. Besides, the case that MSS wakes up and notify to retrieve its data when its traffic arrives is also evaluated for comparison. We denote it as NoAgre.

The evaluated metrics are as follows: the average number of scheduled MSSs, the average sleeping ratio, and the bandwidth utilization. The average number of MSSs provides the number of MSSs that can be supported by a BS with our proposed scheduling approach. Average sleeping ratio shows our approaches' power-saving capability. The bandwidth utilization gives the index of how efficient use of the bandwidth of our approaches.

## 5.3.1 Environment setup

In our simulation, at first, we adapt the type II power saving class defined in 802.16e. We consider downlink scheduling for the UGS connections. Each MSS has **five** type II power saving class connections. We use the parameters recommended by ITU Telecommunication Standardization Sector (ITU-T) for voice traffic. The traffic inter-arrival rate for the VoIP connections is randomly selected from 20 or 40ms. The QoS delay constraint is between 150ms to 350ms. In our simulation, the QoS delay constraint of each connection chooses randomly from the minimum one, which is 150ms, to the maximum one, which is the x-axis label in the figures. Each generated packet size is 100 Bytes.

The maximal data rate that BS offers for all MSSs is 16 Mbps. The length of an OFDM frame is 5ms. The total simulation duration is 300 seconds and the simulation results are averaged from 100 tries.

## 5.3.2 Numerical results

Figure 5-6 shows the bandwidth utilization. Without considering the gap between OFDM frames, the MMPS-BF has the best result among all approaches, respectively. Its bandwidth utility is about 96%. By reusing the fragment OFDM frame of previous MSS for the new scheduled MSS, the MMSP-FC approach has 79% bandwidth utilization. For the MMPS, allocating too much bandwidth for a MSS lowers the bandwidth utilization. It only has 27% bandwidth utilization.

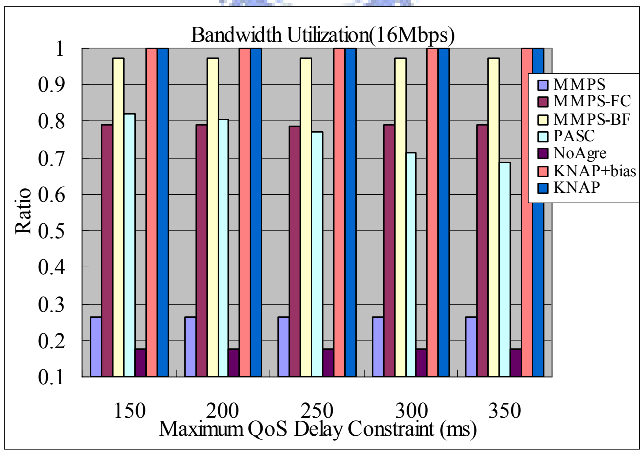



**Figure 5-6 bandwidth utilization**

For the approach of PASC, the bandwidth utilization is slightly better than the MMPS-FC approach when the maximum QoS delay constraint is less than 200ms (about

81%). After the maximum QoS delay constraint grows more than 250 ms, its bandwidth utilization decays. Especially in the case of 350ms maximum QoS delay constraint; its bandwidth utilization is only 70% which decays 10% than the one in 150ms maximum QoS delay constraint.

The reason comes from the difference of QoS delay constraint between MSSs. More difference sizes of QoS delay constraint introduces higher chance for MSSs to overlap their listening periods with others. Because the bandwidth of each OFDM frame is limited, too much overlapping may cause the bandwidth allocation overflow and the payment is reducing the number of scheduled MSSs.

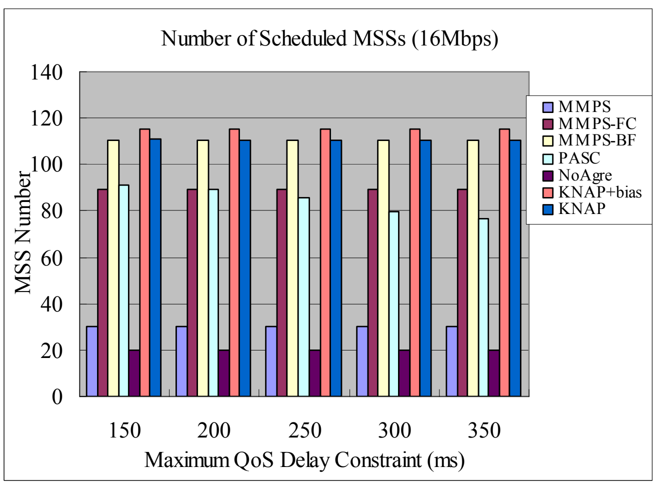



**Figure 5-7 the number of serviced MSSs**

If BS does not have a well plan to schedule the wakeup time of MSSs and lets them wake up to receive data while data arrives, as shown in the curve of NoAgre, overlapping of the listening period will be very sever. The 18% bandwidth utilization proves its defect.

For the sack of comparison, we add the KNAP schedule result in this figure. The bandwidth utilization is almost 100%. Thus, by scheduling the static bandwidth requirement of MSSs we can almost achieve 100% bandwidth utilization.

Figure 5-7 shows the number of serviced MSSs. Basically, the effect of number of serviced MSSs has reflected on the bandwidth utilization. More serviced MSSs will contribute higher bandwidth utilization. An interesting result we want to show in this figure is the KNAP, the KNAP-bias, and MMPS-BF.

The number of service MSSs of KNAP and the MMPS-BF are similar. However, KNAP has higher bandwidth utilization. The result proves that applying dynamic programming to find a schedule from a group of static MSSs can help us find the optimal solution on bandwidth utilization. The service number of MSSs of the KNAP and the KNAP-bias proves that by adding the bias value on the items of KANP, we can find a solution that has the same bandwidth utilization but more serviced MSSs.

Similar to the result of bandwidth utilization, without a well plan to schedule the wakeup time of each MSS, the NoAgre can only schedule about 20 MSSs in our simulation.

Figure 5-8 shows the sleeping ratio of each approach. It is obviously that the sleeping ratio of proposed PASC approach increases as the maximum delay constraint becomes large, respectively. By autonomously selecting the sleeping cycle, each MSS can sleep and wakeup following its own wakeup schedule without limited by the common sleeping cycle used in the PUSC approaches.

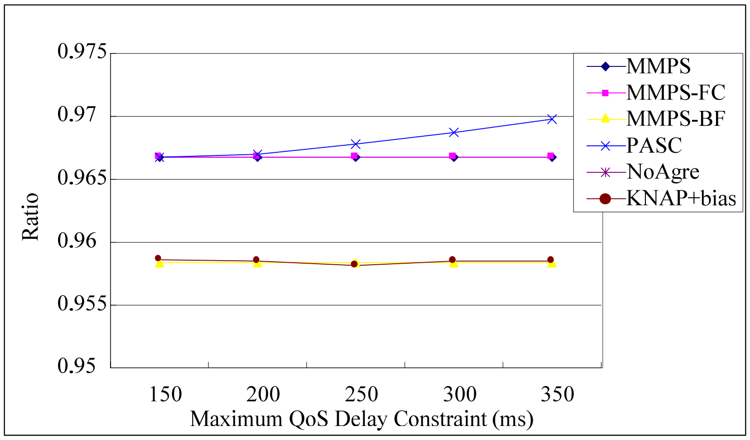


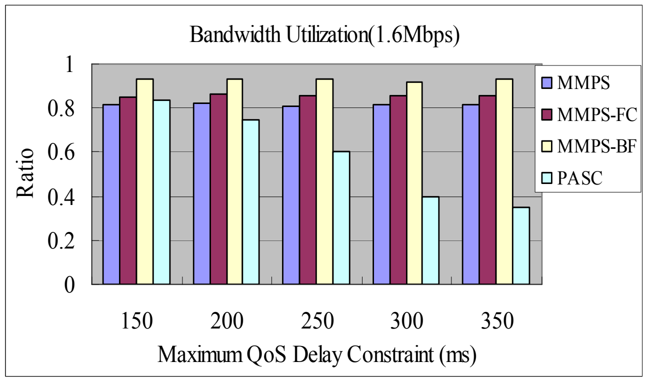

**Figure 5-8 comparison of sleeping ratios (16Mbps)**

For the PUSC approaches, MMPS, MMPS-FC, and MMPS-BF, all MSSs have to use the same size of sleeping cycle. This uniform cycle is determined by the minimum QoS delay constraint. It does not reflect to the change of the maximum QoS delay constraint. Thus, the sleeping is almost the same in different maximum QoS delay constraint.

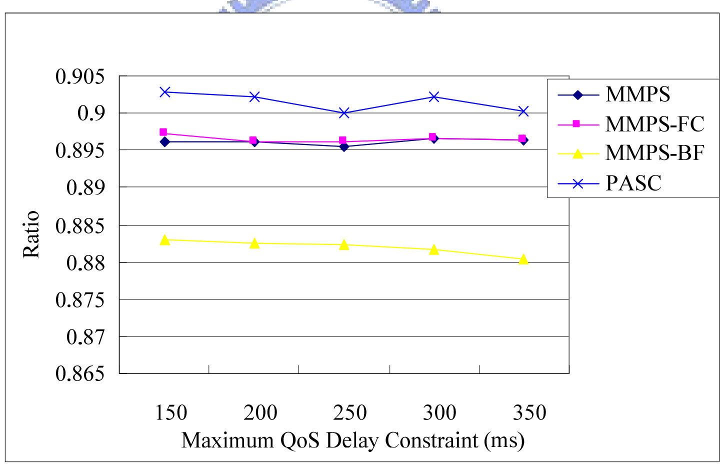
The sleeping ratio of KNAP-bias and MMPS-BF is almost the same. It is because they do not consider the gap of OFDM frame. For the NoAgre which does not have a well planning schedule, it sleeping ratio is only 31%.

For examining impacts of the bandwidth capacity of our proposed approaches, we tighten the available bandwidth capacity on each OFDM frame to 1.6Mbps, the 10% of original 16Mbps, and show the sleeping ratio and the bandwidth utilization again, shown in figure 5-9.

The bandwidth utilization of figure 5-9 is similar to the one of 16Mbps. However, the sleeping ratio of the PASC becomes the best one among our proposed approaches. Because the capacity of an OFDM can not completely consumes the traffic generated by each MSS within the common cycle of the PUSC approaches, the average sleeping time of each PUSC approach reduces. Consequently, the average sleeping time of the PASC approach is also decrease, but the ratio is smaller than the PUSC. The reason is that MSSs in PASC have longer sleeping cycle instead of the minimum common sleeping cycle.




(a)




(b)

**Figure 5-9 the bandwidth utilization and sleeping ratio under the case of 1.6Mbps rate**

No matter the PUSC or PASC, BS needs to stall starting service time of the new scheduled MSS. For his stalling period, BS needs to buffer the data of the new MSS. Table 5-1 shows the average required buffer size. The average needed buffer size for each MSS is less than 1.3Kbytes. Thus, if there are 90 MSSs, the total necessary buffer size is no more than 120Kbytes.

**Table 5-1 the average required buffered size for each MSS**

| **Maximum delay constraint** | **150** | **200** | **250** | **300** | **350** |
|---|---|---|---|---|---|
| **Buffer size(Kbytes)** | 1.2 | 1.2 | 1.2 | 1.0 | 1.0 |

# 5.4 Summary

This chapter proposes two kinds of power saving scheduling approaches for multiple MSSs in IEEE 802.16e wireless network, the PUSC and the PASC. In the PASC approach, each MSS uses its own sleeping cycle for power saving schedule. It benefit is that BS can easily schedule the new MSS without preempting those been scheduled MSS. In the PUSC approaches, we add extra presumption which the QoS requirements of all possible connections can be known beforehand to improve the utility of scheduling. All MSSs use a uniform length of sleeping cycle for their wakeup schedule. BS simply schedule MSSs' wakeup time in the periodical uniform sleeping cycle. Besides, for the case to find the optimal bandwidth utilization for a group of static MSSs, we transform the schedule problem to 0-1 knapsack problem and use dynamic programming solution.

# Chapter 6

# Conclusions and Future Works

In this dissertation, we give the power saving mechanisms for each 802.11 infrastructure mode, ad hoc mode, and the 802,16e. Each of the proposed mechanism gives well results for stations to schedule the traffic of stations or MSSs. In 802.11 infrastructure mode, the LAWS scheme reduces the possible contention by controlling the indicated wakeup station to access the medium. In the 802.11 ad hoc mode, the SCPS scheme provides a distributed and self-configuration mechanism to schedule the wakeup station. The number of stations in each beacon interval can efficiently be balanced to reduce the collision. Besides, the wakeup period of each station is greatly reduced. Stations do not have to wake up at each beacon interval. In the 802.16e, we find the minimum necessary number of wakeup OFDM frames for single station. We try to find a feasible schedule for multiple stations under the constraint of minimum necessary number of wakeup OFDM frames of each station.

There are still many uncovered problem in the power saving schedule of 802.16e. For example, how to find an available schedule if the allocated slots in an OFDM frame of each station are different, and how to find a tightening bound of feasible schedule. All of the above issues will be deeply investigate in the future.

# Bibliography


[1] IEEE Std 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", Nov 1999.

[2] IEEE Std 802.16, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", Nov 2004.

[3] IEEE Std 802.11f, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", Nov 2003.

[4] IEEE Std 802.11a, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", Nov 1999.

[5] IEEE Std 802.11b, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", Nov 1999.

[6] IEEE Std 802.11g, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", Nov 2003.

[7] IEEE Std 802.11i, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications", 2003.

[8] IEEE Draft Std 802.16e/D7, "Draft IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems", IEEE P802.16e/D7, Apr 2005.

[9] IEEE Draft Std 802.16i/D5, "Draft IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems", IEEE P802.16e/D7, Oct 2007.

[10] IEEE Draft Std 802.16k/D3, "Draft IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems", IEEE P802.16e/D7, Oct 2006.

[11] IEEE Draft Std 802.16j/D1, "Draft IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems", IEEE P802.16e/D7, Aug 2006.

[12] G.Razzano, L.Andreani, R.Cusani, "An adaptive algorithm to reduce power consumption", *Proceedings of IEEE International Conference on Communications (ICC '03)*, vol.2, pp.1096-1100, May 2003.

[13] S.T.Sheu, J.Chen, "MR$^2$RP: the multi-rate and multi-range routing protocol for ad hoc wireless networks", *Proceedings of 7$^{th}$ International Symposium on Computers and Communications*, pp.435-440, July 2002.

[14] C.S.Hsu, J.P.Sheu, Y.C.Tseng, "Minimize waiting time and conserve energy by scheduling transmissions in IEEE 802.11-based ad hoc networks", *Proceedings of 10$^{th}$ International Conference on Telecommunications*, vol.1, pp.393-399, Feb 2003.

[15] J.Jelitto, A.N.Barreto, H.L.Truong, "Power and rate adaptation in IEEE 802.11a wireless LANs", *Proceedings of 57$^{th}$ IEEE Semiannual Vehicular Technology Conference*, vol.1, pp.413-417, April 2003.

[16] C.A.S.Oliveira, P.M.Pardalos, "A distributed optimization algorithm for power control in wireless ad hoc networks", *Proceedings of 18$^{th}$ International Parallel and Distributed Processing Symposium*, pp.177-182, April, 2004.

[17] A.Sheth, R.Han, "Adaptive power control and selective radio activation for low-power infrastructure-mode 802.11 LANs", *Proceedings of 23$^{rd}$ International Conference on Distributed Computing Systems Workshops*, pp.812-818, May 2003.

[18] S.Gitzenis, N.Bambos, "Efficient data pre-fetching for power-controlled wireless packet networks", *Proceedings of 1$^{st}$ Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services*, pp.64-73, Aug 2004.

[19] Z.T.Chou, "A Randomized Power Management Protocol with Dynamic Listen Interval for Wireless Ad Hoc Networks", *Proceedings of IEEE 63$^{rd}$ Vehicular*

*Technology Conference (VTC 2006)*, vol.3, pp.1251-1255, April 2006.

[20] M.Liu, M.T.Liu, "A power-saving scheduling for IEEE 802.11 mobile ad hoc network", *Proceedings of IEEE International Conference on Computer Networks and Mobile Computing (ICCNMC 2003)*, pp.238-245, Oct 2003.

[21] S.L.Wu, P.C.Tseng, "An energy efficient MAC protocol for IEEE 802.11 WLANs", *Proceedings of IEEE 2nd Annual Conference on Communication Networks and Services Research*, pp.137-145, May 2004.

[22] E.S.Jung, N.H.Vaidya, "An energy efficient MAC protocol for wireless LANs", *Proceedings of IEEE 21st Annual Joint Conference of the Computer and Communications Societies (INFOCOM 2002)*, vol.3, pp.1756-1764, June 2002.

[23] W.Ye, J.Heidemann, D.Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor network", *IEEE/ACM Transactions on Networking*, vol.12, issue 3, pp.493-506, June 2004.

[24] C.Suh, Y.B.Ko, "A traffic aware, energy efficient MAC protocol for wireless sensor networks", *Proceedings of IEEE International Symposium on Circuits and Systems2005 (ISCAS 2005)*, vol.3, pp.2975-2978, May 2005.

[25] V.Rajendran, J.J.Garcia-Luna-Aceves, K.Obraczka, "Energy-efficient, application-aware medium access for sensor networks", *Proceedings of IEEE International Conference on Mobile Ad hoc and Sensor Systems*, pp.8-14, Nov 2005.

[26] R.P.Mann, K.Namuduri, R.Pendse, "Energy aware routing protocol (EARP) for ad-hoc wireless sensor networks", *Proceedings of IEEE Global Telecommunications Conference Workshops2004 (GlobeCom Workshops 2004)*, pp.419-427, Nov 2004.

[27] D.Kim, J.J.Garcia-Luna-Aceves, K.Obraczka, J.C.Cano, P.Manzoni, "Routing mechanisms for mobile ad hoc networks based on the energy drain rate", *IEEE Transactions on Mobile Computing*, vol.2, issue 2, pp.161-173, April 2003.

[28] T.Park, K.G.Shin, "Optimal tradeoffs for location-based routing in large-scale ad hoc networks", *IEEE/ACM Transactions on Networking*, vol.13, issue 2,

pp.398-410, April 2005.

[29] J.R.Lee, S.W.Kwon; D.H.Cho, "Adaptive beacon listening protocol for a TCP connection in slow-start phase in WLAN", *IEEE Communications Letters*, vol.9, issue 9, pp.853-855, Sep 2005.

[30] C.M.Chao, J.P.Sheu, I.C.Chou, "An adaptive quorum-based energy conserving protocol for IEEE 802.11 ad hoc networks", *IEEE Transactions on Mobile Computing*, vol.5, issue 5, pp.560-570, May 2006.

[31] K.Han, S.Choi, "Performance analysis of sleep mode operation in IEEE 802.16e mobile broadband wireless access systems", *Proceedings of IEEE 63rd Vehicular Technology Conference (VTC 2006-Spring)*, vol.3, pp.1141-1145, Spring 2006.

[32] J.B.Seo, S.Q.Lee, N.H.Park, H.W.Lee, C.H.Cho, "Performance analysis of sleep mode operation in IEEE 802.16e", *Proceedings of IEEE 60th Vehicular Technology Conference, 2004 (VTC2004-Fall)*, vol.2, pp.1169-1173, Sep 2004.

[33] X.Yang, "Energy saving mechanism in the IEEE 802.16e wireless MAN", *IEEE Communications Letters*, vol.9, issue 7, pp.595-597, July 2005.

[34] Y.Zhang, M.Fujise, "Energy management in the IEEE 802.16e MAC", *IEEE Communications Letters*, vol.10, issue 4, pp.311-313, April 2006.

[35] K.Lei, D.H.K.Tsang, "Performance study of power saving classes of type I and II in IEEE 802.16e", *Proceedings of 31st IEEE Conference on Local Computer Networks*, pp.20-27, Nov 2006.

[36] J.Jang, K.Han, S.Choi, "Adaptive power saving strategies for IEEE 802.16e mobile broadband wireless access", *Proceedings of Asia-Pacific Conference on Communications*, pp.1-5, Aug 2006.

[37] Y.L.Chen, S.L.Tsao, "Energy-efficient sleep-mode operations for broadband wireless access systems", *Proceedings of IEEE 64th Vehicular Technology Conference (VTC-2006 Fall. 2006)*, pp.1-5, Sep 2006.

[38] G.Fang, E.Dutkiewicz, Y.Sun, J.Zhou, J.Shi, Z.Li, "Improving mobile station

energy efficiency in IEEE 802.16e WMAN by burst scheduling", *Proceedings of IEEE Globecom 2006*, pp.1-5, Dec 2006.

[39] K.Lei, D.H.K.Tsang, "Optimal Selection of Power Saving Classes in IEEE 802.16e", *Proceedings of Wireless Communications and Networking Conference(WCNC 2007)*, pp.1836-1841, Mar 2007.

[40] M.H.Bjorndal, "Some Thoughts on Combinatorial Optimization", *European Journal of Operation Research 83*, pp.253-270, 1995.

[41] G.A.Safdar, W.G.Scanlon, "Pointer controlled power saving medium access control protocol for IEEE 802.11 infrastructure networks", *Proceedings of 15th IEEE International conference on Personal, Indoor and Mobile Radio Communications (PIMRC 2004)*, vol.2, pp.915-919, Sep 2004.

[42] G.A.Safdar, W.G.Scanlon, "Performance analysis of an improved power-saving medium access protocol for IEEE 802.11 point coordination function WLAN", *Proceedings of IEE Communications*, vol.153, issue 5, pp.697-704, Oct 2006.

[43] S.Chatterjee, D.Falkenthal, T.Ree, "Exploring Adaptive Power Saving Schemes for Mobile VoIP Devices in IEEE 802.11 Networks", *Proceedings of Second International Conference on Digital Telecommunications (ICDT)*, pp.13-15, Jul 2007.

[44] J.Lorchat, T.Noel, "Energy Saving in IEEE 802.11 communications using frame aggregation", *Proceedings of IEEE Globecom*, vol.3, issue 1, pp.1296-1300, Dec 2003.

[45] M.Sarkar, R.L.Cruz, "A MAC Layer Dynamic Power Management Scheme for Multiple Users in a WLAN", *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC 2007)*, pp.2126-2130, Mar 2007.

[46] V.Rajendran, K.Obraczka, J.J.Garcia-Luna-Aceves, "Energy-efficient MAC: Energy-efficient collision-free medium access control for wireless sensor networks", *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, pp.181-192, Nov 2003.

[47] B.Hohlt, L.Doherty, E.Brewer, "Oral presentation session III: energy efficient design: Flexible power scheduling for sensor networks", *Proceedings of 3rd International Symposium on Information Processing in Sensor Networks*, pp.205-214, April 2004.

[48] K.Han, S.Choi, "Performance analysis of sleep mode operation in IEEE 802.16e mobile broadband wireless access systems", *Proceedings of IEEE 63rd Vehicular Technology Conference (VTC 2006-Spring)*, vol.3, pp.1141-1145, April 2006.

[49] J.B.Seo, S.Q.Lee, N.H.Park, H.W.Lee, C.H.Cho, "Performance analysis of sleep mode operation in IEEE 802.16e", *Proceedings of IEEE 60th Vehicular Technology Conference, 2004 (VTC2004-Fall)*, vol.2, pp.1169-1173, Sep 2004.

[50] X.Yang, "Energy saving mechanism in the IEEE 802.16e wireless MAN", *IEEE Communications Letters*, vol.9, issue 7, pp.595-597, Jul 2005.

[51] Y.Zhang, M.Fujise, "Energy management in the IEEE 802.16e MAC", *IEEE Communications Letters*, vol.10, issue 4, pp.311-313, April 2006.

[52] K.Lei, D.H.K.Tsang, "Performance study of power saving classes of type I and II in IEEE 802.16e", *Proceedings of 31st IEEE Conference on Local Computer Networks*, pp.20-27, Nov 2006.