Computers and Electrical Engineering 38 (2012) 891–905

Contents lists available at SciVerse ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

LBPS: Load-Based Power Saving in the IEEE 802.16e Network ☆,☆☆,☆☆☆

Chun-Chuan Yang [a], Yi-Ting Mai [b,*], Jeng-Yueng Chen [b], Yu-Shiuan Shen [a], Yu-Chen Kuo [a]

[a] Dept. of Computer Science and Information Engineering, National Chi Nan University, Puli, Nantou County, Taiwan, ROC
[b] Dept. of Information Networking Technology, Hsiuping University of Science and Technology, Taichung City, Taiwan, ROC

ARTICLE INFO

Article history:
Received 26 December 2011
Received in revised form 3 March 2012
Accepted 5 March 2012
Available online 28 March 2012

ABSTRACT

Previous works in IEEE 802.16e power saving mainly focused on standard Type I or Type II power saving class. The limitation of adopting Type I or Type II is discussed in the paper, and the idea of Load-Based Power Saving (LBPS) is proposed. The base station in LBPS measures the traffic load and estimates the sleep window size for mobile subscriber stations by setting a threshold for data accumulation. A basic version of LBPS, LBPS-Aggr, is presented in the paper, in which all of the traffic in the network is treated as a single aggregate flow in estimation of the sleep window size. Adopting the idea of group splitting or merging in sleep schedule, two enhanced versions namely LBPS-Split and LBPS-Merge are proposed. Simulation results demonstrate that better power saving efficiency can be achieved by proposed LBPS schemes than standard Type I and Type II.

1. Introduction

IEEE 802.16 (*WiMAX*) [1,2] is an emerging and promising broadband wireless access (BWA) technology that provides high-speed and high-bandwidth wireless access. In 2005, IEEE released the version of IEEE 802.16e [2] (*Mobile BWA*), which enhances the IEEE 802.16 standard to support *mobile subscriber stations* (*MSS*). That is, MSS can roam around anywhere within the range of the network and not to be bound to a single location. As in other wireless networking devices, IEEE 802.16e MSS relies on batteries for power supply, and without proper power management, the energy requires to keep MSS connected to the network over extended periods of time quickly dissipates. Therefore, *power saving* in IEEE 802.16e has been an important issue in recent years.

The most waste of power has been identified as a wireless device such as MSS listening on the radio channel while there is nothing there to receive, thus existing power saving techniques at the MAC layer consist primarily of sleep scheduling protocols, in which the scheduler cycles the radio between *on* and *off* power states to reduce power consumption. Three power saving classes are defined in the standard of IEEE 802.16e, namely Type I, Type II, and Type III, to make sleep scheduling more flexible and accommodate different traffic characteristics of various applications and services. As presented in the next section, a fundamental difference of the three standard classes is the pattern of sleep, which determines the size of sleep windows succeeding the initial sleep window in the case of long idling. Type I, with an exponential increase of the sleep window size, is recommended for non-real-time variable bit rate or best-effort connections. Type II, with a constant sleep window size, is recommended for time-sensitive (real-time) connections. Type III, with the sleep window size controlled by the base

☆ Reviews processed and approved for publication by Editor-in-Chief Dr. Manu Malek.
☆☆ Part of the research work in this paper has been published in Proc. of International Conference on Information and Communication Technologies, May 2010, Tokyo, Japan.
☆☆☆ This work was supported in part by the National Science Council, Taiwan, R.O.C., under Grant No. NSC 98-2221-E-260-009 and NSC 99-2221-E-260-013.

* Corresponding author.
E-mail address: wkb@mail.hust.edu.tw (Y.-T. Mai).

http://dx.doi.org/10.1016/j.compeleceng.2012.03.002

station (BS), is recommended for multicast and management connections, although it is not addressed too much in the standard.

Most of the researches in the literature focused on Type I and Type II. The performance of existing Type I and Type II power saving protocols is mainly affected by two important operational parameters, *the waiting time threshold* and *the sleep window size*. The *waiting time threshold* is the time interval of idling that the MSS waits before entering into the sleep mode, and *the sleep window size* is the duration of sleep. Power saving efficiency can be improved by selecting proper values for the two operational parameters. However, since these protocols inherit the characteristic of Type I or Type II, the sleep pattern in the protocols is limited to either exponential pattern (Type I) or constant pattern (Type II), which implies the limitation of the protocols in dealing with variable bit rate (VBR) connections.

In our opinion, neither exponential nor constant sleep patterns can provide enough capability to effectively deal with power saving for VBR traffic. A straightforward and better method is to proactively model and measure the traffic in the network, and the sleep window size is determined according to traffic parameters obtained from traffic measurement. We assume the arrival process of each connection is *Poisson process*, thus the proposed power saving strategy in the paper is called *Load-Based Power Saving* (*LBPS*). Since the BS is at the best position for traffic measurement, the BS is responsible for notifying each MSS of the sleep window size, which means LBPS belongs to Type III power saving class. A basic version of LBPS called *LBPS-Aggr*, which treats the traffic for all MSSs as aggregate traffic to calculate the sleep window size for MSS, is proposed in the paper. Moreover, two enhancements of *LBPS-Aggr* namely *LBPS-Split* and *LBPS-Merge* are also proposed. Simulation study shows that significantly better efficiency in power saving can be achieved by *LBPS-Aggr* over Type I, and *LBPS-Aggr* also outperforms in power saving efficiency than Type II at the cost of slightly more delay. Simulation results also show that *LBPS-Split* and *LBPS-Merge* can achieve more power saving efficiency than *LBPS-Aggr*.

The rest of the paper is organized as follows. Standard power saving classes in IEEE 802.16e and some related research works are surveyed in Section 2. The basic idea of LBPS and the protocol of LBPS-Aggr are presented in Section 3. Two enhanced protocols LBPS-Split and LBPS-Merge are presented in Section 4. Simulation study and performance comparison are presented in Section 5. Finally, Section 6 concludes this paper.

2. Related work

A In IEEE 802.16e [2], an MSS has two operation modes, *awake* mode and *sleep* mode, in the three standard power saving classes, Type I, II, and III. The awake mode is the normal mode of operation. Two operating windows, *the sleep window* and *the listening window*, are further defined in the sleep mode of Type I and Type II. When a Type I or Type II MSS has no data to transmit or receive for a fixed period of time (called the waiting time threshold), it sends a sleep request message *MOB_SLP-REQ* to the BS. The message carries the information about the class of power saving, the size of the initial sleep window, the size of the final sleep window, and the size the listening window. Upon receiving the response message *MOB_SLP-REP* from the BS, the MSS turns off its radio transceiver and enters into the initial sleep window in the sleep mode. If some data destined to the MSS arrives during its sleep window, the BS buffers the data and sends positive traffic indicator *MOB_TRF-IND* to the MSS in the listening window that follows the initial sleep window. Otherwise, the MSS receives a negative *MOB_TRF-IND* and enters into the next sleep window.

In Type I, the sleep window is increased exponentially until reaching the maximum size or some data has arrived for the MSS to transmit or receive (i.e. positive *MOB_TRF-IND* from the BS) as illustrated in Fig. 1. The specification of IEEE 802.16e recommends Type I is suitable for traffic of *non-real-time variable rate* (*NRT-VR*) service and *best effort* (*BE*) service, such as web browsing. Type II power saving uses an isochronous pattern of the sleep and listening windows and the MSS is allowed to transmit or receive data during listening windows as shown in Fig. 2. The MSS switches back to the awake mode if data transmission cannot be completed in the listening window. Thanks to the constant size of the sleep window, Type II is recommended to support traffic of *real-time variable rate* (*RT-VR*) service and *unsolicited grant service* (*UGS*), such as voice over

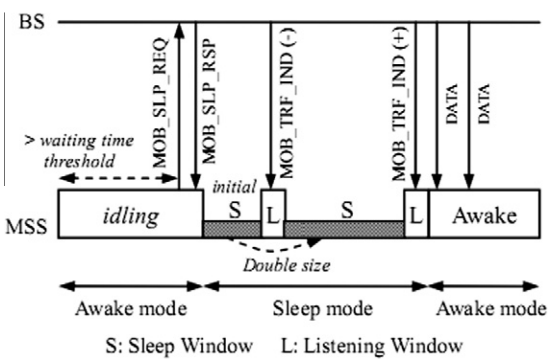

Fig. 1. IEEE 802.16e Type I power saving class.

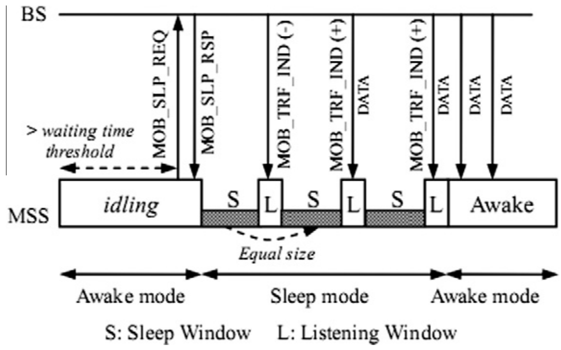

Fig. 2. IEEE 802.16e Type II power saving class.

Internet Protocol (VoIP) and video streaming. As a less addressed power saving class, Type III has no listening windows. An MSS of Type III is activated or deactivated by *TLV* (*Type-Length-Value*) encoding in *RNG_RSP* message sent by the BS as illustrated in Fig. 3. The size of the next sleep window is determined by the offset value in TLV encoding. The MSS switches back to the awake mode if the offset value is zero. Type III is recommended for multicast connections and management operations.

Power saving in wireless network [3–5] is an important issue. Most of the research works for IEEE 802.16e power saving in the literature focused on Type I and II. Performance analysis in terms of power saving efficiency as well as delay performance for the standards was investigated in [6–8]. Jin and Yue [9] proposed a theoretical analysis of Type III power saving class in the case of self-similar multimedia traffic, which was characterized by the Pareto distribution with a batch arrival queueing model.

Enhanced mechanisms to improve power saving efficiency by properly selecting the size of the sleep window were proposed, including heuristic algorithms based on traffic types [10] or traffic loads [11], and enhancements based on stochastic modeling tools to adaptively adjust the sleep window size [4,12–14]. Sanghvi et al. [15] proposed an algorithm to optimally determine the waiting time threshold according to the traffic arrival pattern. In [16], a semi-Markov decision process was used to select the optimal sleep mode between Type I and Type II.

Some research works focused on the design of scheduling mechanisms in the case of multiple real-time and non-real-time connections (multiple power saving classes) [17–20]. Their goal is to minimize power consumption while the QoS of the connections is also guaranteed. The ideas of cycle synchronization [21], harmonization between Type I and Type II connections [22,23], and maximization of unavailability interval [25,26] by applying Chinese Remainder Theorem in scheduling design were also proposed in the literature.

A new sleep mode protocol with the idea of periodic traffic indications was proposed by Hwang et al. [27]. The authors claimed that by introduction of periodic traffic indications, the proposed power saving scheme was better than the IEEE 802.16e standard in terms of simple implementation, reduction of energy consumption, and saving of the resource. As mentioned in Section 1, the idea of applying traffic modeling and measurement in determination of Type III sleep window size has not been addressed in the literature, which is the main goal of this paper.

3. Load-based power saving

3.1. Basic idea

The objective of LBPS is to adaptively adjust sleep window size of each MSS to better fit in current traffic condition (load) by traffic measurement. LBPS achieves this goal by setting a target threshold of data accumulation in the buffer for an MSS

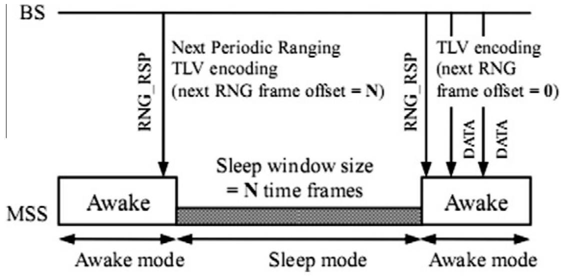

Fig. 3. IEEE 802.16e Type III power saving class.

and dynamically calculating next sleep window size. In this way, LBPS can adapt to different traffic load and still achieves a proper level of powering saving. As illustrated in Fig. 4a, the benefit of powering saving based on data accumulation is that LBPS does not awake the MSS when only a small amount of data arrives, while standard Type I or Type II does not have much chance entering the sleep mode under the same traffic condition as shown in Fig. 4b. Under heavier load, LBPS still expects to achieve a reasonable level of powering saving while Type I or Type II has no gain at all in power saving as displayed in Fig. 5.

There are a couple of things that need to be done in order to realize the goal of LBPS. Firstly, we need a stochastic model to characterize the traffic in the network. In this paper, *Poisson process* is adopted as the modeling tool and *exponential averaging* is used for estimation of the traffic load (rate). Moreover, only downlink traffic (from the BS to MSSs) is considered in the paper for compactness, although LBPS can also easily deal with uplink traffic. Secondly, considering a larger value of the threshold for data accumulation achieves more power saving gain but also results in larger delays, *one time frame* of data should be a suitable upper bound for the threshold. Finally, the sleep window size is calculated as the number of time frames required to reach the threshold of data accumulation. The basic version of LBPS, *LBPS-Aggr*, in which all the traffic in the network is treated as an aggregate flow in calculating the size of the sleep window, is presented in the following.

3.2. LBPS-Aggr protocol

In LBPS-Aggr, the BS needs to estimate the current load in the network (denoted by packets per time frame) by collecting and exponentially averaging the samples of load as in *TCP Round-Trip Time* (*RTT*) estimation. Since the traffic arrival process is assumed to be Poisson, data accumulation under load λ in a time frame is calculated by the following equation

$$Prob\ [i\text{packet arrivals in a time frame}] = \frac{e^{-\lambda T}(\lambda T)^i}{i!}$$

where T is the length of a time frame.

The threshold of data accumulation is denoted by *Data_TH* (packets). The probability of data accumulation exceeding *Data_TH* packets over K time frames in a row can be calculated as follows:

$P_{Acc}(K, Data\_TH) \equiv Prob$ [# of packet arrivals in K time frames > $Data\_TH$]

$$= \sum_{i=Data\_TH+1}^{\infty} \frac{e^{-\lambda KT}(\lambda KT)^i}{i!}$$

$$= 1 - \sum_{i=0}^{Data\_TH} \frac{e^{-\lambda KT}(\lambda KT)^i}{i!}$$

The number of time frames (including the current awake time frame) before the next awake time frame for an MSS is calculated as the smallest value of K such that $P_{Acc}(K, Data\_TH)$ is higher than a predefined probability threshold denoted by *Prob_TH*. That is, *The length of one awake-and-sleep cycle* $\equiv$ *LengthAwkSlpCyl* $(\lambda, Data\_TH) \equiv K^*$

$$= Min\{K | P_{Acc}(K, Data\_TH) \geqslant Prob\_TH\},$$

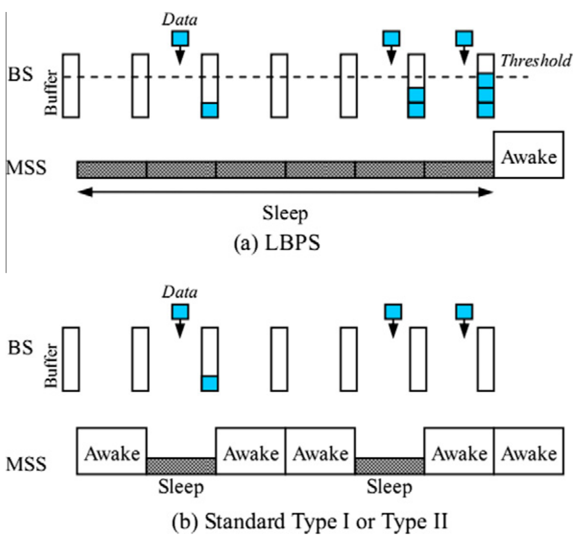

Fig. 4. Power saving in light load: LBPS vs. Type I/II.

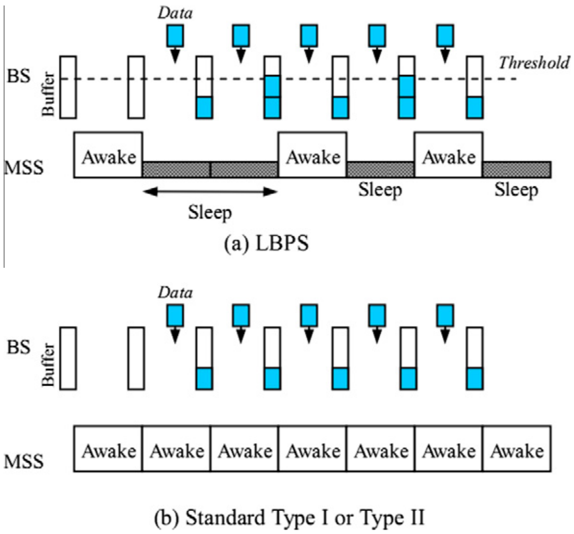

Fig. 5. Power saving in heavy load: LBPS vs. Type I/II.

where an *awake-and-sleep cycle* is composed of the current awake time frame and the following sleep window.

The size of the sleep window in a cycle is therefore K^*-1, which is sent by the BS to the currently awake MSSs to prepare for entering the sleep mode. Since the load in the network may change dynamically, the BS calculates the new value of K^* in each awake time frame of MSS. The protocol of LBPS-Aggr is illustrated in Fig. 6. It is worth mentioning that all of the traffic in the network is treated as an aggregate traffic flow in K^* calculation, but each MSS is independently scheduled for entering the sleep mode in LBPS-Aggr. That is, with the same length of the awake-and-sleep cycle, each MSS may have a different starting time for the cycle.

4. LBPS enhancements

4.1. LBPS-Split

As will be shown in the section of performance evaluation, power saving efficiency of LBPS-Aggr is significantly better than that of the standard Type I or II. Experiments also showed the possibility to further improve the performance of LBPS-Aggr in power saving. Considering the case that $K^* = 2$ (the length of the awake-and-sleep cycle is 2 time frames) in LBPS-Aggr, conceptually it implies all MSSs as a whole should be assigned with one awake time frame out of the cycle of two time frames. But in the schedule we could also split the MSSs into two groups and assign a different awake time frame for each group. Given that the load of a split group is always lighter than the load of the original and bigger group, it is very likely that the new K^* value for each of the split groups (with the same value of *Data_TH*) is larger than the original value of 2. The case of the minimal value of the two new K^* values larger than 2 implies the feasibility of further splitting, which leads to an enhanced LBPS protocol namely *LBPS-Split* in the paper.

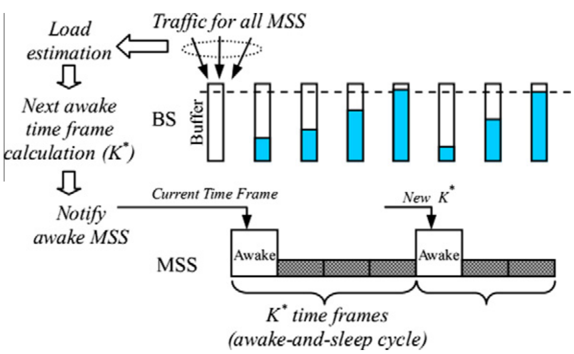

Fig. 6. LBPS-Aggr protocol.

An example of LBPS-Split with seven MSSs is illustrated in Fig. 7. All of the seven MSSs is treated as one group (as does in LBPS-Aggr) in the first step. The value of $K_G^* = 2$ in the first step leads to the splitting of the MSSs into two groups in the second step. The length of the awake-and-sleep cycle for each group is re-calculated, and the minimal value of K_{G1}^* and K_{G2}^* in the second step leads to three split groups in the third step. The splitting process continues until the new minimal value of K^* remains unchanged as in the fourth and the fifth step in the example. The final value of K^* is the length of the awake-and-sleep cycle for all the split groups, and each group is assigned with a different awake time frame by the BS as displayed in the figure. The algorithm of LBPS-Split for the general case of any number of MSSs is displayed in Fig. 8.

Power saving performance of LBPS-Split is inevitably affected by the splitting mechanism. In order to maximize power saving efficiency, the minimal value of K^* in each iteration should be maximized. Therefore, the splitting mechanism should try to divide the total load to each split group as equally as possible in order to minimize load difference among the groups.

4.2. LBPS-Merge

The best case of LBPS-Split in power saving is that each of the split group is composed of a single MSS, and the final value of K^* is therefore determined by the MSS with least load. In such case, with the same length (the final K^*) of the awake-and-sleep cycle, each MSS is assigned with one whole awake time frame in a cycle. The idea leads to another perspective of grouping MSSs. Instead of treating all MSSs as one group from the start, we could firstly make each MSS a single-member group for K^* calculation. Since the load of each MSS varies, each group usually has a different value of K^*. In order to achieve a better gain of power saving, the sleep scheduling algorithm should be able to accommodate different values of K^* as long as a feasible sleep schedule can be found. In the case that a feasible sleep schedule can not be found for the current status of grouping, merging of some groups has become necessary. The idea of treating each MSS as a single-member group from the start and merging groups when necessary leads to another enhanced protocol called *LBPS-Merge*. Since it's difficult to check the schedulability of groups with any possible value of K^*, the value of K^* is converted to the closest and smaller power of 2, denoted by $K^\#$ (i.e. $K^\# = 2^{\lfloor \text{Log}_2 K^* \rfloor}$) in LBPS-Merge. With the property of powers of 2, a quick check for schedulability can be obtained. Schedulability of a number of groups with different $K^\#$ values is defined by the following equation

$$Schedulability = \sum_i \frac{1}{K_i^\#}$$

Schedulability equal or smaller than 1 ($Schedulability \leqslant 1$) indicates that a feasible schedule can be found. *Schedulability* > 1 indicates the necessity of merging some groups. Group merging should not reduce as much power saving efficiency as possible, which means the value of $K^\#$ after group merging should be kept as larger as possible. Therefore, the merging process in LBPS-Merge is divided into two phases: (1) *non-degraded merge* and (2) *degraded merge*. Merging of two groups that does not result in a smaller value of $K^\#$ is called a non-degraded merge. A degraded merge is accepted only when a non-degraded merge cannot be found.

An example of LBPS-Merge with seven MSSs is illustrated in Fig. 9. For better efficiency in merging, the load of each group is sorted in the ascendant order. The initial value of K^* for each single-member group is converted to its corresponding $K^\#$ during the first step. The result of *Schedulability* > 1 presents the necessity of group merging. As illustrated in Part I of Fig. 9, the merging process starts from groups with smaller loads and looks for non-degraded merge. A non-degraded merge is found by merging MSS_1 and MSS_3. The new grouping result is shown in Part II of the figure, in which the value of $K^\#$ for each group is updated and sorted. Since the value of *Schedulability* is still larger than 1, group merging is triggered again. As shown

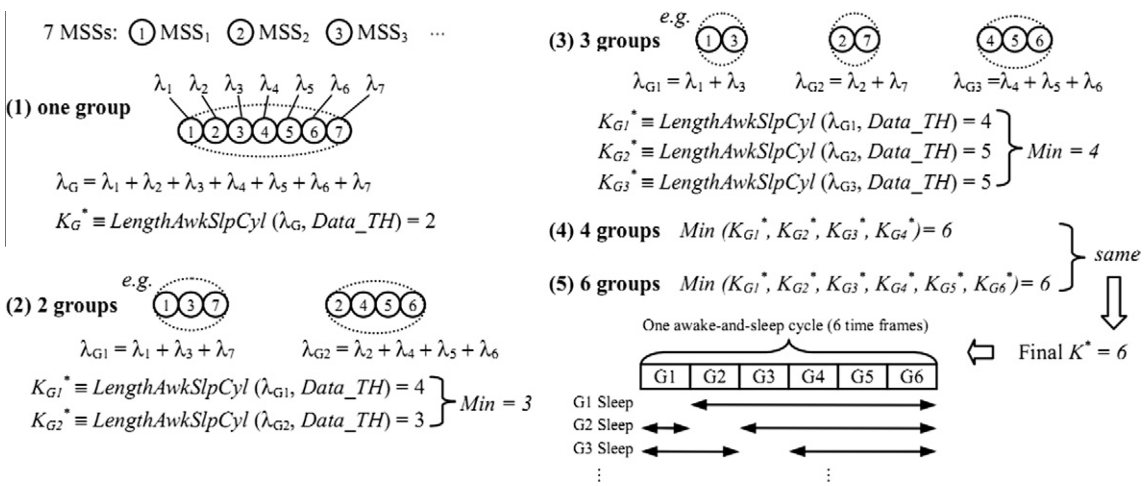

Fig. 7. An example of LBPS-Split.

1. Estimate the current load of each $MSS_i = \lambda_i$
   Initial number of groups = $K^* = 1$
2. Split all MSSs to K^* groups, $G_1 \sim G_{K^*}$
   For each group G_i, calculate K_{Gi} =

   $$LengthAwkSlpCyl\ (\lambda_{Gi},\ Data\_TH),$$

   $$\text{where } \lambda_{Gi} = \sum_{MSS_m \in Gi} \lambda_m$$

   New_K^* = the smallest one among K_{Gi}
   If $New\_K^* \neq K^*$, set $K^* = New\_K^*$ and repeat step 2
   Else go to step 3.
3. Schedule K^* groups into awake-and-sleep cycle with length K^* time frames. Each group is assigned with a different awake time frame.

Fig. 8. Algorithm of LBPS-Split.

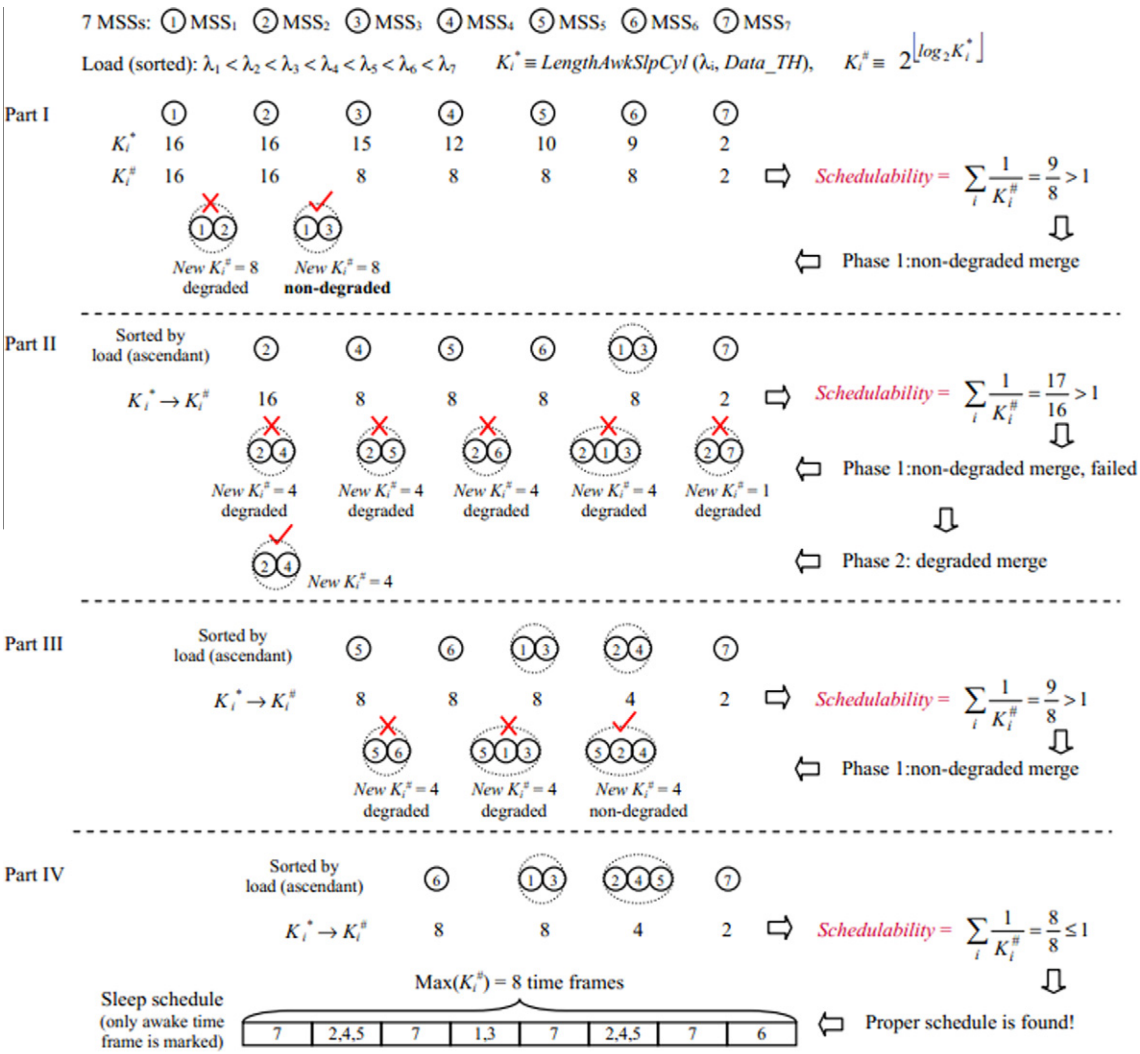

Fig. 9. An example of LBPS-Merge.

in Part II of the figure, a degraded merge for the two groups (MSS_2 and MSS_4) with smallest loads is accepted since no non-degraded merge can be found. The merging process continues until $Schedulability \leqslant 1$, and a feasible sleep schedule for all groups can be easily obtained as displayed in Part IV of the figure. The algorithm of LBPS-Merge is displayed in Fig. 10.

5. Performance evaluation

5.1. Performance analysis of LBPS-Aggr

In the normal operation of LBPS-Aggr, in which transmission of the data accumulated in K^* time frames can be finished in one awake time frame, the power saving efficiency is $\frac{K^*-1}{K^*}$. If the amount of the accumulated data cannot be finished transmission in a time frame, the MSS must be stay awake until all of its data is cleared. In the case, the power saving efficiency becomes $\frac{K^*-1}{K^*+N_{ext}}$, where N_{ext} is the number of extra awake time frame to clear out the accumulated data. Therefore, the average power saving efficiency (denoted by *PSE*) for an MSS is calculated as the following equation

$$PSE = \sum_{i=0}^{\infty}\left(\frac{K^*-1}{K^*+i} \times Prob[N_{ext} = i]\right) \tag{1}$$

We assume the packet arrival time at the BS is uniformly distributed among the time frames in one awake-and-sleep cycle, the average access delay for a packet is the half of the cycle length. Therefore, the average access delay for a packet (denoted by *AvgDelay*) considering different cycle length is calculated as the following equation

$$AvgDelay = \sum_{i=0}^{\infty}\left(\frac{K^*+i}{2} \times Prob[N_{ext} = i]\right) \tag{2}$$

5.2. LBPS-Aggr vs. standards

Simulation study was conducted to compare the performance of LBPS-Aggr, standard Type I, and standard Type II, in terms of power saving efficiency as well as the average access delay. Parameters used in the simulation are listed in Table 1 Note that the threshold of data accumulation *Data_TH* in LBPS-Aggr is set as a full time frame, but since each MSS operates its awake-and-sleep cycles independently of others, the accumulated data for concurrently awake MSSs can be cleared out in one time frame in most of the time, i.e. $Prob[N_{ext} = 0] \approx 1$.

Power saving efficiency (*PSE*) of LBPS-Aggr and Type I under different input loads is displayed in Fig. 11, which also includes the numerical result from Eq. 1. The figure demonstrates a significantly better power saving performance of LBPS-Aggr over Type I. In Fig. 12, LBPS-Aggr is compared with an enhanced scheme of standard Type I called LISA (Listening Interval Spreading Approach) [19]. The idea of LISA is to eliminate the idle time prior sleep mode, where the potential wasted idle period to enable sleep mode is collected and spread at the early stage of growing sleep pattern. Although compared with Type I, LISA presents better power saving efficiency in load under 0.5, Fig. 12 shows LBPS-Aggr significantly outperforms LISA since LISA inherits the weakness of Type I as mentioned in Section 1.

In order to investigate the performance of LBPS-Aggr for traffic with delay constraint (*RT-VR*), LBPS-Aggr can be easily extended to support the requirement of bounded delay, denoted by $D_{RT\text{-}VR}$. LBPS-Aggr calculates the value of K^* as presented in Section 3, but the final value of the awake-and-sleep cycle is set as the smaller one of K^* and $D_{RT\text{-}VR}$. Fig. 13 displays the power

1. Estimate the current load of each $MSS_i = \lambda_i$, each MSS_i initially forms a group, λ_G = the total load in a group.
2. Sort the groups in the ascendant order of load.

   For each group, calculate $K_G^* = LengthAwkSlpCyl\ (\lambda_G, Data\_TH)$, and convert K_G^* to the closest and smaller power of 2 = $K_G^{\#} = 2^{\lfloor \log_2 K_G^* \rfloor}$.
3. If $Schedulability = \sum \frac{1}{K_G^{\#}} \leq 1$, go to step 4.

   Else {

   Phase 1: *Non-degraded merge*

   Try to merge the smallest load group to another group until a non-degraded merge is found.

   If a non-degraded merge is found, repeat step 2 & 3 , else perform Phase 2.

   Phase 2: *Degraded merge*

   Merge the two groups with the smallest loads and repeat step 2 & 3.

   }
4. Schedule the groups according to the final set of $K_G^{\#}$.

Fig. 10. Algorithm of LBPS-Merge.

Table 1
Simulation parameters.

# of MSS	10, 20, 40, 80
Load distribution	Equal, 8:2, Random
Time frame size	160 mini-slots, 1 mini-slot = 1 packet
Type I initial sleep window size	2^0 time frame
Type I max sleep window size	2^9 time frames
Type II sleep window size	2 time frames
Listening window size (Type I, II)	1 time frame
Data_TH (LBPS-Aggr)	160 packets
Data_TH (LBPS-Split, LBPS-Merge)	0.8 ∗ 160 packets
Prob_TH (LBPS schemes)	0.8

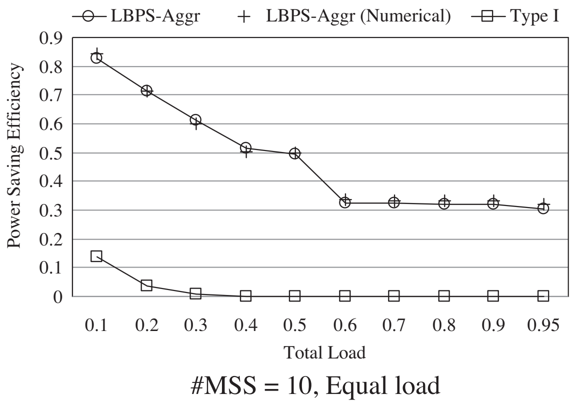

Fig. 11. PSE: LBPS-Aggr vs. Type I.

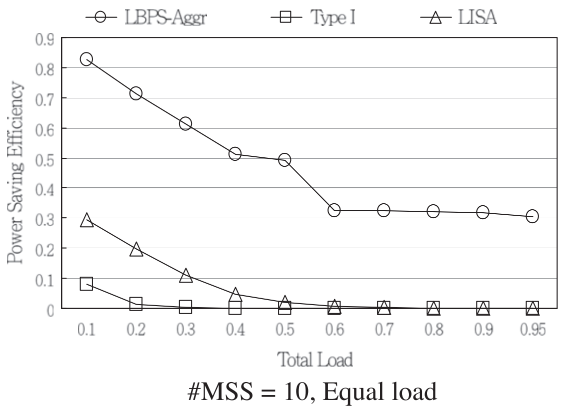

Fig. 12. PSE: LBPS-Aggr vs. LISA.

saving efficiency of LBPS-Aggr and Type II for RT-VR traffic, in which $D_{RT\text{-}VR}$ = 3 time frames. Delay performance (*AvgDelay*) for the case of Fig. 13 is shown in Fig. 14. The two figures demonstrate that a better power saving efficiency can be achieved by LBPS-Aggr over Type II at the cost of slightly more access delay.

5.3. LBPS-Aggr vs. LBPS-Split and LPBS-Merge

Simulation results for comparing three LBPS schemes in terms of *PSE* and *AvgDelay* in the case of 10 MSSs with *equal load* are displayed in Figs. 15 and 16, respectively. Note that the MSSs in a same group in LBPS-Split and LBPS-Merge are scheduled to be awake in the same time frames, thus the value of *DATA_TH* for the two enhanced schemes is set as 80% of a time frame (i.e. 0.8 ∗ 160 packets as shown in Table 1) to reduce the probability of data overflow in an awake time frame. As

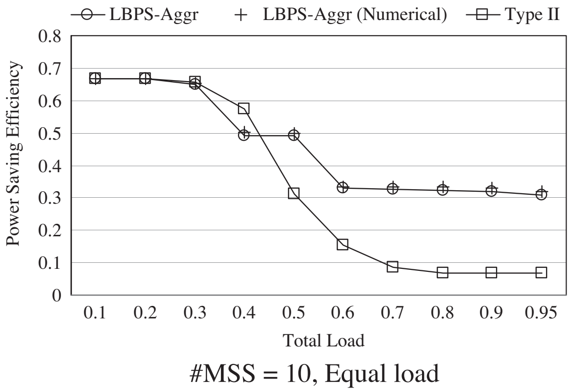

Fig. 13. PSE: LBPS-Aggr vs. Type II.

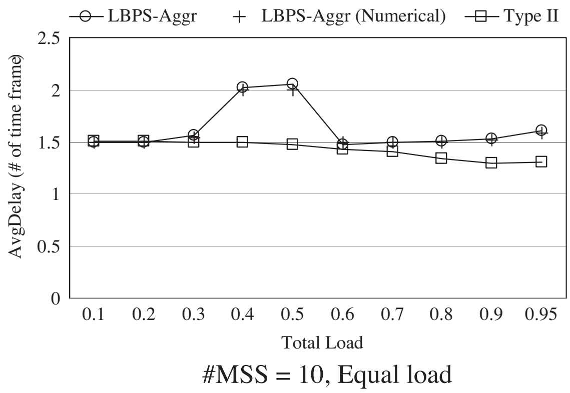

Fig. 14. AvgDelay: LBPS-Aggr vs. Type.

shown in Fig. 15, power saving efficiency of LBPS-Split and LBPS-Merge is significantly better than LBSP-Aggr, which correspondingly leads to larger *AvgDelay* as displayed in Fig. 16. Moreover, since LBPS-Aggr is a special case of LBPS-Split, *DATA_TH* of LBPS-Split under very heavy load is set as the original 160 packets (one time frame of data) in the simulation and thus the performance results of LBPS-Split and LBPS-Aggr under load 0.9 and load 0.95 converge at same points as displayed in Figs. 15 and 16.

Performance comparison of LBPS-Split and LBPS-Merge yields different results under different traffic load. As illustrated in Fig. 15, *PSE* of the two enhanced schemes is similar under lighter load (below 0.5). Power saving efficiency of LBPS-Split outperforms LBPS-Merge when the traffic load is 0.6 and 0.7, but LBPS-Merge outperforms LBSP-Split in heavy load above

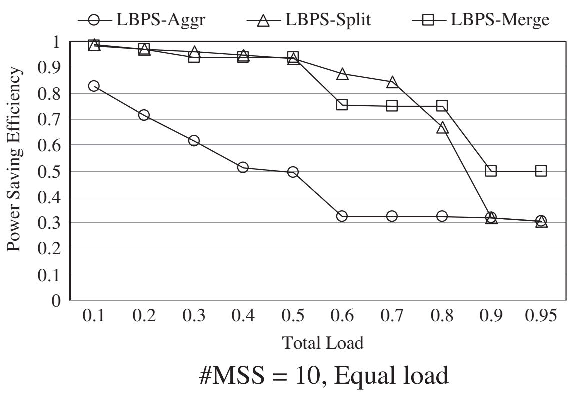

Fig. 15. PSE: LBPS-Aggr vs. -Split vs. -Merge.

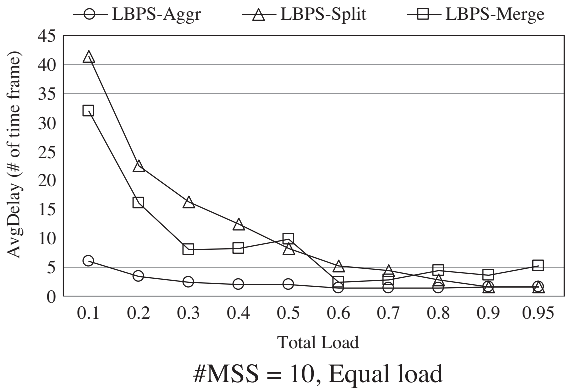

Fig. 16. AvgDelay: LBPS-Aggr vs. -Split vs.-Merge.

0.8. The reason for different results under different load lies in the impact of the key features of the two schemes. LBPS-Split equalizes the length of the awake-and-sleep cycle for all groups, which presents a limitation in sleep scheduling. On the other hand, LBPS-Merge accommodates different lengths of the awake-and-sleep cycle, but the length must conform to a power of 2 for schedulability, which in a way introduces some loss in power saving efficiency. Better *PSE* of LBPS-Split under load 0.6 and 0.7 in Fig. 15 reflects the smaller impact of equalizing cycle length than the conformity to the power of 2 in LBPS-Merge. Better *PSE* of LBPS-Merge under heavy load above 0.8 reflects more gain in power saving to accommodate different cycle length in sleep scheduling.

Figs. 17 and 18 display the performance results of *PSE* and *AvgDelay* in the case of "8:2 load", in which 80% of the traffic load goes to 20% of the MSS. As displayed in Fig. 17, *PSE* of LBPS-Split in the load under 0.6 is worse than LBPS-Merge. The reason is again due to the limitation of equalizing the cycle length in LBPS-Split. Furthermore, since the cycle length in LBPS-Split is the minimal value of *K* among all groups, *PSE* of LBPS-Split is bounded by the group with the heaviest load. It gives the reason why *PSE* of LBPS-Split is lower in the case of higher variation in traffic load distribution.

5.4. Impact of the number of MSS and load distribution

Results of power saving efficiency of LBPS-Split and LBPS-Merge under different numbers of MSS are displayed in Figs. 19 and 20, respectively. Both figures demonstrate that better *PSE* is achieved for a larger number of MSS, since a larger number of MSS provides more flexibility in splitting or merging groups and thus more gain in *PSE* can be obtained. Figs. 21 and 22 display the results of *PSE* of the two enhanced schemes under three different load distributions: *equal load*, *random load*, and *8:2 load*. As demonstrated in both figures, higher variation of load distribution makes lower *PSE* in both schemes, although the variation of load distribution presents a more consistent impact on LBPS-Split than LBPS-Merge.

5.5. Discussion

(1) Power Saving Efficiency (PSE) defined in this paper is the ratio of the sleep mode (window) over time for the MSS, which does not represent the accurate performance in reduction of power consumption. Although the modeling of the exact power consumption at mobile communication devices is complicated, a couple of more aspects should be

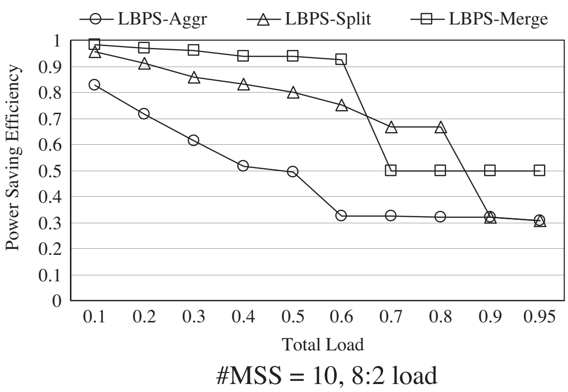

Fig. 17. PSE: LBPS-Aggr vs. -Split vs. -Merge.

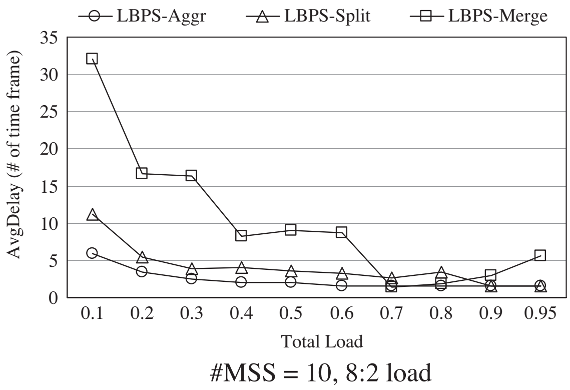

Fig. 18. AvgDelay: LBPS-Aggr vs. -Split vs. -Merge.

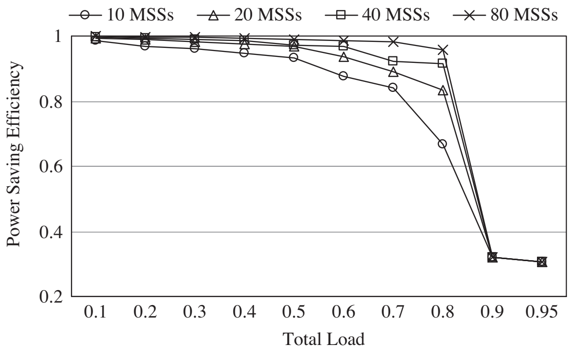

Fig. 19. PSE of LBPS-Split (Equal load).

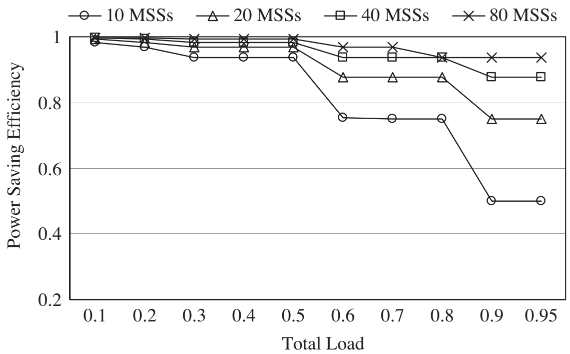

Fig. 20. PSE of LBPS-Merge (Equal load).

considered in obtaining more accurate performance of power saving in LBPS mechanisms. Firstly, power consumption in the sleep mode should be estimated carefully, instead of the simple "zero" assumption. Secondly, since more buffers and buffer operations are required to support LBPS mechanisms, increased power consumption due to a larger size of memory in the hardware design should be also taken into consideration. Lastly, the increase in power consumption at the BS for supporting LBPS mechanisms should be also considered. More accurate modeling in evaluation of power reduction in LBPS mechanisms is left as the future work of the research.

(2) As the first attempt to make use of traffic modeling in sleep scheduling, the simple tool of Poisson distribution is adopted in this paper for demonstration purpose. More sophisticate distributions such as self-similar or Pareto distributions are left as the next step of our research, in which calculation of the sleep window size in LBPS schemes adopting distribution like Pareto requires more elaboration (theoretically or numerically) in order to capture the characteristic of traffic involving multiplexing multiple connections over multiple time frames.

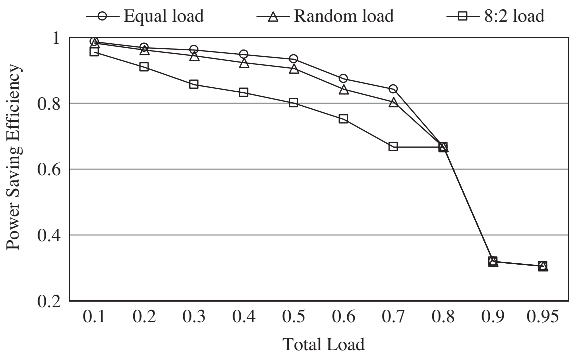

Fig. 21. PSE of LBPS-Split (10 MSSs).

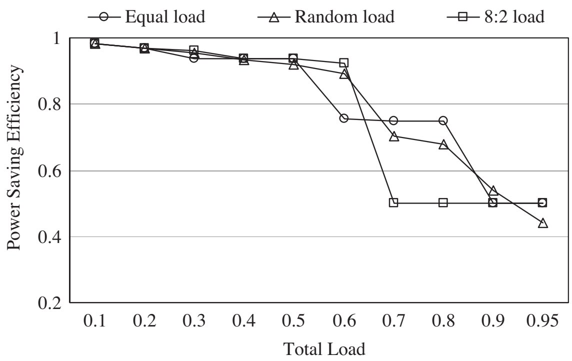

Fig. 22. PSE of LBPS-Merge (10 MSSs).

(3) Data_TH and *Prob_TH* are user-selected parameters in the proposed schemes. *Data_TH* serves as the threshold for data accumulation in calculating the sleep window size. Since non-real-time traffic is the main focus, and in order to make the best of the awake time frame, data accumulation should make the most of the awake time frame, which implies "one time frame of data" is the basic unit of *Data_TH*. Increasing *Data_TH* makes better power saving performance at the cost of increased access delay. Therefore, as a heuristic and typical option, the case of *Data_TH* = 1 time frame of data is investigated in the simulation.

On the other hand, the value of *Prob_TH* indicates the probability of data accumulation reaching *Data_TH*. A large value of *Prob_TH* (e.g. 0.9 or even higher) introduces the risk of too much data accumulated, while a small value (e.g. below 0.5) presents high possibility of too little data. According to the common rule of thumb, *Prob_TH* = 0.8 is adopted in the simulation.

6. Conclusion

Most of the research works in IEEE 802.16e power saving focused on standard Type I or Type II, which means these previous works inherited the limitation of Type I or Type II in selection of the sleeping pattern: either adopting the exponential pattern of Type I or the constant pattern of Type II for the sleep window size. In this paper, the idea of applying traffic modeling and measurement mechanisms to adaptively determine the sleep window size that can fit for different traffic loads, which is called Load-Based Power Saving (LBPS), has been proposed. LBPS models and measures the traffic in the network, and estimates the sleep window size by setting a proper threshold for data accumulation.

A basic version of LBPS, called LBPS-Aggr, in which all of the traffic is treated as an aggregate flow for estimating the sleep window size, has been proposed in the paper. Two enhanced LBPS schemes, namely LBPS-Split and LBPS-Merge have also been proposed. Instead of treating all traffic as a single aggregate flow, LBPS-Split splits MSSs into different groups in sleep scheduling to achieve more power saving efficiency. A reverse direction is adopted in LBPS-Merge, in which each MSS is treated as a single-member group in the beginning, and the operation of group merging is repeated until a feasible sleep schedule is obtained.

Simulation study has demonstrated that LBPS-Aggr significantly outperforms Type I in power saving, and with the constraint of bounded delay, LBPS-Aggr outperforms Type II in power saving at the cost of slightly more delay. Furthermore, LBPS-Split and LBPS-Merge can achieve even better power saving efficiency than LBPS-Aggr. Impact of the number of MSS and the variation of load distribution on the performance of power saving were discussed in the paper.

References

[1] IEEE Std. 802.16-2004, ''IEEE Standard for Local and Metropolitan Area Networks – Part 16: Air Interface for Fixed Broadband Wireless Access Systems'', Oct. 2004.

[2] IEEE Std. 802.16e-2006, ''IEEE Standard for Local and Metropolitan Area Networks – Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems -Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands'', Feb. 2006.

[3] Nascimento A., Rodriguez J. Cluster cooperation for energy saving in wireless personal networks. In: Proceedings of IEEE International Conference on Computer as a Tool (EUROCON), April 2011, pp. 1–4.

[4] Hsu C-H, Feng K-T. A Statistical Power-saving Mechanism for IEEE 802.16 Networks. In: Proceedings of 2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2009), Sept. 2009, pp. 27-31.

[5] Chamam A, Pierre S. A distributed energy-efficient clustering protocol for wireless sensor networks. Comp Electric Engineer 2010;36(2):303–12.

[6] Han K, Choi S. Performance Analysis of Sleep Mode Operation in IEEE 802.16e Mobile Broadband Wireless Access System. In: Proceedings of 63rd IEEE Vehicular Technology Conference (VTC 2006-Spring), vol. 3, May 2006, pp. 1141–1145.

[7] Kong L, Tsang DHK. Performance Study of Power Saving Classes of Type I and II in IEEE 802.16e. In: Proceedings of 31st IEEE Conference on Local Computer Networks (LCN 2006), Nov. 2006, pp. 20–27.

[8] Xiao Y. Performance Analysis of an Energy Saving Mechanism in the IEEE 802.16e Wireless MAN. In: Proceedings of 3rd IEEE Consumer Communications and Networking Conference (CCNC), vol. 1, Jan. 2006, pp. 406–410.

[9] Jin S, Yue W. Performance Evaluation of Self-similar Traffic in Multimedia Wireless Communication Networks with Power Saving Class Type III in IEEE 802.16e. In: Proceedings of 2010 IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS), June 2010, pp. 436–440.

[10] Jang J, Han K, Choi S. Adaptive Power Saving Strategies for IEEE 802.16e Mobile Broadband Wireless Access. In: Proceedings of Asia-Pacific Conference on Communications (APCC'06), Aug. 2006, pp. 1–5.

[11] Xue J, Yuan Z. An Adaptive Power Saving Strategies based on Cross-layer Design in IEEE 802.16e. J Networks 2010;5(3):359–66.

[12] Hsu C-H, Feng K-T, Chang C-J. Statistical Control Approach for Sleep Mode Operations in IEEE 802.16m Systems. IEEE Transactions on Vehicular Technology, vol. 99, Aug. 2010, pp. 1–12.

[13] Lee JR, Cho DH. Performance Evaluation of Energy-Saving Mechanism Based on Probabilistic Sleep Interval Decision Algorithm in IEEE 802.16e. IEEE Trans Vehicular Technol July 2007;56(4):1773–80.

[14] Zhang Y, Xiao Y, Leung VCM. Energy Management Analysis and Enhancement in IEEE 802.16e WirelessMAN. IEEE Trans Vehicular Technol 2009;58(7):3738–52.

[15] Sanghvi K, Jain PK, Lele A, Das D. Adaptive Waiting Time Threshold Estimation Algorithm for Power Saving in Sleep Mode of IEEE 802.16e. In: Proceedings of 3rd International Conference on Communication System Software and Middleware (COMSWARE), Jan. 2008, pp. 334–340.

[16] Lei K, Tsang DHK. Optimal Selection of Power Saving Classes in IEEE 802.16e. In: Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), March 2007, pp. 1836–1841.

[17] Ferng H-W, Lin C-L, Huang J-J, Tseng C-C. A Scheduling-Based Delay-Tolerant Power Saving Scheme for the IEEE 802.16e Wireless MAN. In: Proceedings of IEEE 67th Vehicular Technology Conference (VTC 2008-Spring), May 2008, pp. 2321–2325.

[18] Chen JJ, Wu SL, Wang SW. Power Saving Class Management for Energy Saving in IEEE 802.16e Wireless Networks. In: Proceedings of 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware (MDM'09), May 2009, pp. 490–495.

[19] Sheu S-T, Cheng Y-C, Chen L-W. Listening Interval Spreading Approach (LISA) for Handling Burst Traffic in IEEE 802.16e Wireless Metropolitan Area Networks. In: Proceedings of IEEE 68th Vehicular Technology Conference (VTC 2008-Fall), Oct. 2008, pp. 1–5.

[20] Chen Y-L, Tsao S-L. Energy-Efficient Sleep-Mode Operations for Broadband Wireless Access Systems. In: Proceedings of IEEE 64th Vehicular Technology Conference (VTC 2006-Fall), Sept. 2006, pp. 1–5.

[21] Wu Y, Le Y, Zhang D. An Enhancement of Sleep Mode Operation in IEEE 802.16e Systems. In: Proceedings of IEEE 69th Vehicular Technology Conference (VTC 2009-Spring), April 2009, pp. 1–6.

[22] Kwon S-W, Cho D-H. A Harmonization Among Power Saving Class in the IEEE 802.16e Systems. In: Proceedings of 2010 7th IEEE Consumer Communications and Networking Conference (CCNC), Jan. 2010, pp. 1-2.

[23] Kwon S-W, Cho D-H. Enhanced Power Saving through Increasing Unavailability Interval in the IEEE 802.16e Systems. IEEE Commun Lett 2010;14(1):24–6.

[25] Chen TC, Chen JC. Extended Maximizing Unavailability Interval (eMUI) Maximizing Energy Saving in IEEE 802.16e for Mixing Type I and Type II PSCs. IEEE Commun Lett 2009;13(2):151–3.

[26] Chen TC, Chen JC. Maximizing Unavailability Interval for Energy Saving in IEEE 802.16e Wireless MANs. IEEE Trans Mobile Computing 2009;8(4):475–87.

[27] Hwang E, Kim KJ, Son JJ, Choi BD. The power-saving mechanism with periodic traffic indications in the IEEE 802.16e/m. IEEE Trans on Vehicular Technol Jan. 2010;59(1):319–34.

Chun-Chuan Yang received his B.S. degree in computer and information science from National Chiao-Tung University, Taiwan, in 1990 and Ph.D. degree in computer science from National Taiwan University in 1996. He joined the Department of Computer Science and Information Engineering, National Chi-Nan University (NCNU), Puli, Taiwan, as an assistant professor in 1998. Since February 2008, he has been a full professor. His research area of interests includes mobile networking, QoS supporting, power saving, and 4G LTE/LTE-A.

Yi-Ting Mai received his B.S. degree in Department of Science Education from National Hualien University of Education, Hualien, Taiwan, in 1998 and M.S. degrees in Department of Information Management from Chaoyang University of Technology, Taichung, Taiwan, in 2004. He got Ph.D. degree in Department of Computer Science and Information Engineering, National Chi Nan University (NCNU), Puli, Taiwan, in 2008. He joined the Department of Information and Networking Technology, Hsiuping University of Science and Technology, Dali, Taiwan, R.O.C., as an assistant Professor in 2008. His current research topics include wireless networks, QoS, power saving, multimedia network protocols, and 4G LTE/LTE-A.

Jeng-Yueng Chen received his Ph.D. degree in Computer Science and Information Engineering from National Chi Nan University in 2009. Currently, he is an assistant professor of Department of Information Networking Technology at Hsiuping University of Science and Technology. His research interests cover routing, mobile computing, network management SIP-based applications, and 4G LTE/LTE-A.

Yu-Shiuan Shen received his B.C. degree in Department of Computer Science and Information Engineering from Ming Chuan University, Taoyuan, Taiwan, in 2007 and M.S. degree at Department of Computer Science and Information Engineering from National Chi Nan University, Puli, Taiwan, in 2009. His current research topic includes power saving in wireless access network.

Yu-Chen Kuo received his B.C. degree in Department of Computer Science and Information Engineering from Chi Nan University, Puli, Taiwan, in 2009 and M.S. degree at Department of Computer Science and Information Engineering from National Chi Nan University, Puli, Taiwan, in 2011. His current research topic includes power saving in wireless access network.