Design of Load-based Power Saving and Scheduling Scheme Integrating Real-time and Non-real-time Services in WiMAX Networks

Yi-Ting Mai[1], Jeng-Yueng Chen[2], Min-Hsien Lin[3]

[1] Department of Sport Management, National Taiwan University of Sport, Taiwan
[2] Department of Information & Networking Technology, Hsiuping University of Science & Technology, Taiwan
[3] Department of Computer Science & Information Engineering, National Chi Nan University, Taiwan
wkb@ntupes.edu.tw, jychen@hust.edu.tw, baseballgamebh@gmail.com

Abstract

There are three types of power saving classes in the specification of IEEE 802.16e/m: Type I, II, and III. Most of the researches focused mainly on Type I or Type II, which means these previous works could also inherit the limitation of how to select the sleeping pattern in Type I and Type II: either adopting the exponential pattern of Type I or adopting the constant pattern of Type II for the sleep window size. In our previous research, we discussed the limitation of adopting Type I or Type II, and had brought up the idea of applying traffic modeling and measurement called Load-Based Power Saving (LBPS). Poisson process is adopted to simplify LBPS for traffic modeling. The objective of LBPS is to adaptively adjust the sleep window size of each MSS to better fit in the current traffic (load) condition by traffic measurement. To consider both real-time and non-real-time traffic characteristics, a new LBPS scheme, namely LBPS-RT, is proposed in this paper to effectively schedule traffic and support the traffic delay requirement. Simulation results demonstrate that better power-saving efficiency can be achieved by the LBPS-RT than the standard Type I and II. In addition, the delay bound requirement can be achieved in our new LBPS-RT scheme.

Keywords: LBPS, Power saving, 802.16, Sleep schedule, Real-time

1 Introduction

IEEE 802.16 (WiMAX) [1-3] is an emerging and promising broadband wireless access (BWA) technology that provides high-speed and high-bandwidth wireless access. As for the terminal mobility issue, IEEE released the version of IEEE 802.16e [1] (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 be restricted to a single location. Like other wireless networking devices, IEEE 802.16e MSS relies on batteries, and without proper power management, the energy that keeps MSS connected to the network over an extended period of time will quickly dissipate. Therefore, power saving in IEEE 802.16e has been an important issue in recent years.

We formerly discussed a basic version of LBPS, LBPS-Aggr, in which all of the traffic in the network is treated as a single aggregate flow in estimation of the sleep window size [5]. Two enhanced versions, namely LBPS-Split and LBPS-Merge, were also proposed [6]. The two enhanced schemes adopt the idea of grouping the split or merging the sleep schedules to achieve the best power saving. These schemes mentioned mainly focused on non-real-time traffic. In this paper, a new scheme called LBPS-RT is introduced to integrate both real-time and non-real-time traffic for power saving.

This paper is organized as follows. Standard power saving classes in IEEE 802.16e and some related field researches are in section 2. The protocol of LBPS-RT is presented in section 3. Simulation study and performance comparison are listed in section 4. Finally, we will conclude this paper in section 5.

2 Related Work

In IEEE 802.16e/m [1-4], 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 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 within a fixed time (called the waiting time threshold), the MSS would change into the sleep mode. Generally, an MSS in the sleep mode wakes up to exchange signaling messages with the *Base Station (BS)* during a listening window, and then

*Corresponding Author: Jeng-Yueng Chen; E-mail: jychen@hust.edu.tw
DOI: 10.3966/160792642020092105004

turn the power down during a sleep window to save the power consumption. The interleaved listening and sleep windows repeat every sleep cycle as long as the MSS is in the sleep mode.

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. The specification of IEEE 802.16e recommends that Type I is suitable for the traffic of *non-real-time variable rate (NRT-VR)* service and *best effort (BE)* service. Type II power saving uses an isochronous pattern of sleep and listening windows, and the MSS is allowed to transmit or receive data during listening windows, and then the MSS switches back to the awake mode if data transmission cannot be completed in the listening window. Type II is recommended to support traffic of *real-time variable rate (RT-VR)* service and *unsolicited grant service (UGS)*. As a less discussed 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 control message sent by the BS. 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. Therefore, Type III is recommended for multicast connections and management operations.

Power saving in the wireless network domain [7-14] is an important and hot issue in recent years. 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 and delay performance for the standards was investigated in [15]. Jin and Yue [16] 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 [17] or traffic loads [18], and enhancements based on stochastic modeling tools to adaptively adjust the sleep window size [19-22] in IEEE 802.16e/m. Sanghvi et al [23] proposed an algorithm to optimally determine the waiting time threshold according to the traffic arrival pattern. In [24], the authors proposed an approach with the *adaptive listening window (ALW)* improving the power-saving efficiency through making some dynamic adjustments to the sleep internal time in PSC II. In [25], 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) [26-29]. Their goal is to minimize power consumption while the QoS of the connections is also guaranteed. The ideas of cycle synchronization [30], harmonization between Type I and Type II connections [31-32], and maximization of unavailability interval by applying Chinese Remainder Theorem in scheduling design were also proposed in [33]. In [34-35], to identify the power-saving efficiency with delay contract, the appropriate parameters should be designed for each traffic requirement.

In our opinion, neither exponential nor constant sleep patterns can provide enough capability to deal effectively with power saving for VBR traffic. A better and direct 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. The idea of *Load-Based Power Saving (LBPS)* was proposed in our previous paper [6]. A basic LBPS scheme namely LBPS-Aggr and two enhanced schemes, namely LBPS-Split and LBPS-Merge, were also proposed. In this paper, an extension of LBPS-Merge integrating power saving of real-time and non-real-time traffics is proposed.

3 Real-time Load-based Power Saving

3.1 Previous Work about LBPS

The objective of LBPS is to adaptively adjust sleep window size of each MSS to better fit in the current traffic condition (load) by traffic measurement. Our group therefore proposed that the LBPS idea can improve the power-saving performance in the wireless network environment. Moreover, LBPS achieves this goal by setting a target threshold of data accumulation in the buffer for an MSS, and dynamically calculating the next sleep window size. In this way, LBPS can adapt to different traffic loads and still achieve a proper level of power saving.

The basic version of LBPS, *LBPS-Aggr*, is a program in which all the traffic in the network is treated as an aggregate flow in calculating the size of the sleep window. 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:

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

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

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
$\equiv LengthAwkSlpCyl\ (\lambda, Data\_TH) \equiv K^*$
$= Min\{K \mid P_{Acc}(K, Data\_TH) \geq Prob\_TH\}$,

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.

The scheme of LBPS-Split was proposed to improve the performance of LBPS-Aggr in power saving as explained in the following. Considering the case that $K^* = 2$ (the length of the awake-and-sleep cycle is 2 time frames) in LBPS-Aggr, conceptually it implies that 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's 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 the enhanced LBPS-Split protocol.

The best case of *LBPS-Split* in power saving is that each of the split groups is composed of a single MSS, and the final value of K^* is therefore determined by the MSS with the 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 cannot 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. $K^\# = 2^{\lfloor Log_2 K^* \rfloor}$) 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 (1).

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

Schedulability equal or smaller than 1 (*Schedulability* ≤ 1) indicates that a feasible schedule can be found. *Schedulability* > 1 indicates the necessity of group merging. Group merging should try not to reduce too much power-saving efficiency, which means the value of $K^\#$ after group merging should be kept as high 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 affect the value of $K^\#$ is called a non-degraded merge. A degraded merge will take place only if a non-degraded merge cannot be found.

3.2 LBPS-RT

Both *LBPS-Split* and *LBPS-Merge* can assign MSSs to several groups to increase the performance of power saving, however, the idea of LBPS-Split is from one group to many and the idea of LBPS-Merge is to combine similar MSSs as the same group. Therefore the two schemes can also achieve good power-saving efficiency, but they present different features in determining the sleep window size. All groups of MSS in *LBPS-Split* share the same size of sleep window, while *LBPS-Merge* allows different window sizes. Since real-time supporting involves delay bound supporting, which requires the ability to allow different sleep window sizes in power saving, the proposed *LBPS-RT* is thus based on *LBPS-Merge*. There are two aspects to be addressed in *LBPS-RT*:

(1) The threshold of data accumulation in *LBPS-Merge* is set as the amount of a whole time frame data so that the MSS (or a merged group of MSSs) can make the best of its awake time frame. However, in the case of real-time MSS, the sleep window size calculated according to the threshold could lead to the violation of the delay bound. Therefore, the delay bound of the real-time MSS must be taken into consideration in determining the sleep window size in *LBPS-RT*. The revised formula for the sleep window size of a real-time MSS is as follows: (More specifically, K^{rt} is the length of an awake-and-sleep cycle, and the sleep window size is thus $K^{rt}-1$)

$K^{rt}=Min\{K^*,\ Delay\ Bound\ of\ MSS\}$, where K^* is calculated according to the data accumulation method. Once the length of an awake-and-sleep cycle for a real-time MSS is determined, data accumulation (denoted by W*) for the MSS is re-calculated as follows:

$W^* \equiv Min\{W \mid P_{LBPS-RT}(K^{rt}, W) \geq Prob\_TH\}$ (Note that Prob_TH = 0.8 in the simulation)

(2) $P_{LBPS-RT}(K^{rt}, W) \equiv$ Prob[# of packet arrival in K^{rt} Time Frames < W] $= \sum_{i=0}^{W} \frac{e^{-\lambda K^{rt} T}(\lambda K^{rt} T)^i}{i!}$, where T is the length of a time frame, and λ is the estimated load of the real-time MSS. Instead of a whole awake time frame for an MSS (or a merged group of MSSs) in sleep scheduling, the MSS only requires part of the awake time frame, i.e. the expected amount of data is W* which is only a fraction of a time frame data. The equation for schedulability test in *LBPS-Merge* is also required to be revised in the following. 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. $K^\# = 2^{\lfloor Log_2 K^* \rfloor}$) 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 $= \sum_i \frac{1}{K_i^\#}$. The value of Schedulability *that equals or is smaller than 1 (*Schedulability *≤ 1) indicates that a feasible schedule can be found.* Schedulability > *1 indicates the necessity of merging some groups.*

Revised equation of *Schedulability* in *LBPS-RT* is as follows: *Schedulability* $= \sum_i [\frac{1}{K_i^{\#rt}} \times (\frac{W_i^*}{TFdata})] + \sum_j \frac{1}{K_j^\#}$, where i indicates the real-time MSS (or group), j as non-real-time MSS (or group), and *TFdata* as the total capacity of data in a time frame. Based on the above revised equation, it is also worth mentioning that non-real-time traffic is calculated on the traffic load as LBPS-Merge. The algorithm of LBPS-RT is displayed in Figure 1 and an example of LBPS-RT with 8 MSSs is illustrated in Figure 2 and Figure 3.Figure 2 is the process of non-real-time merge and Figure 3 is the process of real-time merge.

---

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. Dividing the traffic into real-time and non-real-time parts, and then sort each part of the groups in the ascendant order of load.
   In non-real-time part, for each group, calculate $K_G^* = LenghAkSlpCyl\{\lambda_G, Data\_TH\}$ and convert K_G^* to the closest and smaller power of $2 = K_G^\# = 2^{\lfloor \log_2 K_G^* \rfloor}$..
   In real-time part, for each group, also calculate $K_G^* = LenghAkSlpCyl\{\lambda_G, Data\_TH\}$, and then use $K_G^{rt} = Min\{K_G^*, Delay\ Bound\}$ as the value of the smaller one, finally, the smaller power of $2 = K_G^{\#rt} = 2^{\lfloor \log_2 K_G^{rt} \rfloor}$ and $W_i^* = bandwidthCalculation(\lambda_i, K_i^{\#rt})$.
3. if $Schedulability = \sum_{i=1}^{x} \frac{1}{K_G^{\#rt}} * (\frac{W_i^*}{Time\ Frame}) + \sum_{j=x+1}^{y} \frac{1}{K_j^*} \leq 1$, goto step 4.
   Else if all non-real-time groups >1{
   Phase 1: Non-degraded merge
   Try to merge the smallest load non-real-time group to another group until a non-degraded merge is found.
   If a non-degraded merge is found, repeat steps 2 &3, else perform Phase 2.
   Phase 2: Degraded merge
   Merge the two non-real-time groups with the smallest loads and repeat steps 2 & 3.
   }Else {
   Phase 1: Non-degraded merge
   Try to merge the smallest load real-time group to another group until a non-degraded merge is found.
   If a non-degraded merge is found, repeat steps 2 &3, else perform Phase 2.
   Phase 2: Degraded merge
   Merge the two real-time groups with the smallest loads and repeat steps 2 & 3.
4. Schedule the groups according to the final set of $K_G^\#$ and $K_G^{\#rt}$.

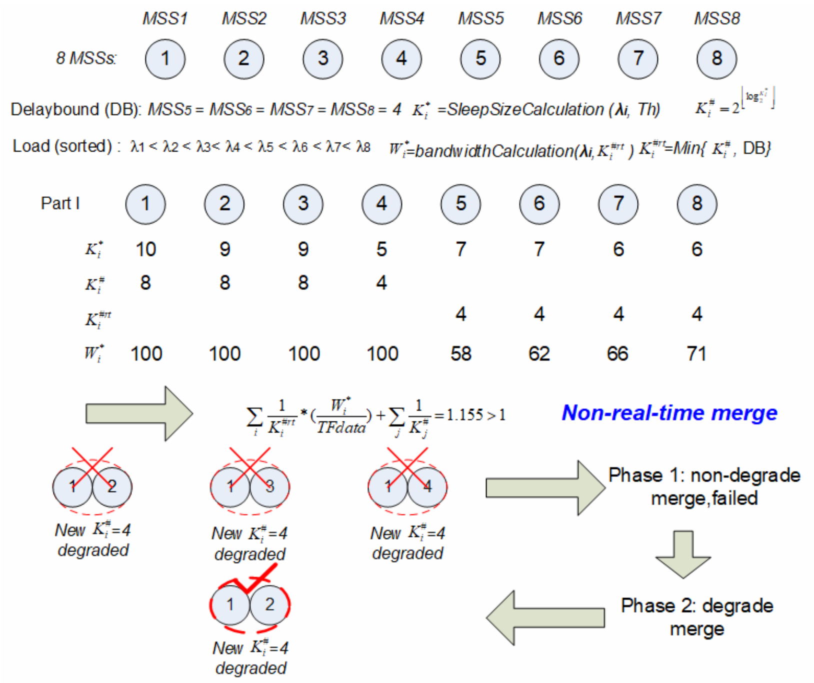

Figure 1. Algorithm of LBPS-RT

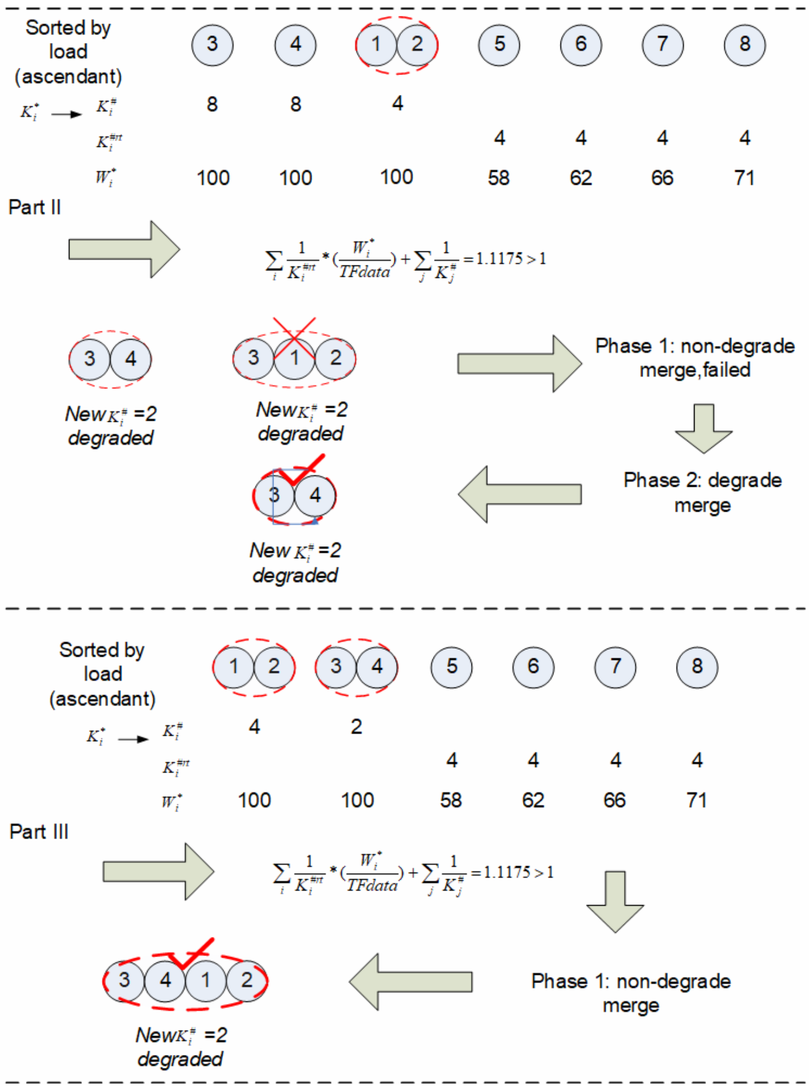

Figure 2. An example of LBPS-RT (Part I-III)

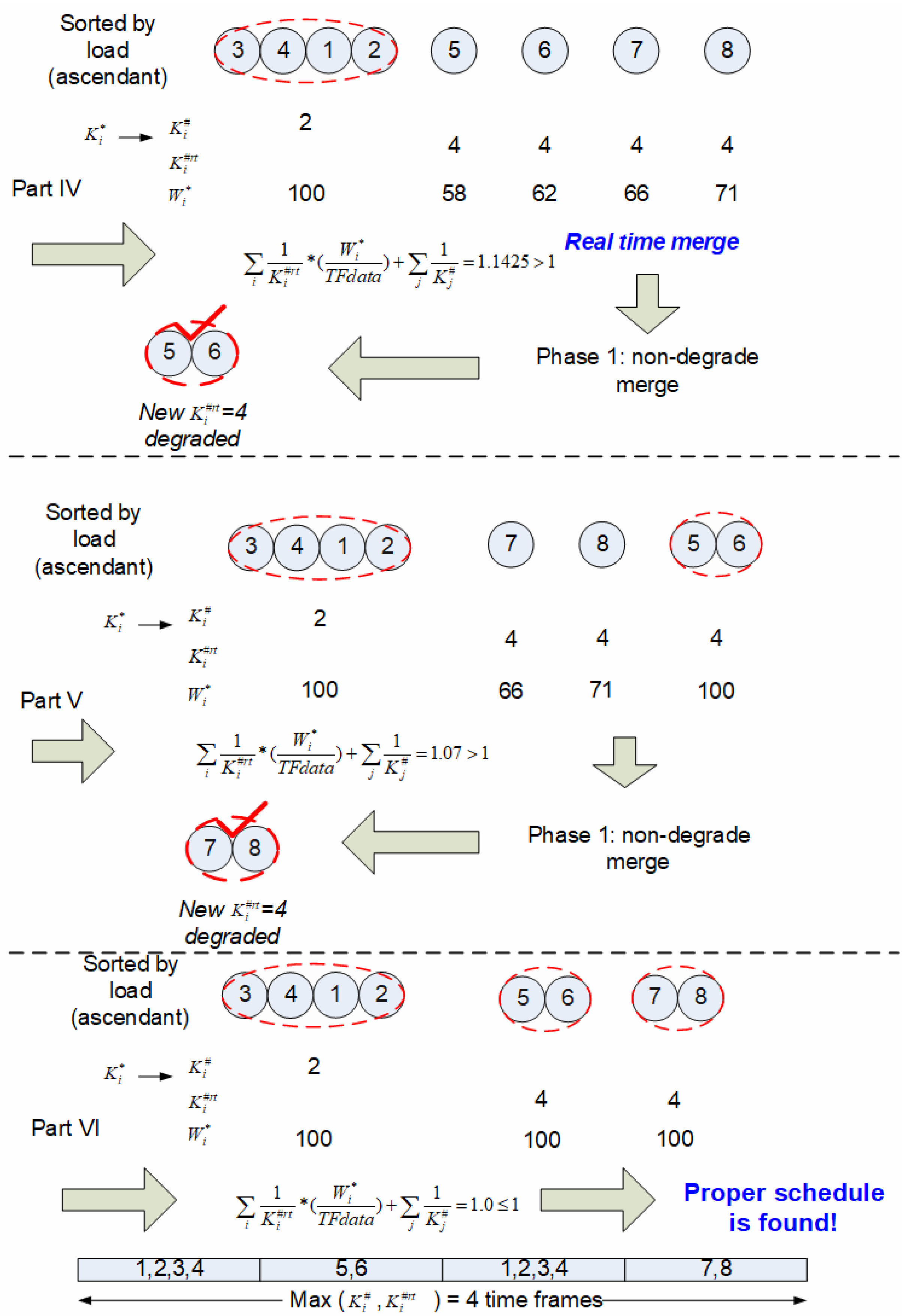

Figure 3. An example of LBPS-RT (Part IV-VI)

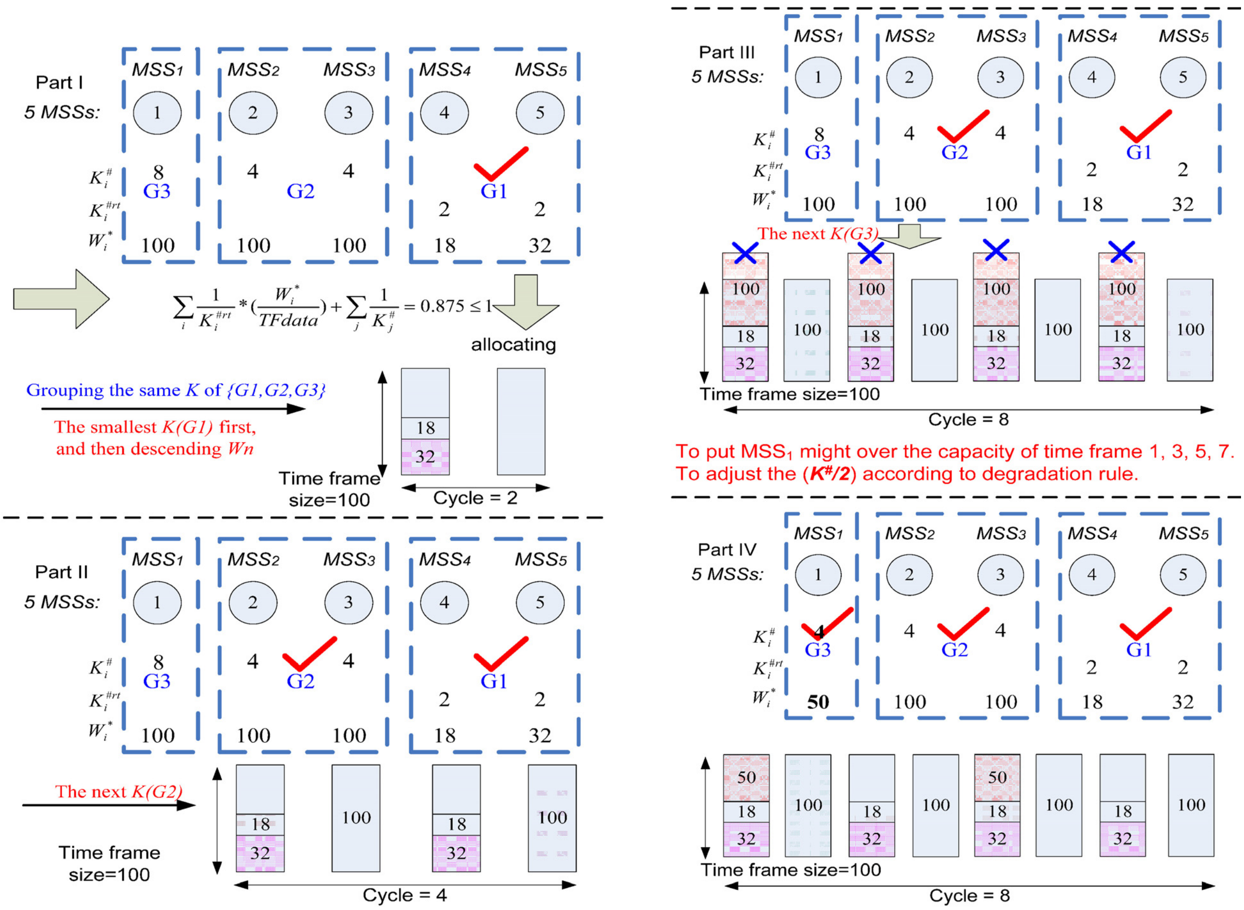

Figure 4. An example of generation of sleep schedule

3.3 GenerAtion of Sleep Schedule

Once the test of Schedulability *is approved,* LBPS-RT *generates a feasible schedule for power saving. So* all MSSs could be scheduled based on their $K^{\#}$ (non-real-time traffic) or $K^{\#rt}$ (real-time traffic) value. Each MSS is scheduled to enter the sleep mode according to the K value. That is, with the same length of the awake-and-sleep cycle, each MSS might have a different starting time for the cycle. We therefore combine some MSSs with the same $K^{\#}$ or $K^{\#rt}$ value as a group, and sort the group sequence depending on their K value. Generally, a smaller K implies either higher load or tighter delay bound. Thus, the group with the smaller K is given a higher priority in sleep scheduling. For example, a case of *LBPS-RT* generation of sleep schedule with 5 MSSs is illustrated in Figure 4.

When the generation schedule is finished, all the MSSs should be put into the time frame based on the designed schedule. There are three different schedule assignment schemes in *LBPS-RT*: *In-order*, *Best-fit*, and *Worst-fit*. To assign each MSS of group in the awake-and-sleep cycle, BS firstly sorts the MSSs based on W^* value in descending order. In the *In-order scheme*, the MSS allocation sequence depends on the smaller group value and the larger W^* value from the start time frame in the whole cycle. In the *Best-fit scheme*, the difference of MSS allocation is the assignment of the individual time frame. The empty time frame will be assigned firstly, and then the smaller remaining space of time frame will be assigned. Finally, the *Worst-fit scheme* is opposite to *Best-fit*, i.e. the bigger remaining space of time frame will be assigned first.

4 Performance Evaluation

4.1 Performance Analysis

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 this case, the power-saving efficiency becomes $\frac{K^*-1}{K^*+N_{ext}}$, where N_{ext} is the number of the 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 (2).

$$\text{PSE} = \sum_{i=0}^{\infty}\left(\frac{K^*-1}{K^*+i}\times Prob[N_{ext}=i]\right) \quad \textbf{(2)}$$

We assume the packet arrival time at the BS is uniformly distributed among the time frames in an 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 a different cycle length is calculated as the following equation (3).

$$\text{AvgDelay} = \sum_{i=0}^{\infty}\left(\frac{K^*+i}{2}\times Prob[N_{ext}=i]\right) \quad \textbf{(3)}$$

4.2 Simulation Environment

Simulation study was conducted to compare the performance of LBPS-RT, standard Type I, and standard Type II, in terms of power-saving efficiency as well as the average access delay. The LBPS-RT that we proposed contains In-order, Best-fit and Worst-fit ordering methods. We would firstly take LBPS-RT-In-order and standard Type I and standard Type II for comparison, and then discuss pros and cons of our three proposed methods. Parameters used in the simulation are listed in Table 1. Note that the threshold of data accumulation *Data_TH* in LBPS-RT 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 for most of the time, i.e. $Prob[N_{ext} = 0] \approx 1$.

4.3 Simulation Results

To compare real-time and non-real-time traffic specifically on power saving, both kinds of traffic are generated from BS.

Table 1. Simulation Parameters

# of BS; # of MSS	1;10, 20, 40, 80 (equal load)
Link bandwidth	20 Mbps
Minislots per time frame	100
Time Frame Size	10 ms
Packet size	128, 256, 512, 1024 Bytes
Prob_TH (LBPS-RT)	0.8
Simulation time	100000 sec.
Delay bound	40 ms (rtPS flow)
Traffic class	Real-time, Non-real-time
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

4.3.1 Effect of Real-time Ratio

Power-saving efficiency (PSE) of LBPS-RT and Type I & II under different input loads with a different real-time ratio are displayed in Figure 5 to Figure 6. There are *30%* and *70%* real-time traffic flows in Figure 5 and Figure 6 respectively. To show the real-time and non-real-time traffic behaviours, those figures have plotted individual results. When the total *load* $\rho \le 0.7$, the real-time traffic of LBPS-RT is paralleled to the Type II in PSE as shown in Figure 5 and Figure 6. This is because the total traffic load is still light, and the BS in both cases can effectively transfer all of the accumulated data to MSS. So the PSE is constant at the same level. But owing to the fact that there's no need for LBPS-RT to wait for idle time to switch to sleep mode, hence the PSE is better than Type II. Besides, if the total *load* $\rho > 0.7$, our proposed LBPS-RT would accumulate data to Data Threshold then back to awake mode, as a result, the time that MSSs stay in the sleep mode is longer than Type II MSSs. To calculate all the effect of real-time and non-real-time PSE, the average PSE is shown in Figure 7 and Figure 8. Consequently, Figure 5 and Figure 8 demonstrate a significantly better power-saving performance of LBPS-RT over standard Type I and II.

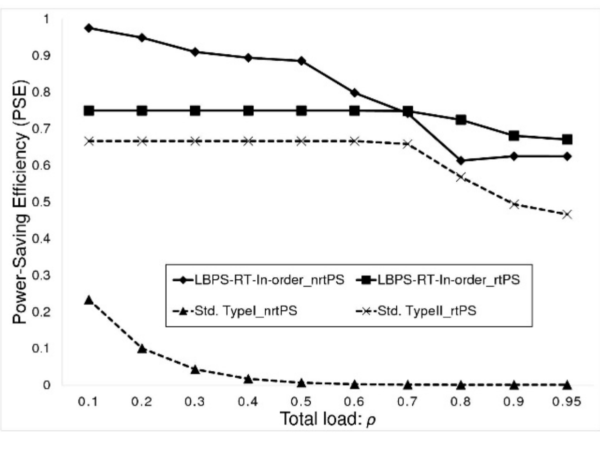

Figure 5. Power-saving efficiency (rtPS ratio=0.3)

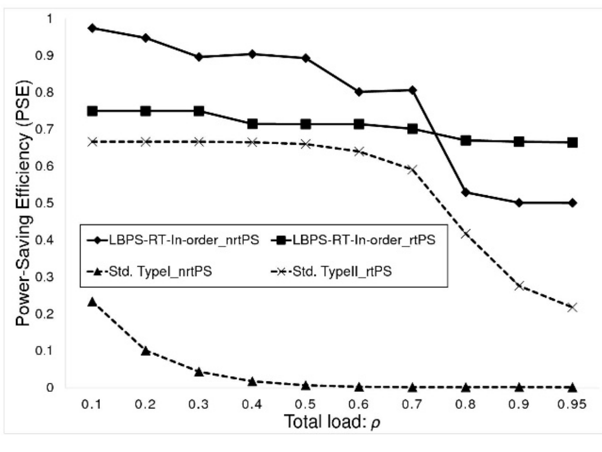

Figure 6. Power-saving efficiency (rtPS ratio=0.7)

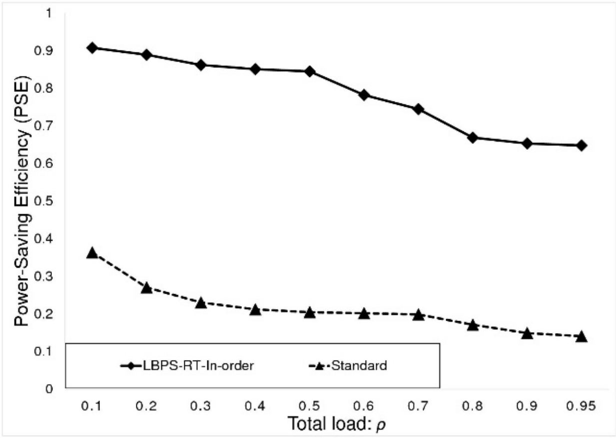

Figure 7. Avg. PSE (rtPS ratio=0.3)

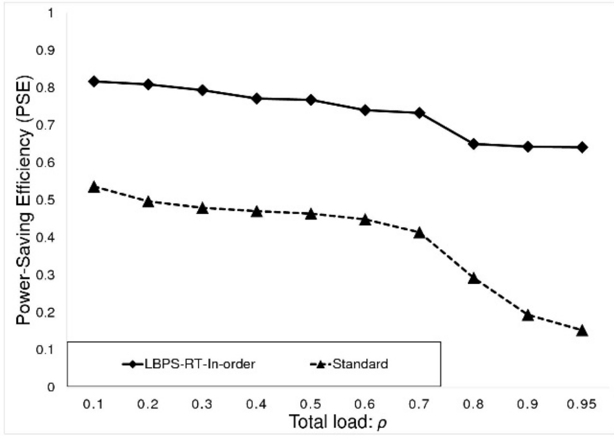

Figure 8. Avg. PSE (rtPS ratio=0.7)

In order to investigate the performance of LBPS-RT for *Real-Time Variable Bit Rate (RT-VBR)* traffic with delay constraint, LBPS-RT can be easily extended to support the requirement of bounded delay, denoted by *DRT-VBR*. LBPS-RT calculates the value of $K^{\#rt}$ as presented in section 3. In Figure 9 and Figure 10, these figures display the delay time of LBPS-RT, Type I and Type II for *RT-VBR* traffic, in which *DRT-VBR = 40 ms* from *Delay bound* of Table 1. From the Figure 5to Figure 10, these figures have demonstrated that a better power-saving efficiency can be achieved by LBPS-RT over Type I & Type II at the cost of slightly more access delay.

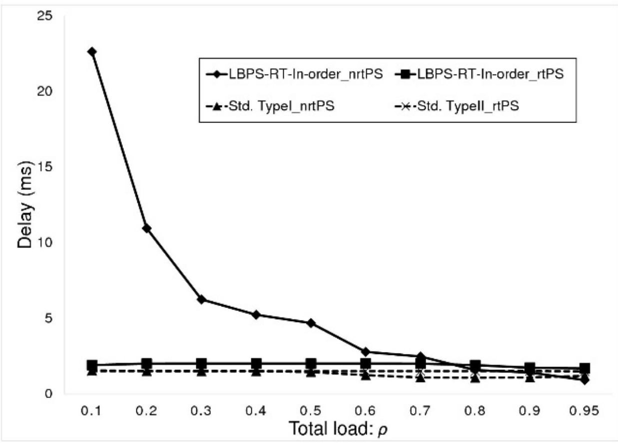

Figure 9. Avg. access delay time (rtPS ratio=0.3)

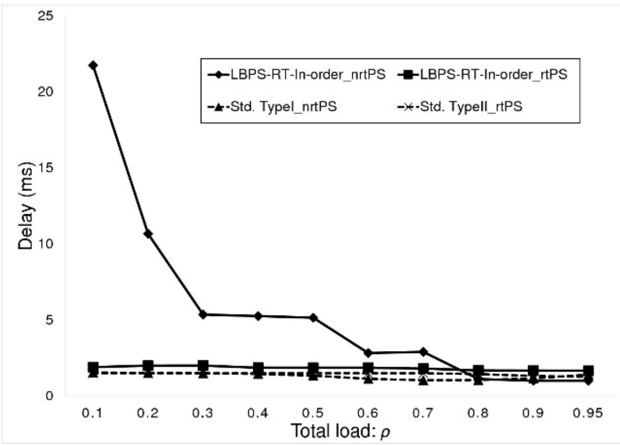

Figure 10. Avg. access delay time (rtPS ratio=0.7)

4.3.2 Effect of # of MSS and Packet Size

The result of power-saving efficiency of LBPS-RT and standard under different numbers of MSS when the real-time traffic flows are 30% is displayed in Figure 11. The figure demonstrates that better PSE is achieved for a larger number of MSS, since a larger number of MSS brings lower load of each MSS, more flexibility in merging groups and thus more gain in PSE can be obtained. Furthermore, the PSE result with the same traffic state under a different packet size is displayed in Figure 12. The figure demonstrates that better PSE is achieved for a larger packet size, since a larger packet size provides higher inter-arrival time MSSs, and thus more gain in PSE can be obtained.

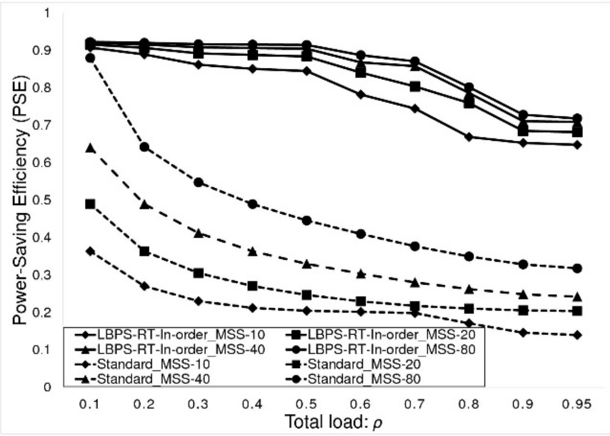

Figure 11. Impact of #MSS on PSE

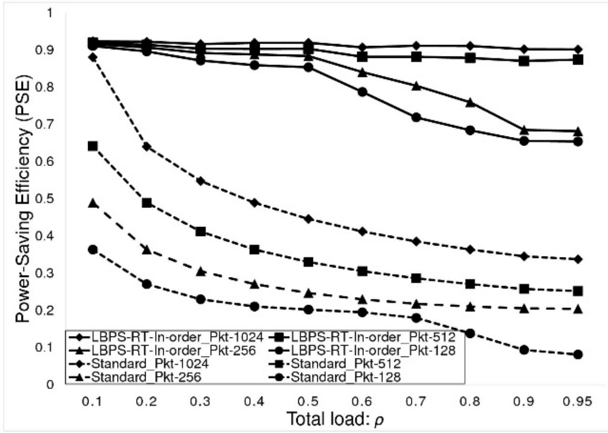

Figure 12. Impact of packet size on PSE

4.3.3 Effect of Schedule Assignment Schemes

Simulation results for comparing three LBPS schedule assignment schemes in terms of PSE in the case of 10 MSSs with equal load are displayed in Figure 13 to Figure 14. Performance comparison of *In-order, Best-fit* and *Worst-fit* yields different results under a different traffic load. As illustrated in Figure 13 and Figure 14, the average PSE in three assignment schemes, the *Best-fit* and *Worst-fit* have better PSE performance under light and middle load, and *In-order* slightly outperforms the others when the traffic in heavy load is above 0.8. The reason for different results under different loads lies in the impact of the key features of the three schemes. *In-order* is lined up according to the order of *K* and *W*. So when MSS needs to be scheduled over two time frames, the *K* value needs to be degraded and the PSE would then be slightly less than *Best-fit* and *Worst-fit*. As the *total load* goes up, the remaining space of time frame also decreases, MSSs need to be merged to be schedulable. The *Best-fit* and *Worst-fit* cannot get advantage when the remaining time frame is quite small. So *In-order* would have better PSE performance when scheduling all groups of MSSs, and it depends on the order of *K* and *W* values. Moreover, the variable load distribution also are displayed in Figure 15 to Figure 16, the 80% load is assigned to 2 MSSs and the other 8 MSSs are assigned 20% load. For the effect of variable load, the PSE of the proposed assignment schemes are almost the same under light and middle load and *In-order* can outperform the others' assignment schemes when the traffic in heavy load is above 0.8. The specificity of variable load distribution would increase the traffic burstiness even when the traffic load is light. As a result, the *In-order* has better performance than the others. Therefore the *In-order* is more flexible and a more useful assignment scheme for our proposed LBPS-RT.

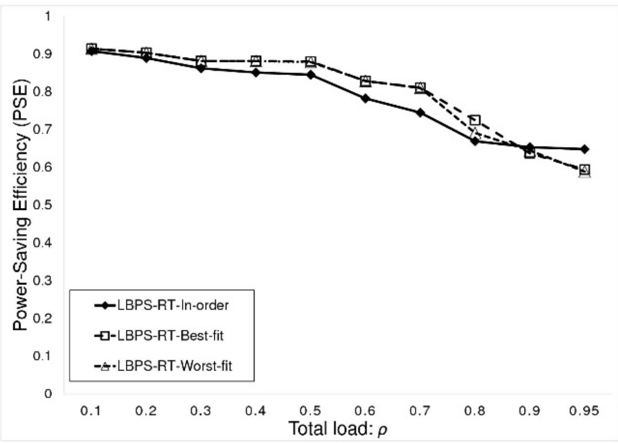

Figure 13. Equal load of PSE (rtPS ratio=0.3)

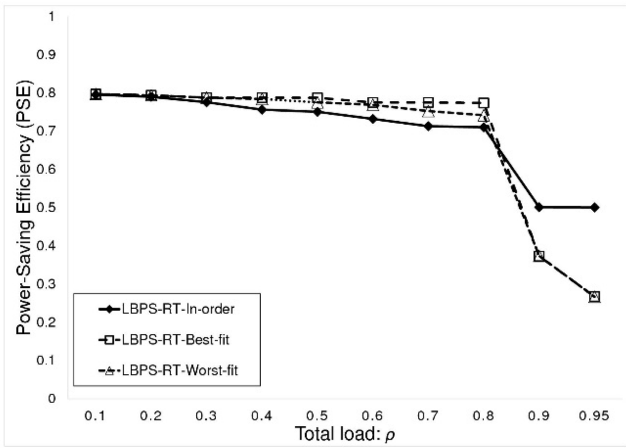

Figure 14. Equal load of PSE (rtPS ratio=0.7)

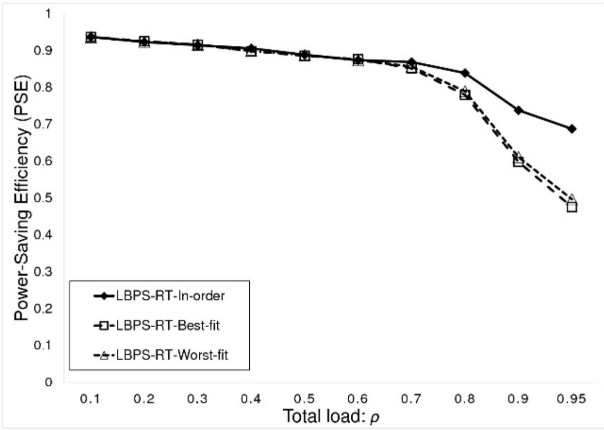

Figure 15. 8:2 load of PSE (rtPS ratio=0.3)

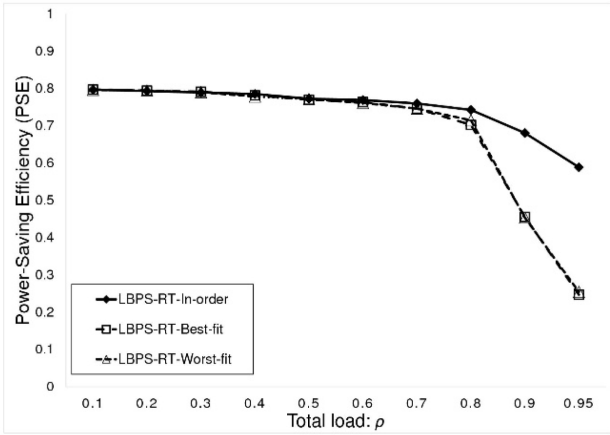

Figure 16. 8:2 load of PSE (rtPS ratio=0.7)

5 Conclusion

In IEEE 802.16e, because of the development progress in the wireless communication technology, the adding mobility functions to SS to make it to MSS has certainly became a very significant issue. Considering energy saving for wireless communication, there are three kinds of power-saving modes to improve the MSS's power usage. However, there is still space for improvement. In our previous work, our focus was on how to accumulate appropriate data at BS so that we can improve power saving for MSS.A basic version of LBPS, LBPS-Aggr, is proposed in our previous paper, in which all of the traffic in the IEEE 802.16e network is treated as an aggregate flow for estimating the sleep window size. The enhanced LBPS schemes, namely LBPS-Merge is also proposed in the paper. Instead of treating all traffic as a single aggregate flow, the 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. However, to support both real-time and non-real-time traffic is necessary, our paper proposed new LBPS-RT can have higher power-saving efficiency and achieve delay requirement of real-time traffic. Moreover, the three assignment schemes are proposed in this paper.

Simulation study has demonstrated that LBPS-RT significantly outperforms Type I & Type II in power saving, and obey the delay requirement of real-time traffic. In addition, LBPS-RT outperforms Type II in power-saving efficiency at the cost of slightly more delay. Impact of the number of MSS and packet size on the performance of power saving have been discussed in this paper. Moreover, three schedule assignment schemes have been also presented in this paper. The Best-fit and Worst-fit assignment schemes may have better performance with light or middle load; however, the In-order scheme is more appropriate for heavy load, especially for the variable load state.

Acknowledgments

This work was supported in part by the National Science Council, Taiwan, R.O.C., under grant no. NSC 99-2221-E-164-006 and Research Project of National Taiwan University of Sport, Taiwan, R.O.C., under grant No. 109DG00104.

References

[1] *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*, IEEE Std. 802.16e-2006, February, 2006.

[2] *IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Broadband Wireless Access Systems*, IEEE Std. 802.16-2009, May, 2009.

[3] B. Kamali, *AeroMACS: An IEEE 802.16 Standard-Based Technology for the Next Generation of Air Transportation Systems*, September, 2018, Doi: 10.1002/9781119281139.

[4] C. H. Hsu, Y. S. Chen, I. K. Fu, P. Cheng, *Power Saving Management for IEEE 802.16m*, IEEE C802.16m-08/567r1, July, 2008.

[5] C. C. Yang, C. H. Fang, J. R. Lin, Adaptive Power Saving Strategy Based on Traffic Load in the IEEE 802.16e Network, *International Conference on Information and Communication Technologies (ICICT 2010)*, Tokyo, Japan, 2010, pp. 1-5.

[6] C. C. Yang, Y. T. Mai, J. Y. Chen, Y. S. Shen, Y. C. Kuo, LBPS: Load-based Power Saving in the IEEE 802.16e Network, *Computers and Electrical Engineering*, Vol. 38, No. 4, pp. 891-905, July, 2012.

[7] A. Nascimento, J. Rodriguez, Cluster Cooperation for Energy Saving in Wireless Personal Networks, *IEEE International Conference on Computer as a Tool (EUROCON)*, Lisbon, Portugal, 2011, pp. 1-4.

[8] E. Hwang, K. J. Kim, J. J. Son, B. D. Choi, The Power-

Saving Mechanism with Periodic Traffic Indications in the IEEE 802.16e/m, *IEEE Transactions on Vehicular Technology*, Vol. 59, No. 1, pp. 319-334, January, 2010.

[9] R. H. Hwang, C. Y. Wang, C. J. Wu, G. N. Chen, A Novel Efficient Power-saving MAC Protocol for Multi-hop MANETs, *International Journal of Communication Systems*, Vol. 26, No. 1, pp. 34-55, January, 2013.

[10] M. Collotta, G. Pau, A Power Management Solution for Bluetooth Low Energy in Smart Homes of Internet of Things, *International Journal of Internet Protocol Technology*, Vol. 9, No. 2-3, pp. 53-61, January, 2016.

[11] H. W. Wang, P. J. Chen, The Performance Optimization of Energy-Efficient Scheduling Algorithm for Cluster-based Wireless Sensor Networks, *Journal of Internet Technology*, Vol. 18, No. 2, pp. 361-370, March, 2017.

[12] M. Collotta, G. Pau A Novel Approach to Improve Power Management in Wireless Local Area Networks, *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 28, No. 1, pp. 27-34, April, 2018.

[13] C. C. Yang, Y. T. Mai, J. Y. Chen, C. M. Yang, Integrated Sleep Scheduling for Multiple Relays and UEs in LTE-TDD, *EURASIP Journal on Wireless Communications and Networking (EURASIP JWCN)*, Vol. 2019, Article No. 226, pp. 1-19, September, 2019.

[14] H. M. Ali, W. Ejaz, D. C. Lee, I. M. Khater, Optimising the Power Using Firework-based Evolutionary Algorithms for Emerging IoT Applications, *IET Networks*, Vol. 8, No. 1, pp. 15-31, January, 2019.

[15] K. Han, S. Choi, Performance Analysis of Sleep Mode Operation in IEEE 802.16e Mobile Broadband Wireless Access Systems, *63rd IEEE Vehicular Technology Conference (VTC 2006-Spring)*, Melbourne, Vic., Australia, Vol. 3, 2006, pp. 1141-1145.

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

[17] J. Jang, K. Han, S. Choi, Adaptive Power Saving Strategies for IEEE 802.16e Mobile Broadband Wireless Access, *Asia-Pacific Conference on Communications (APCC'06)*, Busan, South Korea, 2006, pp. 1-5.

[18] J. Xue, Z. Yuan, An Adaptive Power Saving Strategies based on Cross-layer Design in IEEE 802.16e, *Journal of Networks*, Vol. 5, No. 3, pp. 359-366, March, 2010.

[19] C. H. Hsu, K. T. Feng, C. J. Chang, Statistical Control Approach for Sleep-Mode Operations in IEEE 802.16m Systems, *IEEE Transactions on Vehicular Technology*, Vol. 59, No. 9, pp. 4453-4466, November, 2010.

[20] L. D. Chou, D. C. Li, W. Y. Hong, Improve Energy Efficiency with a Battery Lifetime Aware Mechanism in the IEEE802.16e Broadband Access Networks, *2011 IEEE 14th International Conference on Computational Science and Engineering (CSE)*, Dalian, China, Aug. 2011, pp. 312-316.

[21] Y. Zhang, Y. Xiao, V. C. M. Leung, Energy Management Analysis and Enhancement in IEEE 802.16e WirelessMAN, *IEEE Transactions on Vehicular Technology*, Vol. 58, No. 7, pp. 3738-3752, September, 2009.

[22] S. Jin, W. Yue, X. Zhang, Performance Analysis of the Sleep Mode in WiMAX 2 Networks with Multimedia Application, *Journal of Applied Mathematics*, Vol. 2014, Article ID 976179, January, 2014.

[23] K. Sanghvi, P. K. Jain, A. Lele, D. Das, Adaptive Waiting Time Threshold Estimation Algorithm for Power Saving in Sleep Mode of IEEE 802.16e, *3rd International Conference on Communication Systems Software and Middleware (COMSWARE)*, Bangalore, India, 2008, pp. 334-340.

[24] C. H. Hsu, K. T. Feng, Performance Enhancement and Analysis for IEEE 802.16e/m Sleep Mode Operations with Unsolicited Grant Service/Real-time Variable-rate Connections, *IET Communications*, Vol. 4, No. 9, pp. 1083-1094, June, 2010.

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

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

[27] J. J. Chen, S. L. Wu, S. W. Wang, Power Saving Class Management for Energy Saving in IEEE 802.16e Wireless Networks, *2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware (MDM'09)*, Taipei, Taiwan, 2009, pp. 490-495.

[28] S. T. Sheu, Y. C. Cheng, L. W. Chen, J. S. Wu, J. Chang, Listening Interval Spreading Approach (LISA) for Handling Burst Traffic in IEEE 802.16e Wireless Metropolitan Area Networks, *IEEE 68th Vehicular Technology Conference (VTC 2008-Fall)*, Calgary, BC, Canada, 2008, pp. 1-5.

[29] C. Y. Wu, H. J. Ho, S. L. Lee, L. L. Chen, Fairness Time-Slot Allocation and Scheduling with QoS Guarantees in Multihop WiMAX Mesh Networks, *Mathematical Problems in Engineering*, Vol. 2013, Article ID 293962, December, 2013.

[30] Y. Wu, Y. Le, D. Zhang, An Enhancement of Sleep Mode Operation in IEEE 802.16e Systems, *IEEE 69th Vehicular Technology Conference (VTC 2009-Spring)*, Barcelona, Spain, 2009, pp. 1-6.

[31] S. W. Kwon, D. H. Cho, Enhanced Power Saving through Increasing Unavailability Interval in the IEEE 802.16e Systems, *IEEE Communications Letters*, Vol. 14, No. 1, pp. 24-26, January, 2010.

[32] S. C. Ke, Y. W. Chen, H. A. Fang, An Energy-saving-centric Downlink Scheduling Scheme for WiMAX Networks, *International Journal of Communication Systems*, Vol. 27, No. 11, pp. 2518-2535, November, 2014.

[33] T. C. Chen, J. C. Chen, Y. Y. Chen, Maximizing Unavailability Interval for Energy Saving in IEEE 802.16e Wireless MANs, *IEEE Transactions on Mobile Computing*, Vol. 8, No. 4, pp. 475-487, April, 2009.

[34] C. Cicconetti, L. Lenzini, E. Mingozzi, C. Vallati, Reducing Power Consumption with QoS Constraints in IEEE 802.16e Wireless Networks, *IEEE Transactions on Mobile Computing*, Vol. 9, No. 7, pp. 1008-1021, July, 2010.

[35] J. J. Chen, J. M. Liang, Y. C. Tseng, An Energy Efficient Sleep Scheduling Considering QoS Diversity for IEEE 802.16e Wireless Networks, *2010 IEEE International Conference on Communications (ICC)*, Cape Town, South Africa, 2010, pp. 1-5.

Biographies

Yi-Ting Mai received his Ph.D. degree in computer science from National Chi Nan University, Puli, Taiwan, in 2008. In 2014, Dr. Mai is an Associate Professor in Department of Sport Management, National Taiwan University of Sport. His current research topics include Internet technology in education, MIS in sport, and LTE/LTE-A.

Jeng-Yueng Chen received his Ph.D. degree in computer science from National Chi Nan University, Puli, Taiwan, in 2010. He joined the Department of Information Networking Technology, Hsiuping University of Science and Technology in 2001. His current research topics covers network protocols, network management and 4G/5G.

Min-Hsien Lin received his B.S. degree in Department of Industrial Engineering and Technology Management from Dayeh University, Yuanlin, Taiwan, in 2003 and M.S. degree in computer science from National Chi Nan University, Puli, Taiwan, in 2010. His current research topic includes mobility management in wireless access network.