Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Maximizing power saving with state transition overhead for multiple mobile subscriber stations in WiMAX

Bo LI†[1,2], Sung-kwon PARK[1]

(*[1]Department of Electronic Engineering, Hanyang University, Seoul 133791, Korea*)

(*[2]Guangdong Provincial Key Laboratory of Big Data Analysis and Processing, Sun Yat-Sen University, Guangzhou 510006, China*)

†E-mail: bosign@gmail.com

Received Sept. 25, 2015; Revision accepted Mar. 21, 2016; Crosschecked Sept. 18, 2016

Abstract: In the IEEE 802.16e/m standard, three power saving classes (PSCs) are defined to save the energy of a mobile subscriber station (MSS). However, how to set the parameters of PSCs to maximize the power saving and guarantee the quality of service is not specified in the standard. Thus, many algorithms were proposed to set the PSCs in IEEE 802.16 networks. However, most of the proposed algorithms consider only the power saving for a single MSS. In the algorithms designed for multiple MSSs, the sleep state, which is set for activation of state transition overhead power, is not considered. The PSC setting for real-time connections in multiple MSSs with consideration of the state transition overhead is studied. The problem is non-deterministic polynomial time hard (NP-hard), and a suboptimal algorithm for the problem is proposed. Simulation results demonstrate that the energy saving of the proposed algorithm is higher than that of state-of-the-art algorithms and approaches the optimum limit.

Key words: Power saving class, State transition overhead, IEEE 802.16e/m, Quality of service
http://dx.doi.org/10.1631/FITEE.1500314 **CLC number:** TN929.53

1 Introduction

Worldwide Interoperability for Microwave Access (WiMAX) has been widely used as a last-mile access technology because of its good performance and low cost. However, there are still many issues in WiMAX networks. One of the critical issues is that the mobile subscriber stations (MSSs) in WiMAX networks are battery-powered, which limits their lifetimes. To prolong the lifetimes of MSSs, a power saving mechanism (PSM) and three power saving classes (PSCs) were proposed in the IEEE 802.16e standard (Broadband Wireless Access Working Group, 2006). With the PSM, an MSS is allowed to turn off its receiver circuit to sleep. A PSC defines when an MSS can sleep and when it has to wake up. Type I PSC is recommended for best effort (BE) and non-real-time polling service (nrtPS); Type II PSC is suitable for unsolicited grant service (UGS), real-time polling service (rtPS), and extended real-time polling service (ertPS); Type III PSC is used for multicast services and management purpose.

As shown in Fig. 1, Types I and II PSCs consist of several sleep cycles. A sleep cycle contains a sleep window and a listening window. An MSS will turn off its receiver circuit during sleep windows to conserve energy, while it has to be active in the listening windows. The listening window length in Type I PSC is fixed, while the sleep window length doubles in each sleep cycle from a predefined initial value, until it reaches the maximum sleep window length defined in the IEEE 802.16e standard. This is because non-real-time traffic just comes occasionally and its delay constraint is not stringent. However, a real-time stream is continuous and its packet delay bound is tight. An MSS with a real-time connection has to wake up regularly. Thus, in Type II PSC, both the sleep window length and listening window length are fixed. In Type III PSC, there is only one sleep

ORCID: Bo LI, http://orcid.org/0000-0002-5790-4778

window, and an MSS in sleep mode with Type III PSC will switch to normal mode directly after the single sleep window.

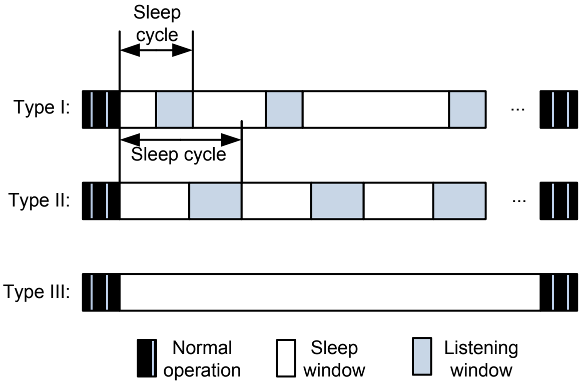

Fig. 1 Power saving classes in IEEE 802.16e

In the literature, many PSC setting algorithms have been proposed (Chen *et al.*, 2009; Tseng *et al.*, 2011; Feng and Li, 2013; Lin and Wang, 2013). These algorithms consider PSC setting only for a single MSS. However, in practice, there is always more than one MSS in the network. Considering the practical scenarios, an algorithm, which is called adaptive bandwidth reservation (ABR) (Wu *et al.*, 2012), was proposed to set Type II PSCs for multiple MSSs. While switching to active state from sleep state, an MSS needs to wake up and synchronize the receiver circuits. This process costs an overhead time, and is nontrivial compared to the sleep cycle length of PSCs. However, this overhead time is ignored in previously proposed PSC setting algorithms. Although the target of the PSC setting algorithms in the literature is to minimize the energy savings of MSSs, the real energy saving is less than expected. To optimize the real energy saving, we must take the state transition overhead into account.

In this paper, we study the problem of maximizing the total energy saving of multiple MSSs with consideration of the transition overhead time. Like the ABR algorithm, we consider the scenario of Type II PSC setting. A suboptimal algorithm is proposed for the PSC setting problem, which is non-deterministic polynomial time hard (NP-hard). We compare the energy saving ratio of the proposed algorithm with that of ABR and the optimal limit through simulations. The simulation results show that the energy saving ratio of our algorithm is about 5% higher than that of ABR and approaches the optimum limit.

2 Preliminary

The power saving performance for the MSS in IEEE 802.16e has been extensively studied in recent years. Xiao (2005) proposed a model to calculate the energy consumption and packet delay in the sleep mode, and evaluated the impact of the minimum sleep window and maximum sleep window on the power saving performance. Later on, Park and Hwang (2009) constructed a semi-Markov-chain-based model to analyze the power saving performance in IEEE 802.16e. They also presented how to set the PSC parameters for the restricted packet delay and energy consumption of the MSSs. The packets for MSSs in the sleep mode have to wait for transmission during the listening windows in a queue in the base station (BS). Thus, some researchers also studied the power saving of IEEE 802.16e using queuing theory (Seo *et al.*, 2004; Zhu *et al.*, 2009).

An MSS may have multiple connections, and each connection can be assigned a PSC as defined in IEEE 802.16e. An MSS with multiple connections can turn off its receiver to save energy only when all the connections are in the sleep windows. The time intervals when all connections of an MSS are in the sleep windows are defined as the unavailability intervals. To maximize the energy saving of an MSS, its unavailability interval should be maximized. Also, the PSC parameters, such as the sleep cycle length and listening window lengths, should be set appropriately to guarantee the packet delay bound of each connection in an MSS. In the literature, many PSC setting algorithms have been proposed to maximize the unavailability intervals with a quality of service (QoS) guarantee.

Most of the proposed algorithms consider only how to arrange different PSCs for different connections in a single MSS. Feng *et al.* (2013) and Lin and Wang. (2013) proposed to adjust both the minimum and maximum sleep window lengths of Type I PSC dynamically, according to the traffic load. In their works, only Type I PSC is considered. For Type II PSC, a maximizing unavailability interval (MUI) algorithm was proposed by Chen and Chen (2009).

The algorithm decides the start frame of each connection in an MSS to maximize the unavailability interval based on the Chinese remainder theorem. The authors also extended the MUI algorithm to set both Types I and II PSCs for an MSS with diverse traffics. There are also other algorithms for the mixture of Types I and II PSCs. For example, Tseng *et al.* (2011) proposed a fold-and-demultiplex method to assign Types I and II PSCs for an MSS. In the algorithm, each connection is assigned with a PSC at first, and then the PSCs are folded together into one series to calculate the total bandwidth requirement. Finally, the series is demultiplexed into multiple PSCs. Besides, a semi-Markov decision process based algorithm is proposed to decide the Types I and II PSCs for connections with different QoS requirements (Wu *et al.*, 2012).

The IEEE 802.16m standard (Broadband Wireless Access Working Group, 2011) is the advanced version of IEEE 802.16e. To improve the performance of IEEE 802.16e, several proposals were adopted in IEEE 802.16m (Baek *et al.*, 2009; Kalle *et al.*, 2009; Hwang *et al.*, 2010; Kim and Mohanty, 2010). One of the significant changes in IEEE 802.16m is that both the sleep cycle length and listening window length can be updated. Another feature is that an MSS with multiple connections is associated with only one PSC in the sleep mode. Consequently, there is no need to consider how to arrange different PSCs in an MSS to maximize the unavailability interval, and the PSC management is simplified. The performance of IEEE 802.16m has been analyzed in many works (Chen *et al.*, 2010; Jin *et al.*, 2010; Park *et al.*, 2010).

The power saving performance of IEEE 802.16m has also attracted the interest of many researchers. Hwang *et al.* (2010) analyzed the benefits of the explicit traffic indication for the MSS. Numerical results show that, with explicit traffic indication, 20%–50% reduction of the energy can be achieved for Type I PSC. Chen *et al.* (2010) constructed a concise model to analyze the sleep mode operation of IEEE 802.16m and compared the power saving efficiency and the mean waiting time of IEEE 802.16m and 802.16e. The performances of IEEE 802.16m with both non-real-time and real-time services were also studied (Jin *et al.*, 2010). Park *et al.* (2010) evaluated the power saving ratio in IEEE 802.16m and demonstrated the results for HTTP and FTP traffic. Jin *et al.* (2011) proposed an adaptive sleep mode management scheme that can adjust an MSS's sleep cycle length and listening window length based on the estimation of the traffic load.

3 System model and problem statement

3.1 System model

We consider an IEEE 802.16e/m network with N MSSs, and each MSS has one or multiple real-time connections. We consider only Type II PSC setting for real-time connections. For each MSS that requests entry into the sleep mode, the serving BS sets only one Type II PSC for its real-time connections. That is, all the real-time connections of an MSS receive data during the same listening window in a sleep cycle. The aggregated bit rate of the real-time connections in MSS M_i is denoted as r_i. We assume that the total available bandwidth for MSSs in sleep mode is B. The parameters of the PSC for an MSS M_i are the start frame number S_i, sleep cycle length T_i, and listening window length L_i. The sleep cycle length and window length are measured by the orthogonal frequency division multiplexing (OFDM) frame length denoted by T_f. The notations are summarized in Table 1.

Table 1 Summary of the key notations

Symbol	Description
b_i	Reserved bandwidth for MSS M_i (Hz)
r_i	Aggregated bit rate of MSS M_i (bit/s)
S_i	Start frame number of Type II PSC for MSS M_i
T_i	Sleep cycle length of Type II PSC for MSS M_i (frame)
L_i	Listening window length MSS M_i (frame)
B	Bandwidth allocated to real-time services (Hz)
D_i	Minimum packet delay bound of MSS M_i (s)
T_0	Time needed for MSS state transition (s)
γ	Average energy saving ratio
T_f	Length of a frame

MSS: mobile subscriber station; PSC: power saving class

According to these PSC parameters, an MSS M_i will start to sleep from frame S_i. Then, in every T_i frames, M_i starts sleeping for T_i–L_i frames and then wakes up to receive data for L_i frames. For example, assume for MSS M_i, S_1=3, T_1=5, and L_1=2. Then, M_1 will start the sleep mode from frame 3, and in every five frames it first starts sleeping for three frames and then receives data for two frames (Fig. 2). The BS buffers the packets for an MSS in its sleep windows and transmits the buffered packets during its listening windows.

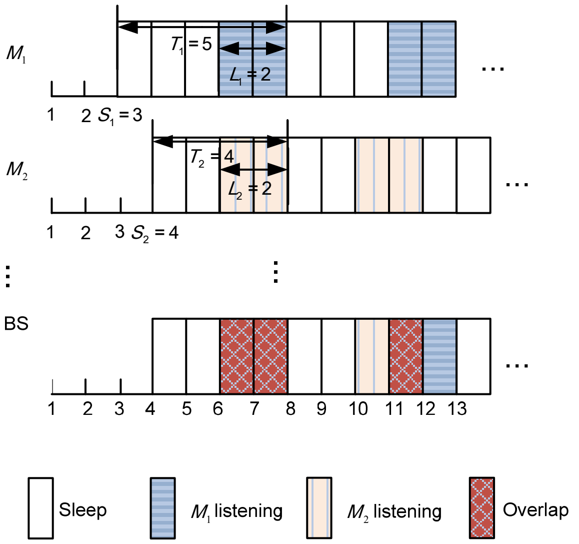

Fig. 2 Example of Type II PSC setting for MSSs
PSC: power saving class; MSS: mobile subscriber station

3.2 Problem statement

Consider that the BS transmits real-time data to M MSSs. To guarantee QoS, the BS needs to reserve a bandwidth for each MSS during its listening windows. The total real-time traffic of an MSS M_i during a sleep cycle length T_i is $T_i r_i$, where r_i is the aggregated bit rate of all real-time connections in MSS M_i. The size of the data that the BS can transmit to M_i during a sleep cycle length is $L_i b_i$. We have to make sure that in each sleep cycle all the packets of M_i can be sent out within the listening window. Thus, we have the constraint $L_i b_i \geq T_i r_i$. Besides, there is a packet delay bound for each connection of M_i. Because all the packets for M_i arriving during a sleep cycle can be sent out in the listening window within this sleep cycle, any packet for M_i will not be delayed longer than the sleep cycle length T_i. We constrain T_i to be shorter than D_i to guarantee the packet delay bounds of all the connections in M_i.

For a single MSS, we need only to consider the two constraints above. However, with multiple MSSs, another problem is that the listening windows of different MSSs may overlap with each other. Suppose for another MSS M_2, $S_2=4$, $T_2=4$, and $L_2=2$. As shown in Fig. 2, among the first 13 frames, the listening windows of both M_1 and M_2 overlap at frames 6, 7, and 11. During these overlapped frames, the BS has to transmit data to M_1 and M_2. Thus, the needed bandwidth is higher in the overlapped frames than others. We have to reserve enough bandwidth to transmit the data of both M_1 and M_2 at the same time. As a result, the bandwidth is wasted during nonoverlapping of frames. To make full use of the available bandwidth for the real-time service, we should avoid the overlap between listening windows of different MSSs as much as possible.

When an MSS wakes up from the sleep state, it spends time T_0 on synchronizing the circuitry before it can receive data. The energy saving time of M_i during a sleep cycle is $T_i - T_i - T_0$, and the average energy saving ratio over the N MSSs can be calculated as

$$\gamma = \sum_{i=1}^{N} (T_i - L_i - T_0) / (T_i N). \qquad (1)$$

Our goal is to maximize γ. In short, the problem is to maximize γ subject to the following constraints: (1) $L_i b_i \geq T_i r_i$ for any $1 \leq i \leq N$; (2) the listening windows of different MSSs are not overlapped with each other; (3) $T_i \leq D_i$ for any $1 \leq i \leq N$.

According to Eq. (1), maximizing γ is to maximize T_i and minimize L_i. To minimize L_i we need only to consider constraint (1). The intractable problem is how to set each T_i for any $1 \leq i \leq N$. According to constraint (3), to maximize each T_i, we should set $T_i = D_i$. However, simply setting each T_i to D_i may result in listening window conflict among different MSSs, which violates constraint (2). Actually, we cannot find the optimal algorithm for this problem. Even when the state transition overhead T_0 is ignored, the problem has still proved NP-hard (Tseng *et al.*, 2011). Thus, we will propose only a suboptimal algorithm in the next section.

4 Factor base algorithm

As mentioned in the previous section, the problem at hand is NP-hard. It is almost impossible to find the optimal algorithm for the problem. In this section, we will propose a suboptimal algorithm called the factor base (FB) algorithm, to solve this problem.

First, we explain how to set the listening window length L_i for each MSS M_i ($1 \leq i \leq N$). Each L_i should be set as small as possible. According to the constraint $L_i b_i \geq T_i r_i$, to minimize L_i, we need to maximize b_i for a given T_i and r_i. We allocate all the available bandwidths to each MSS during its listening windows; i.e., we set $b_i=B$ for any $1\leq i\leq N$. Consequently, each L_i can be calculated as

$$L_i=\lceil r_i T_i / B \rceil, \qquad (2)$$

where $\lceil \ \rceil$ is the ceiling operator. We use the ceiling operator because L_i cannot be less than $r_i T_i/B$, and L_i must be an integer since it is the number of frames.

Next, we explain how to set the sleep cycle length T_i for each MSS M_i ($1\leq i\leq N$). Recall that, to sufficiently use the system bandwidth, the listening windows of different MSSs cannot be overlapping with each other. To achieve this, the existing ABR algorithm set the sleep cycle lengths of MSSs to power of 2 times the shortest sleep cycle length $T_{\min}$. Assume we set $T_2=T_1$, $T_3=2T_1$, and $T_3=4T_1$, the listening windows of the MSSs are synchronized, because any longer sleep cycle is a multiple of any shorter one (Fig. 3). As a result, the listening windows of different MSSs never overlap with each other.

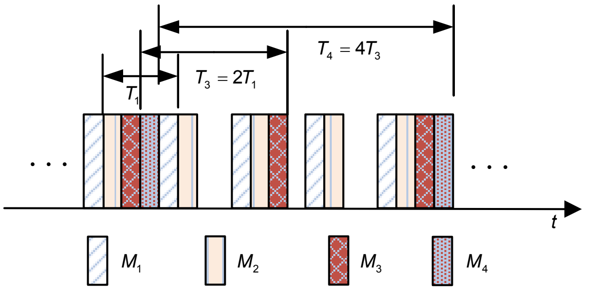

Fig. 3 Sleep cycle lengths of mobile subscriber stations set by the adaptive bandwidth reservation algorithm to the power of 2 times the shortest sleep cycle length

Although the listening window conflict between MSSs is avoided in ABR, the sleep cycle lengths will be shortened. For example, assume there are three MSSs: M_1, M_2, and M_3. Suppose the minimum packet delay bounds of these MSSs are $D_1=6.3T_\mathrm{f}$, $D_2=15.5T_\mathrm{f}$, and $D_3=30.7T_\mathrm{f}$, respectively, where T_f is the length of a frame. Because we have the constraint $T_i\leq D_i$, and each T_i must be an integer, the largest valid values for the sleep cycle lengths are $T_1=6T_\mathrm{f}$, $T_2=15T_\mathrm{f}$, and $T_3=30T_\mathrm{f}$, respectively. However, in the ABR algorithm, T_2 and T_3 must be a power of two times of $T_{\min}=6T_\mathrm{f}$. Thus, we can only reduce T_2 and T_3 to $12T_\mathrm{f}$ and 24 T_f, respectively. As mentioned before, each sleep cycle length should be set as long as possible to achieve a larger sleep cycle length and avoid listening window conflict as well. The core idea of our algorithm is to set each sleep cycle length to a multiple of a factor of the shortest sleep cycle length $T_{\min}$. Take the same example above, where $T_{\min}=T_1=6T_\mathrm{f}$. The factor of $T_{\min}$ is 3, and we set T_2 and T_3 as multiples of 3, i.e., $T_2=3\times5T_\mathrm{f}=15T_\mathrm{f}$, and $T_3=3\times10T_\mathrm{f}=30T_\mathrm{f}$. As we can see, the values of T_2 and T_3 in our algorithm are greater than their counterparts in ABR. The reason is that the increment of the feasible sleep cycle values in our algorithm is smaller than that in ABR and, we can find a sleep cycle length closer to the maximum allowed value than ABR. With larger sleep cycle lengths, we can avoid listening window conflict as well. As shown in Fig. 4, we divide the frames into several regions, each of which contains three frames. Assume $L_1=L_2=L_3=1$, and we set $S_1=1$, $S_2=2$, and $S_3=3$. Then M_1 always occupies the first frame in a region because its sleep cycle is a multiple of $3T_\mathrm{f}$. For the same reason, M_2 and M_3 always occupy the second and the third frame in a region, respectively. In other words, the MSS occupies different frames in any region; i.e., the listening window conflict is avoided.

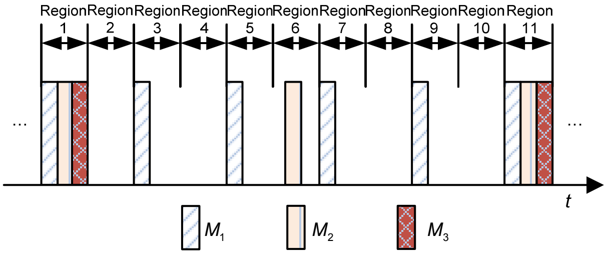

Fig. 4 Listening window conflict avoidance by allocating different frames to different mobile subscriber stations in region 1

The pseudocode of our algorithm is shown in Algorithm 1. The input parameters of the algorithm are the aggregated bit rate of connections r_i, and minimum packet delay bound D_i of each MSS M_i. In the first step, we sort the MSSs according to D_i in ascending order. After sorting, we set its sleep cycle length $T_1=\lfloor D_1\rfloor$ at line 2 in the algorithm for the first MSS M_1, where $\lfloor \ \rfloor$ is the floor operator. Because $T_1=\lfloor D_1\rfloor$, T_1 is an integer and not longer than D_1, which satisfies the packet delay requirement. Since D_1 is the shortest minimum delay bound, T_1 equals the smallest sleep cycle length $T_{\min}$, assuming k is a factor of T_1. As shown at line 5 of the algorithm, for each $1<i\leq N$, we set $T_i=\lfloor D_1/k\rfloor k$. Consequently, the packet delay constraint is satisfied because $T_i\leq(D_1/k)k=D_i$. Also, T_i is a multiple of k because $\lfloor D_1/k\rfloor$ is an integer. When T_i is set, we can calculate the listening window length L_i according to Eq. (2), as shown in line 6 in the algorithm.

Algorithm 1 Factor base algorithm

```
Input: Each $r_i$ and $D_i$, $i$=1, 2, …, $N$
Output: Each $S_i$, $T_i$, and $L_i$, $i$=1, 2, …, $N$
 1: sort MSSs according to $D_i$
 2: $T_1=\lfloor D_1\rfloor$
 3: set $k$ to a factor of $T_1$
 4: for $i$=2 to $N$ do
 5:     $T_1=\lfloor D_1/k\rfloor k$
 6:     $L_i=\lceil \Sigma_j r_{i,j}T_i/B\rceil$
 7:     $\boldsymbol{S}=\{0, 1, \ldots, T_i-1\}$
 8:     for $j$=1 to $i-1$ do
 9:         for $z$=0 to $\lceil T_i/G_{i,j}\rceil-1$ do
10:             $s=zG_{i,j}+S_j$
11:             delete $s+1-L_i$ to $s+L_j-1$ from S
12:         end for
13:     end for
14:     set $S_i$ to the first remaining number
15: end for
```

After deciding T_i and L_i, the next step is to find the start frame S_i for MSS M_i. The first sleep cycle of M_i is from frame 0 to frame T_i-1; thus, we need to pick a number from $S=\{0,1, \ldots, T_i-1\}$ for S_i. For an MSS M_j, denote the greatest common divisor of T_i and T_j as $G_{i,j}$. The frames in a listening window of M_j can be expressed as $xG_{i,j}+S_j+l_j$, where $l_j=0, 1,\ldots, L_j-1$, and x is an integer. The difference between the two expressions is

$$(y-x)G_{i,j}+(S_i+l_i-S_j-l_j), \tag{3}$$

which cannot be let equal to 0; otherwise, the listening windows of M_i and M_j will contain the same frame. The first term of expression (3) is always an integer times $G_{i,j}$. As long as the second term of expression (3) is not an integer times $G_{i,j}$, expression (3) will never be 0. Thus, we have to make sure $S_i\neq zG_{i,j}+S_j+l_j-l_i$, where z is an integer, $l_i=0, 1,\ldots, L_i-1$, and $l_j=0, 1, \ldots, L_j-1$. The range of l_j-l_i is $[1-L_i, L_j-1]$. To avoid listening window conflict between M_i and any M_j ($j<i$), as shown in lines 7–13 in the algorithm, we delete $zG_{i,j}+S_j+1-L_i$ to $zG_{i,j}+S_j+L_j-1$ from the set $S=\{0, 1, \ldots, T_i-1\}$ for any $j<i$ and $z<\lceil T_i/G_{i,j}\rceil$. The remaining numbers in S are feasible for S_i, and we set S_i to the first remaining number, as shown at line 14 in the algorithm. If there is no remaining number in S, we cannot let MSS M_i enter the sleep mode.

The number of MSSs that can enter the sleep mode to save energy is influenced by k, which is the factor of the smallest sleep cycle length in the algorithm. Any T_i or T_j is a multiple of k, and $G_{i,j}$ is the greatest common divisor of T_i and T_j. Thus, $G_{i,j}$ is also a multiple of k. The larger the value of k, the larger the $G_{i,j}$. As shown in lines 9–12 in the algorithm, in every $G_{i,j}$ numbers, L_j-1+L_i numbers are deleted from S. The larger the $G_{i,j}$, the fewer the numbers deleted from S. As a result, it is more probable to find a valid S_i. Thus, the greater the k, the more the MSSs that can enter the sleep mode.

The factor k also has an impact on the energy saving of MSSs. To understand it easily, we show a simple example. Assume the smallest sleep cycle length $T_{\min}$ is $6T_{\rm f}$. The factors of $T_{\min}$ are 2 and 3. If we set k to 2, then the possible values for the sleep cycle length are $6T_{\rm f}$, $8T_{\rm f}$, …, with an increment of $2T_{\rm f}$. If we set k to 3, then the possible sleep cycle lengths are $6T_{\rm f}$, $9T_{\rm f}$, …. Without loss of generality, we assume that the minimum packet delay bound D_i is uniformly distributed between $6T_{\rm f}$ and $12T_{\rm f}$. In case of $k=3$, when $D_i\in[6T_{\rm f}, 9T_{\rm f})$, T_i will be set to $6T_{\rm f}$ because T_i must be less than D_i, and $6T_{\rm f}$ is the largest feasible sleep cycle length. For the same reason, when $D_i\in[9T_{\rm f}, 12T_{\rm f})$, T_i will be set to $9T_{\rm f}$. The mean of the sleep cycle length is $(6+9)/2=7.5T_{\rm f}$. In case of $k=2$, when $D_i\in[6T_{\rm f}, 8T_{\rm f})$, T_i will be set to $6T_{\rm f}$; when $D_i\in[8T_{\rm f}, 10T_{\rm f})$, T_i will be set to $8T_{\rm f}$; when $D_i\in[10T_{\rm f}, 12T_{\rm f})$, T_i will be set to $10T_{\rm f}$. The mean of the sleep cycle length is $(6+8+10)/3=8T_{\rm f}$, which is larger than the value when $k=3$. Generally, assume there are two factors $k_1<k_2$. The increment between two adjacent multiples of k_2 is larger than that of k_1. Thus, within a certain region, there are more multiples of k_1 than of k_2. If we set k to k_1, there will be more candidate values for T_i. As a result, it is more likely to find a large feasible sleep cycle length. Because longer sleep cycles bring less state transition overhead time, setting k to a smaller factor will achieve a higher energy saving ratio.

From the analysis above, we can see that the factor k controls the tradeoff between the energy saving ratio and the number of energy saving MSSs. We will analyze this tradeoff further in Section 5 through simulation.

5 Performance evaluation

In this section, we first run a simulation to see the impact of parameter k, which is a factor of the shortest sleep cycle length in the proposed algorithm. Then, the average energy saving ratio γ of the proposed algorithm is compared with the optimal energy saving ratio $\gamma_{\rm lim}$ and those of state-of-the-art algorithms through simulations. In these simulations, we set different packet delay bounds, different numbers of connections in each MSS, different numbers of MSSs, and different state transition overhead time lengths to show their impacts. The average energy saving ratio γ evaluated in the simulations is defined in Eq. (1). For $\gamma_{\rm lim}$, we cannot evaluate its real value since the optimal algorithm is NP-complete. Instead, we ignore the listening window conflict, and simply set each T_i to D_i to calculate $\gamma_{\rm lim}$. Actually, the resulting $\gamma_{\rm lim}$ is higher than the real optimal energy saving ratio. The simulations were implemented using MATLAB. In the simulations, a single cell with one serving BS and varying numbers of MSSs was considered. Each simulation was repeated 300 times, and we will show the mean values of the results. The fundamental parameters of each simulation are summarized in Table 2.

Table 2 Fundamental simulation parameters

Parameter	Value
Simulation time	30 min
Frame length	5 ms
Time needed for MSS state transition	5 ms
Number of time slots of a frame	256
Number of sub-channels	30
OFDMA PHY mode	PUSC
Packet size	100 bytes

OFDMA PHY: orthogonal frequency division multiplexing access physical; PUSC: partial usage subchannelization

5.1 Impact of factor *k*

As mentioned in Section 4, factor k controls the tradeoff between the energy saving ratio and the number of energy saving MSSs. First, we ran a simulation to show the impact of factor k. In the simulation, we set up 18 MSSs. The bandwidth of real-time services, B, was set to 20 Mb/s, and each MSS had two connections with their bit rates randomly selected from 0.3–0.8 Mb/s. The packet delay bound of each connection was randomly selected from {100, 150, 200, 250, 300} ms. Thus, the minimum sleep cycle length T_1 was 100 ms, which is equal to $20T_{\rm f}$. We set k to 2, 5, and 10, respectively, and repeated the simulation 300 times with each value of k. The simulation results are shown in Table 3. As k increases, the number of supportable energy saving MSSs increases, and the energy saving ratio decreases. As we can see, the average number of supportable energy saving MSSs increases very slowly. It increases only by 0.02 and 0.07 when k changes from 2 to 5 and 5 to 10, respectively. If there are not too many MSSs in the system, a small k is preferred.

Table 3 Average number of supportable energy saving mobile subscriber stations and average energy saving ratios with different *k*'s

k	Number of supportable MSSs	Average energy saving ratio
2	17.58	0.7218
5	17.60	0.7092
10	17.67	0.6852

MSSs: mobile subscriber stations

5.2 Impact of the packet delay bound

In this section, we show the results of a simulation to see the impact of the packet delay bound on energy saving. In this simulation, we used five sets of packet delay bounds (Table 4). We pick one of these five sets to test each time. The packet delay bounds of the connections in each MSS are randomly selected from the tested set. We used the proposed FB algorithm and the state-of-the-art algorithms including ABR (Wu *et al.*, 2012), WAKSLP DECISION (Wong *et al.*, 2010), user counting window (UCW) (Kao *et al.*, 2012), and frame aggregation based power-saving scheduling (FAPS) (Liu *et al.*, 2014) to set the Type II PSCs for MSSs. The average energy saving ratios of these two algorithms were calculated to compare with the optimal limit $\gamma_{\rm lim}$. In this simulation, the factor k of the proposed FB algorithm was set to 2. The total available bandwidth for MSSs in sleep mode was 20 Mb/s. The number of MSSs was 10, and each MSS had two connections with their bit rates randomly selected from 0.3–0.8 Mb/s.

The simulation results are shown in Fig. 5. As we can see, the energy saving ratio increases with the increase of the packet delay bound. The energy saving ratio of the proposed FB algorithm is higher than those of state-of-the-art algorithms. Also, the performance of FB approaches γ_{lim} as the packet delay bound increases. When the first set of packet delay bounds is used where the minimum packet delay bound is 50 ms, the energy saving ratio of FB is only about 1% less than γ_{lim}. When the third packet delay bounds are used where the minimum packet delay is 150 ms, there is almost no difference between the energy saving ratio of the proposed FB algorithm and the optimal limit γ_{lim}.

Table 4 Sets of packet delay bounds used in the simulation

Set	Packet delay bounds (ms)
1	50, 100, 150, 200, 250
2	100, 150, 200, 250, 300
3	150, 200, 250, 300, 350
4	200, 250, 300, 350, 400
5	250, 300, 350, 400, 450

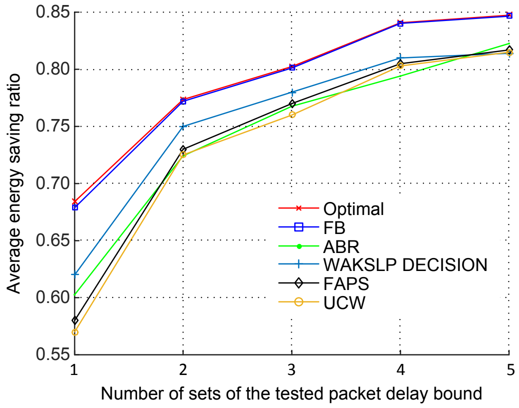

Fig. 5 Average energy saving ratio with different sets of packet delay bounds

FB: factor base; ABR: adaptive bandwidth; FAPS: frame aggregation based power-saving scheduling; UCW: user counting window; MSS: mobile subscriber stations

5.3 Impact of the number of connections per mobile subscriber station

Next, we ran a simulation to see the impact of the number of connections in each MSS. We set the bandwidth B to 30 Mb/s and ran the simulation with different numbers of connections per MSS. In this simulation, the packet delay bounds of MSSs were randomly selected from the values of set 1 in Table 4. Other parameters were the same as in the previous simulation. We changed the number of connections per MSS from 1 to 5. The average energy saving ratio with different numbers of connections per MSS is shown in Fig. 6.

As we can see, when there is just one connection in an MSS, the average energy saving ratio of FB is about 88%. It is only about 1% less than γ_{lim}. As the number of connections per MSS increases, the energy saving ratio decreases. Even when there are five connections in an MSS, FB can still achieve an about 81% average energy saving ratio, and the energy saving ratio of FB is always higher than those of state-of-the-art algorithms.

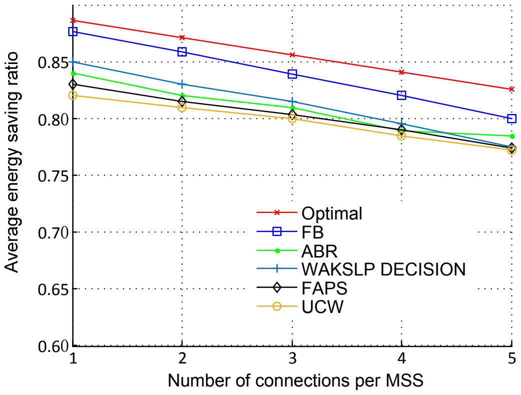

Fig. 6 Number of connections per MSS versus the average energy saving ratio

FB: factor base; ABR: adaptive bandwidth; FAPS: frame aggregation based power-saving scheduling; UCW: user counting window; MSS: mobile subscriber stations

5.4 Impact of the number of MSSs

In this section, we set the number of MSSs to different values to see its impact. We set up two connections for each MSS. Other parameters were the same as in the previous simulation. The simulation results are shown in Fig. 7. Obviously, the energy saving ratio of FB is higher than those of state-of-the-art algorithms. We can also see that as the number of MSSs increases, the energy saving ratios of the algorithms decrease slowly at first, and then the decrease becomes very fast. The reason is that when the number of MSSs reaches a threshold, no more MSS is allowed to enter the sleep mode. As a result, a part of MSSs cannot save energy. Thus, the average energy saving ratio across all MSSs becomes very low.

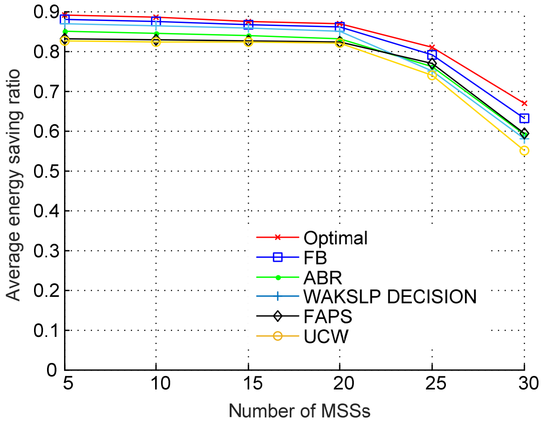

Fig. 7 Number of MSSs versus the average energy saving ratio

FB: factor base; ABR: adaptive bandwidth; FAPS: frame aggregation based power-saving scheduling; UCW: user counting window; MSS: mobile subscriber stations

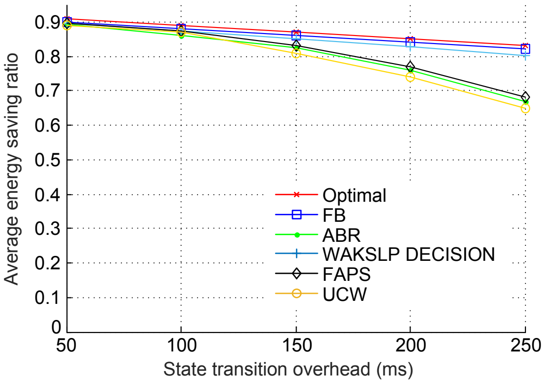

Fig. 8 State transition overhead versus the average energy saving ratio

FB: factor base; ABR: adaptive bandwidth; FAPS: frame aggregation based power-saving scheduling; UCW: user counting window

5.5 Impact of state transition overhead

We can rewrite Eq. (1) as

$$\gamma=\sum_{i=1}^{N}(T_i-L_i)/(T_iN)-T_0\sum_{i=1}^{N}1/(T_iN). \quad (4)$$

For T_i, i=1, 2, …, N, $\sum_{i=1}^{N}1/(T_iN)$ is a constant. Thus, the energy saving ratio γ decreases linearly with the increase of transition overhead T_0. In this section, we ran a simulation with different state transition overheads to show its impact. The number of MSSs was reset to 18. Other parameters remained the same as before. Fig. 8 depicts the simulation results. It is seen that with the same state transition overhead, the average energy saving ratio of FB is higher than those of state-of-the-art algorithms. We can also see that the performance gap between the proposed FB algorithm and state-of-the-art algorithms increases with the increase of state transition overhead. This simulation result proves that the proposed algorithm can save more energy than others, and the more the state transition overhead, the more the benefit the proposed algorithm can bring. This is because the state transition overhead reduces energy saving. The proposed algorithm achieves less state transition overhead than others. Thus, it can save more energy, and the more the state transition overhead, the more the energy it can save, compared to others.

6 Conclusions

In this paper, we proposed a Type II PSC setting algorithm for IEEE 802.16e/m networks with consideration of state transition overhead. The core idea of the proposed algorithm is to set the sleep cycles of the MSSs to multiples of a factor of the shortest sleep cycle. This permits longer sleep cycles to be longer, and a lower resulting state transition overhead, compared to algorithms in the literature. Simulation results showed that, on average, the energy saving ratio of the proposed algorithm is higher than those of state-of-the-art algorithms. Also, the energy saving ratio of the proposed algorithm approaches the limit.

Acknowledgements

This research was supported by the Ministry of Science, ICT and Future Planning (Korea), the Information Technology Research Center Support Program (No. IITP-2016-H8601-16-1005), the Research Fund of Hanyang University, Korea (No. HY-2016), and supervised by the Institute for Information & Communications Technology Promotion.

References

Baek, S., Son, J.J., Choi, B.D., 2009. Performance analysis of sleep mode operation for IEEE 802.16m advanced WMAN. IEEE Int. Conf. on Communications Workshops, p.1-4. http://dx.doi.org/10.1109/ICCW.2009.5208113

Broadband Wireless Access Working Group, 2006. IEEE Standard for Local and Metropolitan Area Networks: Part 16, 802.16e-2005, WG802.16. http://dx.doi.org/10.1109/IEEESTD.2006.99107

Broadband Wireless Access Working Group, 2011. IEEE Standard for Local and Metropolitan Area Networks: Part 16: Air Interface for Broadband Wireless Access Systems Amendment 3: Advanced Air Interface. WG802.16. http://dx.doi.org/10.1109/IEEESTD.2011.5765736

Chen, C.Y., Hsu, C.H., Feng, K.T., 2010. Performance analysis and comparison of sleep mode operation for IEEE 802.16m advanced broadband wireless networks. IEEE Int. Symp. on Personal Indoor and Mobile Radio Communications, p.1425-1430. http://dx.doi.org/10.1109/PIMRC.2010.5671992

Chen, T.C., Chen, J.C., 2009. Extended maximizing unavailability interval (eMUI): maximizing energy saving in IEEE 802.16e for mixing type I and type II PSCs. *IEEE Commun. Lett.*, **13**(2):151-153. http://dx.doi.org/10.1109/LCOMM.2009.081725

Chen, T.C., Chen, J.C., Chen, Y.Y., 2009. Maximizing unavailability interval for energy saving in IEEE 802.16e wireless MANs. *IEEE Trans. Mob. Comput.*, **8**(4):475-487. http://dx.doi.org/10.1109/LCOMM.2009.081725

Cookson, A.H., 1985. Particle Trap for Compressed Gas Insulated Transmission Systems. US Patent 4 554 399.

Feng, H.W., Li, H.Y., 2013. Design of predictive and dynamic energy-efficient mechanisms for IEEE 802.16e. *Wirel. Pers. Commun.*, **68**(4):1807-1835. http://dx.doi.org/10.1007/s11277-012-0551-4

Hwang, E., Kim, K.J., Son, J.J., *et al.*, 2010. The power-saving mechanism with periodic traffic indications in the IEEE 802.16e/m. *IEEE Trans. Veh. Techol.*, **59**(1):319-334. http://dx.doi.org/10.1109/TVT.2009.2032193

Jin, S., Choi, M., Choi, S., 2010. Performance analysis of IEEE 802.16m sleep mode for heterogeneous traffic. *IEEE Commun. Lett.*, **14**(5):405-407. http://dx.doi.org/10.1109/LCOMM.2010.05.091730

Jin, S., Chen, X., Qiao, D., *et al.*, 2011. Adaptive sleep mode management in IEEE 802.16m wireless metropolitan area networks. *Comput. Netw.*, **55**(16):3774-3783. http://dx.doi.org/10.1016/j.comnet.2011.03.002

Kalle, R., Raj, M., Das, D., 2009. A novel architecture for IEEE 802.16m subscriber station for joint power saving class management. Int. Conf. on Communication Systems and Networks, p.1-10. http://dx.doi.org/10.1109/COMSNETS.2009.4808868

Kao, C.C., Yang, S.R., Chen, H.C., 2012. A sleep-mode interleaving algorithm for layered-video multicast services in IEEE 802.16e networks. *Comput. Netw.*, **56**(16):3639-3654. http://dx.doi.org/10.1016/j.comnet.2012.07.013

Kim, R.Y., Mohanty, S., 2010. Advanced power management techniques in next-generation wireless networks. *IEEE Commun. Mag.*, **40**(3):94-102. http://dx.doi.org/10.1109/MCOM.2010.5458369

Lin, Y.W., Wang, J.S., 2013. An adaptive QoS power saving scheme for mobile WiMAX. *Wirel. Pers. Commun.*, **69**(4):1435-1462. http://dx.doi.org/10.1007/s11277-012-0644-0

Liu, W.J., Feng, K.T., Tseng, P.H., 2014. Optimality of frame aggregation-based power-saving scheduling algorithm for broadband wireless networks. *IEEE Trans. Wirel. Commun.*, **13**(2):577-591. http://dx.doi.org/10.1109/TW.2013.123013.121540

Park, Y., Hwang, G.U., 2009. An efficient power saving mechanism for delay-guaranteed services in IEEE 802.16e. *IEICE Trans. Commun.*, **E92-B**(1):277-278.

Park, Y., Leem, H., Sung, D.K., 2010. Power saving mechanism in IEEE 802.16m. IEEE Vehicular Technology Conf., p.1-5. http://dx.doi.org/10.1109/VETECS.2010.5493680

Seo, J.B., Lee, S.Q., Park, N.H., *et al.*, 2004. Performance analysis of sleep mode operation in IEEE 802.16e. IEEE Vehicular Technology Conf., p.1169-1173. http://dx.doi.org/10.1109/VETECF.2004.1400205

Tseng, Y.C., Chen, J.J., Yang, Y.C., 2011. Managing power saving classes in IEEE 802.16 wireless MANs: a fold-and-demultiplex method. *IEEE Trans. Mob. Comput.*, **10**(9):1237-1247. http://dx.doi.org/10.1109/TMC.2010.215

Wong, G.K.W., Zhang, Q., Tsang, D.H.K., 2010. Switching cost minimization in the IEEE 802.16e mobile WiMAX sleep mode operation. *Wirel. Commun. Mob. Comput.*, **10**(12):1576-1588. http://dx.doi.org/10.1002/wcm.875

Wu, C.Y., Ho, H.J., Lee, S.L., 2012. Minimizing energy consumption with QoS constraints over IEEE 802.16e networks. *Comput. Commun.*, **35**(14):1672-1683. http://dx.doi.org/10.1016/j.comcom.2012.06.012

Xiao, Y., 2005. Energy saving management in the IEEE 802.16e wireless MAN. *IEEE Commun. Lett.*, **9**(7):595-597.

Zhu, F., Wu, Y., Niu, Z., 2009. Delay analysis for sleep-based power saving mechanisms with downlink and uplink traffic. *IEEE Commun. Lett.*, **13**(8):615-617.