A Feasible Power Saving Scheduling Approach in WiMAX 802.16e Networks

WEN-CHUAN HUANG
Department of Computer Science and Information Engineering
Tamkang University
No.151, Yingzhuan Rd., Tamsui District, New Taipei City 25137
TAIWAN (R.O.C.)
wenchuanhuang@gmail.com

Abstract: - IEEE 802.16 is a novel broadband wireless access standard. Quality of service (QoS) guarantee and power saving are two important factor in IEEE 802.16 standard. Unsolicited grant service (UGS) is a constant bit rate QoS service and is defined in IEEE 802.16 standard. Power saving classes of type II (PSCs-II) is provided in IEEE 802.16 standard for repeat fixed active and sleep frames. However, IEEE 802.16 standard does not define how to develop an approach for power saving. This paper aims to provide a solution that provides a power saving schedule for one mobile station with multi UGS connections. The proposed approach is named Bucket Checker (BC) that follows the IEEE 802.16 standard definitions, UGS and PSCs-II. In addition, BC aims to provide a best power saving efficiency. By the numerical results, it shows that the proposed approach has better power saving efficiency and always follows QoS requirements.

Key-Words: - WiMAX, 802.16, power saving, schedule, UGS, PSCs

1 Introduction

IEEE 802.16[1] is a long distance and fast transmission broadband wireless access (BWA) standard. Worldwide Interoperability for Microwave Access (WiMAX) is a forum that certifies and promotes the compatibility and interoperability of wireless products based upon IEEE Standard 802.16. Quality of service (QoS) is another advantage of IEEE 802.16 standard. QoS is one of important technologies for real-time streaming application in BWA. In IEEE 802.16 standard, QoS data delivery service is associated with certain predefined set of QoS-related service parameters. IEEE 802.16 supports five kinds of QoS-related services, unsolicited grant service (UGS), extended real-time variable rate (ERT-VR) service, real-time variable rate (RT-VR) service, non-real-time variable rate (NRT-VR) service, and best effort (BE) service. UGS is defined to support real-time applications generating fixed-rate data in IEEE 802.16 standard. Minimum reserved traffic rate and maximum latency are two of UGS parameters. Both ERT-VR and RT-VR services are defined to support real-time applications with variable data-rates in IEEE 802.16 standard. And both ERT-VR and RT-VR services require guaranteed data and delay. NRT-VR is defined to support applications that require a guaranteed data rate but are insensitive to delays in IEEE 802.16. The latest QoS-related service BE service is defined for applications with no rate or delay requirements in IEEE 802.16.

In addition, WiMAX system supports mobile station (MS) which can send and receive data when station is moving. Due to mobility, MS's power is supported from battery. For extenuation MS operation time, IEEE 802.16 define sleep mode. Sleep mode is a state in which an MS conducts pre-negotiated periods of absence from the Serving base station (BS) air interface. In these periods, MS turn off its wireless transceiver model to saving power usage. For management MSs enter or leave sleep mode, IEEE 802.16 standard defines Power Saving Class (PSC). PSC is a group of connections that have common demand properties. PSC may be repeatedly activated and deactivated. Activated and deactivated frames also are named as availability and unavailability interval, respectively. Unavailability interval is a time interval that does not overlap with any listening window of any active power saving class. Availability interval is a time interval that does not overlap with any unavailability interval. As mentioned above, during unavailability interval the BS shall not transmission to the MS, due to MS powers down its transceiver. And, during availability interval the MS is expected to receive all buffered data in unavailability interval. IEEE 802.16 standard defines three types of power saving classes, type I, II, and III, which indicate the procedures of activation or deactivation and policies of MS availability for data transmission. Fig. 1 illustrates theses three types of power saving classes. Power saving classes of type I (PSCs-I) is recommended for connections of BE and NRT-VR. PSCs-I becomes active at the frame specified as start frame number for first sleep window. Each next sleep window of PSCs-I is twice the size of the previous one (S1), but not greater than specified final value. Sleep windows of PSCs-I are interleaved with listen windows of fixed duration. Power saving classes of type II (PSCs-II) is recommended for connections of UGS and RT-VR. PSCs-II includes three parameters, initial-sleep window, listening window, and start frame number for first sleep window. PSCs-II becomes active at the frame specified as “start frame number for first sleep window”. All sleep windows are of the same size as initial-sleep window (S2). Sleep windows are interleaved with listen windows of fixed duration. Power saving classes of type III (PSCs-III) is recommended for multicast as well as for management operations. Deactivation of PSCs-III occurs automatically after expiration of sleep window (S3).

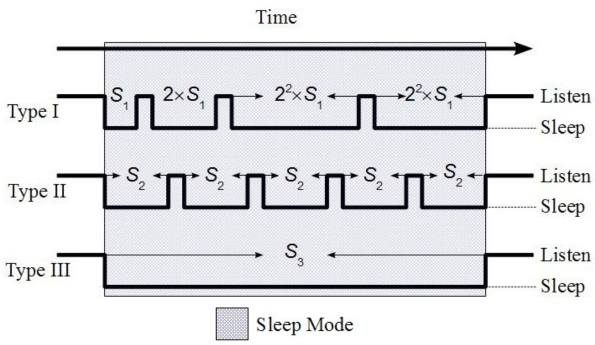

Fig. 1 Power saving classes

Following IEEE 802.16 standard suggestion, MS which connects to BS with several UGS connections chooses PSCs-II to conserve power. The procedure of enter PSCs-II is described as below. First, MS sends MOB_SLP-REQ message to BS. Then, BS return MOB_SLP-RSP message. On the other hand, BS can send MOB_SLP-RSP message to initiate activation of PSCs-II without receiving MOB_SLP-REQ. Both these two procedures must indicate three parameters, the start frame, the number of frames in an unavailability interval, and the number of frames in an availability interval. But, the method to decide these three parameters does not be defined in IEEE 802.16 standard.

To sum up the above arguments, this paper aims to provide an approach which can decide the start frame, the number of frames in an unavailability interval, and the number of frames in an availability interval for one and only one PSCs-II. The PSCs-II is provided for one MS which connect to BS by several UGS connections. In addition, the granted bandwidth in each frame is fixed and cannot increase. The proposed approach only provides one PSCs-II, because each PSCs-II construction procedure needs at least one message. Meanwhile, the proposed approach does not violate all QoS request of UGS connections. Furthermore, the proposed approach should complete all procedures in a feasibility time interval, for example the time of one frame.

This paper is organized as follows: related works are presented in section II. This paper proposed approach will be described in section III. Section IV and V present the simulation result and conclusion, respectively.

2 Related Work

As mention previously, power saving is one of mobile communication problems. Many articles study at power saving problems. Reference [2] explains what is PSCs-I. References [3] – [5] provide the analyzing methodologies for PSC. But all of [2] – [5] do not propose any methodology to decide the parameters of PSC. References [6] – [11] provide decision approaches for the parameters of PSC. Reference [6] enhances the performance of PSC. References [7] and [8] do not distinguish the connection QoS requirements. Reference [9] works for PSCs-I. References [10] and [11] decide unavailable and available intervals with semi-Markov. All aforementioned articles cannot serve for PSCs-II and do not fit UGS connections.

For PSCs-II, recent papers [12] and [13] suggest a methodology to decide unavailable and available intervals by Chinese Remainder Theorem. But the power saving efficiency of this methodology can be improved when consider maximum latency. To improve power saving efficiency, MPC is provided in [14]. MPC employs two major parameters, the frame numbers of unavailable interval (I_u) and the frame numbers of available interval (I_a). The computations of MPC are following two constraints by I_u and I_a. One is delay constraint. The other is bandwidth constraint. The delay constraint of MPC limits I_a+I_u have to less than the minimum of all UGS connections’ maximum latency. The bandwidth constraint of MPC makes sure that all data arrived in I_u can be transmitted in next I_a and all data arrived in I_a have to be transmitted before the end of I_a. For example, there are three UGS connections between one MS and BS. The parameters of these three UGS connections are list in Table 1. The granted bandwidth of MS is 20 kilobits (Kb) per frame. Fig. 2 is the illustration of computation result. The x-axis and y-axis are the number of available interval frames and the number of unavailable interval frames, respectively. However MPC does not indicate the start frame number for first sleep window. In addition, the suggest results of MPC may violate QoS requirements, due to the bandwidth constraint of MPC following average bandwidth requisition. Fig. 3 illustrates the against QoS result which is suggested by MPC in the previous example. In Fig. 3, y-axis is the number of bits. And, x-axis is frame number. Each block is a packet. The first part of packet label is link ID. The second part of packet label is packet sequence number. All three UGS link arrive a packet in frame 0. The light blue line indicates MS is in unavailable interval or available interval. In frame 1, MS is available and only can transmit two packet, $Packet_{2,1}$ and $Packet_{3,1}$. $Packet_{1,1}$ does not be transmitted before its maximum latency, frame 2.

Table 1 The parameters of three example UGS connections

link ID	data arrival interval	data size	maximum latency
l_1	2 frames	10 Kb	3 frames
l_2	3 frames	10 Kb	3 frames
l_3	6 frames	10 Kb	2 frames

To fix against result of MPC, [15] and [16] propose new bandwidth constraint, replacing average bandwidth requisition per frame with the maximum bandwidth requisition in an I_u+I_a interval. The approach of [15] is named PS. Because bandwidth constraint of PS following the maximum bandwidth requisition of I_u+I_a interval, the power saving efficiency is worse than MPC. The worst of PS is no unavailability intervals. Fig. 4 is the computation result for the previous example listed in Table I and also is one of the worst computation results.

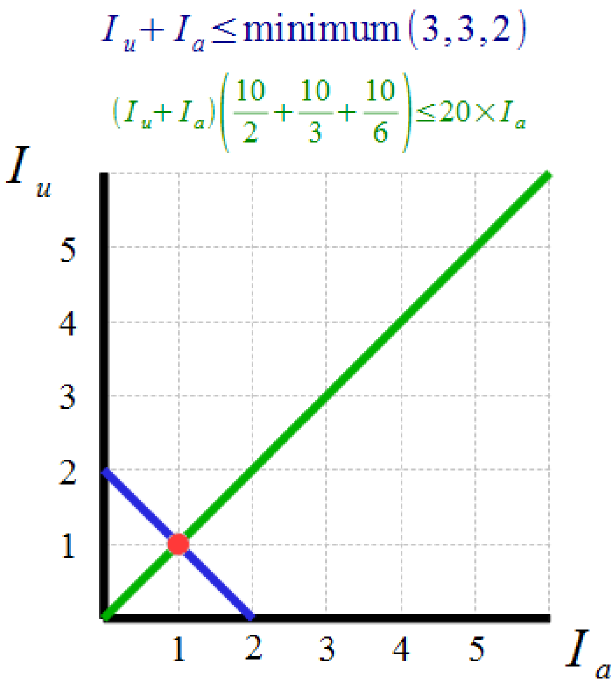

Fig. 2 Computation result of MPC example

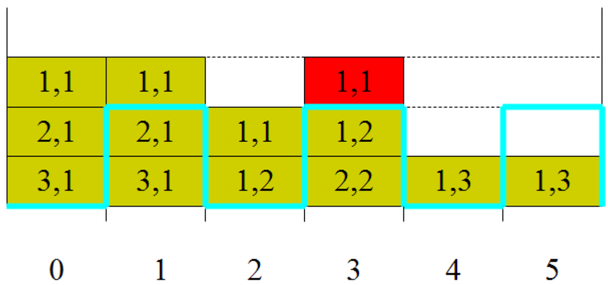

Fig. 3 MPC result is no following QoS requisition

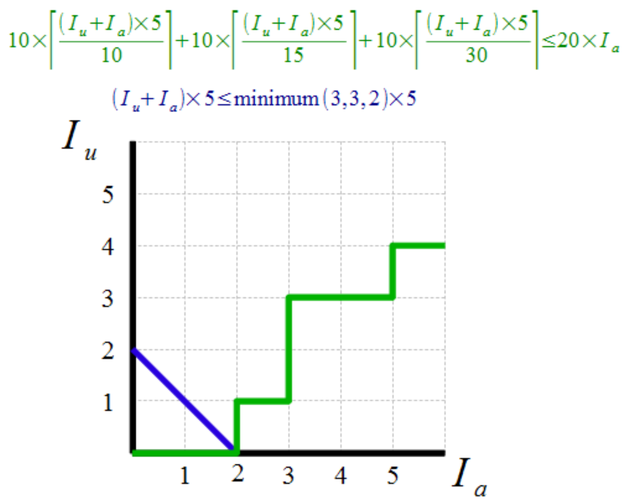

Fig. 4 The computation result of PS example

Reference [17] enhances power saving efficiency by FD approach. FD also has delay constraint and bandwidth constraint. But, delay constraints of FD are decided connection by connection. The tightest delay constraints equals to half of the minimum of all I_u+I_a. The others delay constraints must be multiple of the tightest delay. But the bandwidth constraints of FD are also following the maximum bandwidth in I_u+I_a. As PS, the worst result of FD does not have unavailable intervals.

This paper will propose a novel approach, bucket check (BC), which considers the delay constraint, packet arrival interval. Meanwhile, the proposed approach only needs one PSCs-II for all UGS connections. Next session is the detail of proposed approach.

3 Bucket Checker Approach

This paper considers an MS with n UGS links, l_i, i =1...n. Link l_i's QoS parameters, $l_i.d_{max}$ and $l_i.I_d$, are already known to the MS and are summarized as follows:

- $l_i.d_{max}$: The delay constraint in number of frames for UGS connection l_i.
- $l_i.I_d$: The packet arrival interval (frames) for UGS connection l_i.
- $l_i.r$: The packet size of bits.
- $l_i.f$: The frame number of first arrival packet for UGS connection l_i, where $0 \le l_i.f \le l_i.I_d - 1$.

To guarantee the bandwidth requirement of UGS links, the maximum granted transmission rate, r (bits per frame), is granted. The goal of this paper is to compute a PSCs-II to maximum the power saving for this MS. The following parameters of PSCs-II will be determined:

- I_u: The frame number of unavailable interval.
- I_a: The frame number of available interval.
- f: The start frame number for first sleep window.

Specifically, power saving cycle (C_p), schedule cycle (C_s), the set of available C_p for all UGS links (S), and efficiency of power (e) saving are defined as bellow:

- $p_{i,j}$: the jth packet of l_i.
- $p_{i,j}.f_a$: the arrival frame number of $p_{i,j}$.
- $p_{i,j}.f_d = p_{i,j} + l_i.d_{max}-1$.
- $p_{i,j}.r = l_i.r$.
- $C_p = I_u + I_a$ where I_u, $I_a > 0$.
- C_s: least common multiple (LCM) of C_p and all $l_i.I_d$.
- $S = \{s_1, s_2..s_m\}$ where s_j is a member of S.
- $e=I_u/(I_u+I_a)$.

By definition of C_s, the scenario of packets arrival of all UGS links and power saving status of MS will repeat after C_s frames. In addition, giving any k continuation frames, $[f_x, f_{x+k-1}]$, contains m available frames. S is a set of packets and is composed by packets which $f_x \le p_{i,j}.f_a$ and $p_{i,j}.f_d \le f_{x+k-1}$. Then, equation (1) always is true when QoS of all UGS links do not violate. The left-hand side of equation (1) is the total demand bandwidth which has to be sent in $[f_x, f_{x+k-1}]$. And, the right-hand side of equation (1) is the maximum available bandwidth in $[f_x, f_{x+k-1}]$.

$$\sum_{p\in P} p_{i,j}.r \le m\times r \tag{1}$$

Based on equation (1), a brute force algorithm is showed in fig. 5. However, this brute force algorithm needs a huge computation time. This paper will propose a solution to decrease computation. The propose solution contents there methods. First method is deciding available power saving cycle. Second method is computation the best efficiency of power saving. Last method is bucket checker algorithm. The detail of these methods will describe in the next subsections.

Brute Force

Input: r, n, $l_i.d_{max}$, $l_i.I_d$, $l_i.r$, $l_i.f$

Output: I_u, I_a, f

forall s_j
  forall I_u
    $I_a = s_j - I_u$
    forall f
      if $\sum_{p\in P} p_{i,j}.r \le m\times r$
        $e = \frac{I_u}{I_u+I_a}$
        if $e_{best} < e$
          $e_{best} = e$

Fig. 5. Brute Force algorithm.

3.1 Available Power Saving Cycle

For each UGS requisition, the immediate delay constraint is small than its maximum latency. For instance, a UGS connection l_i with $l_i.d_{max}$=2 and $l_i.I_d$=6 is limited by delay constraint I_u<2. The immediate delay constraint of li illustrates in fig. 6a and can be defined as equation (2). However, fig. 6a shows that there are two unnecessary available intervals. Furthermore, fig. 6b shows C_p=6 also not violation the QoS requirements of l_i. This observation means QoS requirements of one UGS link can be implemented by a power saving cycle which brought every packet arriving in available intervals. This situation can be defined as equation (3). It is simple to extend equations (2) and (3) to multi UGS links. When equations (2) and (3) are followed for all UGS links, the QoS requirements of all UGS links are not violated.

$$I_u < l_i . d_{max} \quad (2)$$

$$(l_i . I_d \times x + l_i . f - f) \equiv y \pmod{C_p} \text{ where } x \in I \text{ and } y > I_u \quad (3)$$

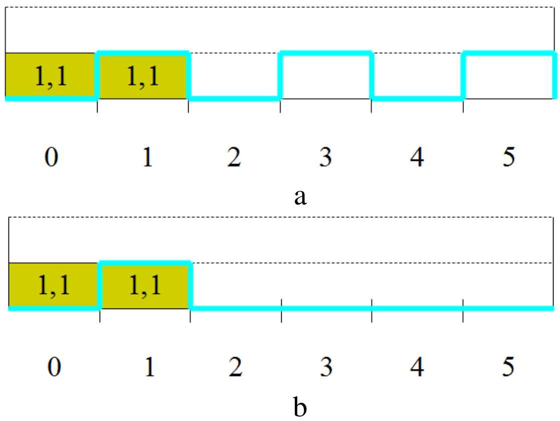

Fig. 6. Example of C_p for each UGS connection.

3.2 The best efficiency of power saving

As mention in equation (1), any k continuation frames have to contain enough available frames. However, the computation complexity for choice the best efficiency of power saving is too complex. But, fig. 7 shows that the proposed approach can computation the best efficiency of power saving with finite types. The light blue line in fig. 7 is the power saving cycle which contains 2 frames unavailable intervals and 2 frames available intervals. Frames 1 and 2, [f_1, f_2], are unavailable interval. And [f_3, f_4] is available interval. [f_2, f_5] is one of the continue k frames where k=4. For reducing computation complexity, [f_1, f_6] will replace [f_1, f_5], [f_1, f_4], [f_2, f_6], [f_2, f_5], [f_2, f_4], [f_3, f_6], [f_3, f_5], and [f_3, f_4], because all of these continue frames contain the same available interval [f_3, f_4]. These finite typifications continue frames are named buckets. The proposed approach BC partitions time axis into buckets. Every bucket does not cross over the others. The first frame of each bucket is the next frame of each available frame. And, the last frame of each bucket is the first available frame since its first frame. For example, [f_1, f_3], [f_4, f_4], and [f_5, f_7] are buckets, in fig. 7. By buckets, the parameters of PSCs-II can be checked whether all buckets follow equation (1) to confirm all QoS requirements of UGS links. Because the number of buckets is less than any k continue frames, the computation complexity will reduce when any k continue frames are replaced by buckets. For decreasing more computation complex, this paper proposed bandwidth checker function to decreasing computation complexly. Bandwidth checker function is a table–based algorithm and return whether bandwidth in bucket is enough or not. Let f_0 is the start frame number for first sleep window. The table grids of bucket checker are labeled as $B_{k,m}$ where the first index k is the bucket number from f_0, and the last index m is the available frame number in this grid. In addition, $B_{k,m}.f_s$ and $B_{k,m}.f_e$ are the first frame and the last frame of $B_{k,m}$, respectively. Bandwidth checker will return v after checking whether requisition of ($B_{k,m}.c$) over volume for each bucket. v is a binary variable and defines in below.

$$v = \begin{cases} 0 & \text{if bandwidth enough} \\ 1 & \text{otherwise} \end{cases} \quad (4)$$

Fig. 8 is the algorithm of bandwidth checker function.

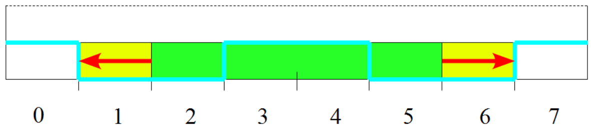

Fig. 7. Classification frames into finite buckets.

Bandwidth Checker

Input: r, n, f, C_s, C_p, I_u, I_a, $l_i.d_{max}$, $l_i.I_d$, $l_i.r$, $l_i.f$
Output: v
01. forall $p_{i,j}$ where $p_{i,j}.f_a \geq f$ and $p_{i,j}.f_d \leq f + C_s - 1$
02.   forall $B_{k,m}$ where $1 \leq k \leq \frac{C_s}{C_p} \times I_a$ and $0 \leq m \leq \frac{C_s}{C_p} \times I_a$
03.   if $B_{k,1}.f_s \leq p_{i,j}.f_a \leq B_{k,1}.f_e$
04.   if $B_{k+m,1}.f_s \leq p_{i,j}.f_d < B_{k+m,1}.f_e$
05. $B_{k,m}.c = B_{k,m}.c + p_{i,j}.r$
06. for m = 2 to $\frac{C_s}{C_p} \times I_a$
07.   for k = 1 to $\frac{C_s}{C_p} \times I_a$
08. $B_{k,m}.c = B_{k,m}.c + B_{k,m-1}.c + B_{k+1,m-1}.c - B_{k+1,m-2}.c$
09. if $\exists\ B_{k,m}.c > m \times r$ where $1 \leq k \leq \frac{C_s}{C_p} \times I_a - 1$ and $0 \leq m \leq \frac{C_s}{C_p} \times I_a$
10. $v = 1$
11. otherwise $v = 0$

Fig. 8. Bandwidth checker function.

For example, there are three UGS links list in table I. The parameters of these three UGS links are $l_1.d_{max}$=3, $l_1.I_d$=2, $l_1.r$=10, $l_1.f$=0, $l_2.d_{max}$=3, $l_2.I_d$=3, $l_2.r$=10, $l_2.f$=0, $l_3.d_{max}$=2, $l_3.I_d$=6, $l_3.r$=10, and $l_3.f$=0. And the parameters of PSCs-II are r=20, f=0, I_u=1, and I_a=1. In addition, n=3, C_p=2, and C_s=6. After classing, $p_{1,1}$, $p_{2,1}$, and $p_{3,1}$ belong to $B_{1,1}$. $p_{1,2}$, $p_{1,3}$, and $p_{2,2}$ belong to $B_{2,1}$, $B_{3,1}$, and $B_{2,2}$, respectively. Through lines 1 to 5 of bandwidth checker function, $B_{1,1}.c$=10+10+10=30, $B_{2,1}.c$=10, $B_{2,2}.c$=10, and $B_{3,1}.c$=10. After lines 8 of bandwidth checker function, $B_{1,1}.c$=30, $B_{2,1}.c$=10, $B_{2,2}.c$=10+10+10-0=30, and $B_{3,1}.c$=10. Finally, v=1 because $B_{1,1}.c$=30>20. So, this PSCs-II is not acceptable for these three UGS lines.

3.3 Bucket checker

Although bandwidth checker function can verify whether bandwidth is enough or not, all iterations still need to be tested. This paper proposes bucket checker algorithm to decrease computation complexity. Fig. 9 is the detail of bucket checker algorithm where maximum available power saving efficiency (e_{max}) and minimum available power saving efficiency (e_{min}) are employed to decrease computation times. Only some iterations which have power saving efficiency between e_{max} and e_{min}, need to be checked, illustration in fig. 10. For instance, there are two UGS links with $l_1.d_{max}$=5, $l_1.I_d$=3, $l_1.r$=30, $l_1.f$=0, $l_2.d_{max}$=5, $l_2.I_d$=6, $l_2.r$=10, and $l_2.f$=1. And the other parameters are r=60, and n=2. By line 2, S = {2, 3, 4, 5}. First round of for loop, I_u=2, I_a=0, C_p=2, and e=1. Because e equals to e_{max}, I_u, I_a, and e are changed to 1, 1, and 0.5, respectively. Due to $I_u<l_i.d_{max}$ for all l_i, this iteration need to be test by bandwidth checker function. And the result of bandwidth checker function is 0. First round ends after line 12. In the same time, e_{min} changes to 0.5. Fig. 10 illustrates $l_i.d_{max}$ for all l. as red line and e_{min} change from x-axis to right green line. In second and third rounds, bandwidth checker function only run one time for each round. I_u=2, I_a=1, in second round. I_u=3, I_a=1, in third round. Both results of bandwidth checker function are 0, in second and third rounds. The last e_{min} equals to 0.75. In fourth round, I_u=4, I_a=1, bandwidth checker function also need be run. This time result of bandwidth checker function is 1. Then, e_{max} is changed from y-axis to 0.8, and is showed as left green line in fig. 10. There is no more test, because there is no any available I_u and I_a values which let $e_{min}<e<e_{max}$. Final round, there is also no any available I_u and I_a values which let $e_{min}<e<e_{max}$. Bandwidth checker function only is run 4 times, in this example. This example presents that bucket checker algorithm can reduce the computation complex. In next section, the simulation also shows that bucket checker algorithm is a feasible solution for choosing the best power saving cycle of multi UGS links.

Bucket Checker

Input: r, n, $l_i.d_{max}$, $l_i.I_d$, $l_i.r$, $l_i.f$

Output: I_u, I_a, f

01. initiation $e_{max}=1$, $e_{min}=0$
02. computation S
03. forall s_j in increasing order
04. $I_u=\lfloor e_{max}\times s_j \rfloor$; $I_a=s_j-I_u$; $C_p=s_j$; $e=\frac{I_u}{I_u+I_a}$
05. while $e\geq e_{max}$
06. $I_a=I_a+1$; $I_u=I_u-1$; $e=\frac{I_u}{I_u+I_a}$
07. while $e<e_{max}$ and $e>e_{min}$ and $\{(l_i.I_d\times x+l_i.f-f)\equiv y \pmod{C_p}$ or $(I_u<l_i.d_{max})\}$
08. v=**bandwidth checker**
09. if v=1
10. $e_{max}=e$; $I_a=I_a+1$; $I_u=I_u-1$; $e=\frac{I_u}{I_u+I_a}$
11. else
12. $e_{min}=e$

Fig. 9. Bucket Checker algorithm.

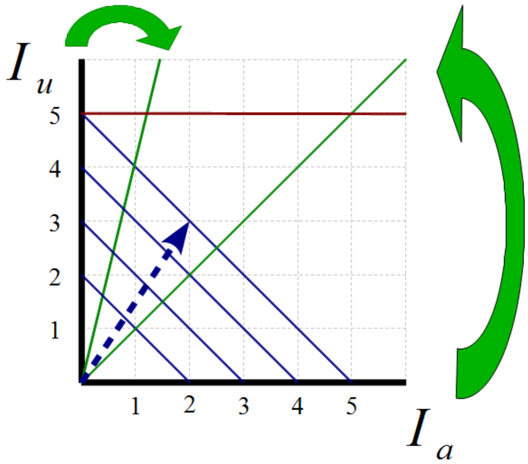

Fig. 10. Bucket Checker operation steps.

4 Simulation

A simulator has been developed in JAVA to evaluate the power saving efficiency, unsuitable result number, and computation time, in this paper. In the developed simulator the frame length is 5 ms, and r = {384, 448, 512, 576}, $2\leq n\leq 5$, $5\leq l_i.d_{max}\leq 10$, $3\leq l_i.I_d\leq 8$, $64\leq l_i.r\leq 192$. This paper compares the proposed approach (BC) against with the approaches in [14] (MPC), [15] (PS), and [17] (FD). The simulation result is the statistic of running developed simulator 100 000 times.

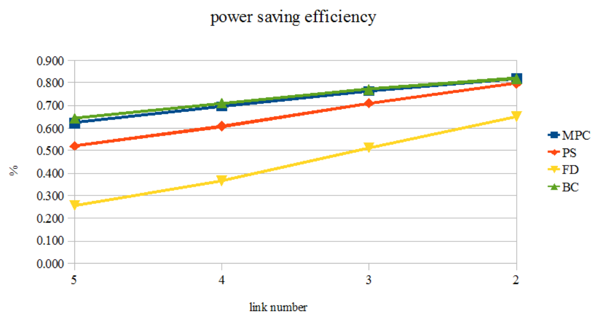

Fig. 11. Power saving efficiency versus link number.

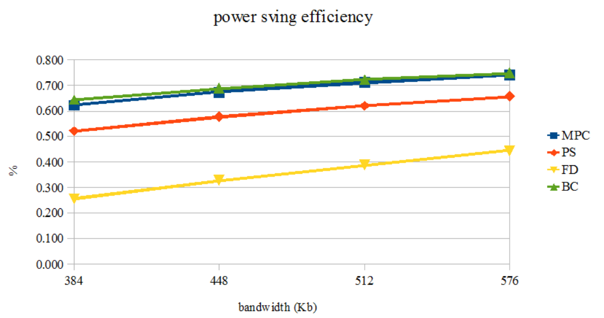

Fig. 12. Power saving efficiency versus maximum granted bandwidth.

First, fig. 11 and fig. 12 show the difference of power saving efficiency and sleep ratio, between MPC, PS, FD, and BC. Fig. 11 focuses on the relation of power saving efficiency and link number. As link number decreases, power saving efficiency decreases for all MPC, PS, FD, and BC. On the other hand, fig. 12 focuses on the relation of power saving efficiency and maximum granted transmission rate. As maximum granted transmission rate increases, power saving efficiency decrease for all MPC, PS, FD, and BC. Both figurations show that the proposed approach BC always has the better power saving efficiency than PS and FD. Only MPC has the similar power saving efficiency. This is because both PS and FD use the maximum transmission requisition in one power saving cycle as their bandwidth constrain. But transmission requisition of each power saving cycle does not always equal to the maximum transmission requisition in one power saving cycle. So, both PS and FD have worse power saving efficiency.

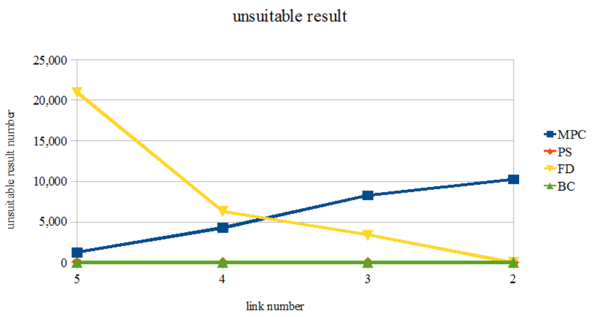

Fig. 13. Unsuitable result number versus link number.

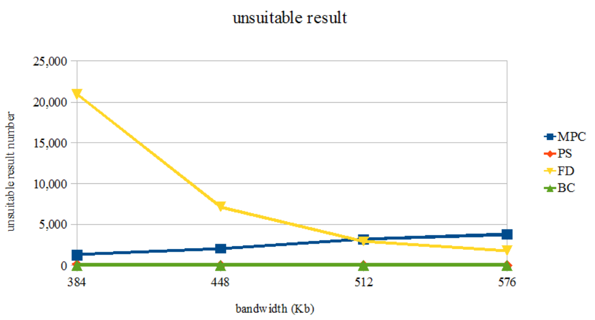

Fig. 14. Unsuitable result number versus maximum granted bandwidth.

However, MPC has more violation QoS results than BC. Fig. 13 illustrates that the proposed approach BC always follows QoS requirements and the numbers of MPC violation QoS results are 1 263, 4 266, 8 305, and 10236, in different link number. In addition, the lines of PS and FD in fig.13 indicate the number of no unavailable frames result. As link number decreases, unsuitable result number decreases for FD. But, unsuitable result number increases for MPC. PS only has a tiny unsuitable result number. In different maximum granted bandwidth of each frame, fig. 14 is similar to fig. 13. The proposed approach BC still always follows QoS requirements. Both MPC and FD have many unsuitable result numbers.

For choosing the best power saving efficiency, BC needs more computation time. For evaluation the computation time of BC, the developed simulator run in a laptop with i3 CPU and 2GB RAM. The computation time of BC is still less than 3.2 ms, illustration in fig. 15 and fig. 16. One frame often sets to 5 ms that is greater than the computation time of BC. Hence, the proposed approach can take a best power saving efficiency result in one frame.

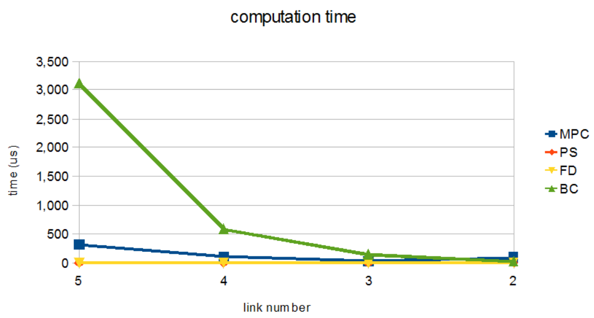

Fig. 15. Computation time versus link number.

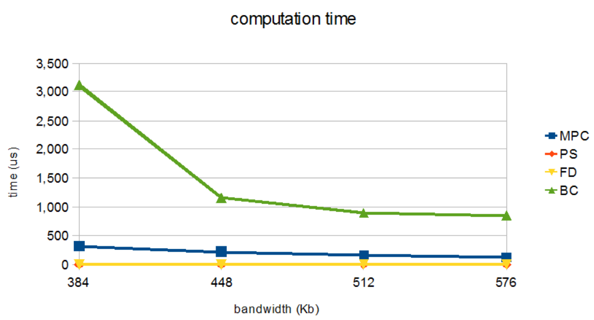

Fig. 16. Computation time versus maximum granted bandwidth.

5 Conclusion

This paper has proposed a novel sleep scheduling approach naming Bucket Checker. BC can provide best power saving efficiency that is present in simulation result. In addition, BC always provides a sleep schedule flowed QoS requirements.

Furthermore, the power saving efficiency of BC is always better than PS and FD. And, BC is always following QoS requirements. Although, BC need more computation time, it is still less than one frame in a general device. Therefore, BC is a feasible and good performance approach.

References:

[1] *IEEE Standard for Local and metropolitan area networks - Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems - Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1*, IEEE standard 802.16e, 2006.

[2] Yang Xiao, Energy saving mechanism in the IEEE 802.16e wireless MAN, *IEEE Commun. Lett.*, Vol. 9, pp. 595 – 597, Jul., 2005.

[3] Yunju Park and Gang Uk Hwang, Performance modelling and analysis of the sleep-mode in IEEE802.16e WMAN, *in Proc. IEEE Veh. Technology Conf.* (VTC-2007 Spring), Dublin, 2007, pp. 2801-2806.

[4] Yan Zhang, Performance modeling of energy management mechanism in IEEE 802.16e mobile WiMAX, in *Proc. IEEE Wireless Commun. and Networking Conf.* (WCNC), Hong Kong, 2007, pp. 3205-3209.

[5] Claudio Cicconetti, Luciano Lenzini, Enzo Mingozzi, and Carlo Vallati, Reducing power consumption with QoS constraints in IEEE 802.16e wireless networks, *IEEE Trans. Mobile Computing*, Vol. 9, pp. 1008-1021, Jul., 2010.

[6] Yang Xiao, Performance analysis of an energy saving mechanism in the IEEE 802.16e wireless MAN, in *Proc. IEEE Consumer Commun. and Networking Conf.* (CCNC), Las Vegas, NV, 2006, pp.406-410.

[7] Jinglin Shi, Gengfa Fang, Yi Sun, Jihua Zhou, Zhongcheng Li, and Eryk Dutkiewicz, Improving mobile station energy efficiency in IEEE 802.16e WMAN by burst scheduling, in *Proc. IEEE Global Telecommun. Conf.* (GLOCOM), SF, California, 2006, pp. 1-5.

[8] Li-Hsing Yen, Chih-Yuan Cheng, and Chun-Hsin Wang, Sleep scheduling that minimizes state transitions for IEEE 802.16e mobile subscriber stations, in *Proc. New Trends Inform. Service Sci.* (NISS), Beijing, 2009, pp. 1341-1346.

[9] Huei-Wen Ferng, Cheng-Lun Liu, Jeng-Ji Huang, and Chih-Cheng Tseng, A scheduling-based delay-tolerant power saving scheme for the IEEE 802.16e wireless MAN, in *Proc. IEEE Veh. Technology Conf.* (VTC-2008 Spring), Singapore, 2008, pp.2321-2325.

[10] Kwanghun Han and Sunghyun Choi, Performance analysis of sleep mode operation in IEEE 802.16e mobile broadband wireless access systems, in *Proc. IEEE Veh. Technology Conf.* (VTC-2006 Spring), Melbourne, 2006, pp. 1141-1145.

[11] Lei Kong and Danny H.K. Tsang, Optimal selection of power saving classes in IEEE 802.16e, in *Proc. IEEE Wireless Commun. and Networking Conf.* (WCNC), Hong Kong, 2007, pp. 1836-1841.

[12] Tuan-Che Chen, Ying-Yu Chen, and Jyh-Cheng Chen, An efficient energy saving mechanism for IEEE 802.16e wireless MANs, *IEEE Trans. Wireless Commun.* , Vol. 7, pp. 3708-3712, Oct., 2008.

[13] Tuan-Che Chen, Jyh-Cheng Chen, and Ying-Yu Chen, Maximizing unavailability interval for energy saving in IEEE 802.16e wireless MANs, *IEEE Trans. Moblie Computing*, Vol. 8, pp. 475-487, Apr., 2009.

[14] Hsin-Lung Tesng, Yu-Pin Hsu, Chung-Hsien Hsu, Po-Hsuan Tseng, and Kai-Ten Feng, A maximal power-conserving scheduling algorithm for broadband wireless networks, in *Proc. IEEE Wireless Commun. and Networking Conf.* (WCNC), Las Vegas, NV, 2008, pp. 1877-1882.

[15] You-Lin Chen and Shiao-Li Tsao, Energy-efficient sleep-mode operations for broadband wireless access systems, in *Proc. IEEE Veh. Technology Conf.* (VTC-2006 Fall), Montreal, 2006, pp. 1-5.

[16] Shiao-Li Tsao and You-Lin Chen, Energy-efficient packet scheduling algorithms for real-time communications in a mobile WiMAX system, *Comput. Commun.*, Vol. 31, pp. 2350-2359, Jun., 2008.

[17] Yu-Chee Tseng, Jen-Jee Chen, and Yen-Chih Yang, Managing power saving classes in IEEE 802.16 wireless MANs: a fold-and-demultiplex method, *IEEE Trans. Mobile Computing*, Vol. 10, pp. 1237-1247, Sep., 2011.