Performance Analysis of Sleep Mode Operation in IEEE 802.16e Mobile Broadband Wireless Access Systems

Kwanghun Han
School of Electrical Engineering and INMC
Seoul National University, Seoul, Korea
Email: khhan@mwnl.snu.ac.kr

Sunghyun Choi
School of Electrical Engineering and INMC
Seoul National University, Seoul, Korea
Email: schoi@snu.ac.kr

Abstract— **In this paper, we analytically evaluate the sleep mode operation of the IEEE 802.16e, defined for the power saving. In the sleep mode, an Mobile Subscribe Station (MSS) sleeps for a sleep window and wakes up at the end of the sleep window in order to check pending download packet(s) destined to itself. If there is no such a packet, the MSS doubles the sleep window up to the maximum value and sleeps again. For a quantitative analysis, we model the sleep mode operation as a semi-Markov chain. Then, the average packet delay and average power consumption are analytically derived. Based on the performance analysis, we discuss selecting the proper set of operational parameter values for a given traffic arrival pattern.**

I. Introduction

During the past decades, the mobile hand-held devices including cellular phones have become very popular. Moreover, to provide both voice and high-bandwidth data services, new systems are being developed. IEEE 802.16e is one of the candidates for the next-generation mobile networking. Originally, the IEEE 802.16 have been designed for the fixed subscriber stations (SSs) [1]. On the other hand, the emerging IEEE 802.16e, which is currently under the standardization, is an extension targeting at the service provisioning to the *Mobile Service Stations* (MSSs) [2]. As part of the mobility extension, the 802.16e defines a handoff procedure and a sleep mode operation. Especially, the sleep mode operation for the power saving is one of the most important features for the battery-powered MSSs to extend their operational lifetime.

Under the sleep mode operation, an MSS initially sleeps for a fixed amount of time, called *sleep window*, and then wakes up in order to find if the base station (BS) has any buffered downlink traffic destined to itself. If there is no such traffic, it basically doubles the sleep window size up to the maximum sleep window size, and then checks with the BS when it wakes up again. The related operational parameters including the initial and maximum sleep window sizes can be negotiated between the MSS and BS. On the other hand, if an MSS has packets to send for uplink transmissions, it can wakes up prematurely to prepare for the uplink transmission, i.e., bandwidth request, and then the MSS can transmit its pending packets upon bandwidth allocation by its BS.

There have been many studies that evaluate the power saving in various systems. In [4], the authors evaluate the energy consumption of various access protocols for wireless infrastructure networks. To support short-lived traffic such as HTTP efficiently, bounded slowdown method which is similar to that of the 802.16e is also proposed for the 802.11 WLAN [7]. The power saving mechanism for the 3G UMTS system is also evaluated in [6].

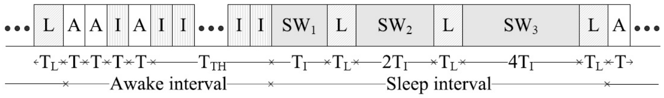
We model the sleep mode operation of the 802.16e as a semi-Markov chain in order to analyze its performance quantitatively. With this model, we obtain the steady state probability distribution in order to derive the packet delay and power consumption performances. From these results, we can select the best values for the operational parameters related to the sleep mode operation, i.e., initial sleep window size and final sleep window size for a given packet arrival rate.

The rest of the paper is organized as follows. In Section II, the sleep mode operation of IEEE 802.16e is briefly explained. The Markov chain modeling as well as the delay and power consumption analysis are presented in Section III. After we evaluate the performance of the sleep mode operation in Section IV, we conclude in Section V.

II. Sleep Mode Operation in IEEE 802.16e

An 802.16e MSS with registration with a specific BS can be in one of two operational modes, namely, *awake mode* and *sleep mode*. MSSs in the awake mode can send or receive data according to the Base Station (BS)'s scheduling. On the other hand, MSSs in the sleep mode can be absent from the serving BS during pre-negotiated intervals. Before switching to the sleep mode from the awake mode, the MSS shall inform the BS using a sleep request message (MOB-SLP-REQ) and obtain its approval through a sleep response message (MOB-SLP-RSP) from the BS. After receiving an MOB-SLP-RSP message from the BS, the MSS can enter the sleep mode.

The sleep mode involves two operational windows (i.e., time intervals), namely, *sleep window* and *listening window*, and an MSS in the sleep mode basically switches between two windows. During a sleep window, an MSS turns off most of its circuits in order to minimize the energy consumption, and hence cannot recieve/transmit any message. If any packet(s) destined to the MSS in the sleep mode arrive at the BS during the sleep window of an MSS, these packets are buffered so that they can be delivered to the MSS when it is awake in the future. During a listening window, an MSS synchronizes with its serving BS' downlink (i.e., BS-to-MSS) and listens to a traffic indication message (MOB-TRF-IND), which indicates



T: frame duration, T_I: initial sleep window duration, T_L: listening window duration, T_{TH}: idle frame threshold, A: packet serving frame, I: idle frame, SW_i: i-th sleep window, L: listening window

Fig. 1. A timing diagram of IEEE 802.16e sleep mode operation

whether there is any buffered packet(s) destined to the MSS, to decide whether to stay awake to receive the pending packet(s) or go back to sleep.

The sleep window size basically increases binary exponentially. That is, when an MSS enters the sleep mode from the awake mode, it sleeps during the *initial* sleep window long first. Then, at the beginning of the listening window, the MSS wakes up to receive a MOB-TRF-IND, and if there is no packet buffered and destined to itself, it doubles the sleep window size, and sleeps until the next listening window. Once the MSS has reached the final sleep window size, it shall continue its sleep mode without increasing the sleep window size any further. The values of both initial and final sleep window sizes (along with the listening window size) are determined during the MOB-SLP-REQ/MOB-SLP-RSP exchange. These sleeping-and-listening events repeat with updated sleep window sizes until the MSS is notified of the buffered packets destined to itself, at which instance the MSS enters the awake mode by completing a sleep interval in order to receive the buffered packets.

Fig. 1 illustrates the relationship among the different time intervals when an MSS is served by the serving BS. A and I represent packet serving and idle MAC frames[1], respectively, and SW_i and L represent the i-th sleep window and listening window, respectively. T, T_I and T_L represent the durations of a MAC frame, the initial sleep window, and the listening window duration, respectively. T_F represents the final sleep window although it is not shown in this figure. Finally, T_{TH} is an idle frame threshold; the MSS enters the sleep mode from the awake mode when there is no traffic destined to itself for the time interval of the idle frame threshold.

III. Analytical Modeling

A. Markov Chain Modeling

The sleep mode operation of the 802.16e can be modeled using a semi-Markov chain as shown in Fig. 2, where each state in the chain represents the state of the corresponding MSS. There are basically four types of states, namely, $P_A(i)$ (in the awake mode with i pending packets at the BS), $P_I(j)$ (in the awake mode without receiving any packet for j consecutive MAC frames), $P_S(k)$ (in the sleep mode's sleep window of the size equal to $2^k\cdot$ initial sleep window, $T_S(k)$), and $P_L(l)$ (in the sleep mode's listening window). The semi-Markov chain in the figure assumes (1) a transmission buffer of three downlink packets in the serving BS, (2) an idle frame threshold of three MAC frames, and (3) the final sleep window size equal to two times the initial window size, as an example. In the figure, $P(t,k)$ and $P(t,n>k)$ represent the probabilities that k packets and more than k packets arrive at the BS during time t, respectively.

In addition, we make the following assumptions for the analytical simplicity: (1) we consider the downlink packet arrivals from a single source according to a Poisson process of rate λ ($packets/s$); (2) just a single packet is served during one MAC frame duration; and (3) a listening window is one MAC frame long.

In the awake mode, there are three possible state transitions. First, when there is no packet arrival with probability $P(T,0)$ in the n-th active state, the state goes to the $(n-1)$-th active state when n is larger than 1. When n is 1, i.e., there is only one packet in the serving BS's packet queue, the MSS enter the first idle state. Second, when one packet arrives, the MSS stays at the same state. Last, when more than one packet arrive, according to the number of arrived packets, the MSS's next state will be determined. The excessive packets over the buffer size will be discarded. The followings are the active state transition equations.

$1 \le k < N-1$,

$$P_A(k) = \sum_{i=1}^{k+1} P_A(i)P(T,k-i+1) + \sum_{l=2}^{Q} P_I(l)P(T,k) + \sum_{j=1}^{M}\{P_L(j)\sum_{m=1}^{k}[P(T_S(j)+T,m)P(T,k-m)]\}$$

$k = N-1$,

$$P_A(k) = P_A(k+1) + \sum_{i=1}^{k} P_A(i)P(T,n>k-i) + \sum_{j=1}^{M}\{P_L(j)\sum_{m=1}^{k}[P(T_S(j)+T,m)P(T,k-m)]\} + \sum_{l=2}^{Q} P_I(l)P(T,k)$$

$k = N$,

$$P_A(k) = \sum_{j=1}^{M}\{P_L(j)P(T_S(j)+T,n>k-1\} + \sum_{l=2}^{Q} P_I(l)P(T,n>k-1)$$

If the MSS stays in an idle state, there are two possible state transition cases. First, in the case of one or more packet arrival, the state transition is quite similar to the active state transition case, but much simpler than that because the MSS state shall always move to the same state unlike the active state case if the same number of packets arrive at any idle state. Second, when no packet arrives, the MSS moves to the next idle state, and at the last idle state. If the MSS is in the last idle state, it moves to the first sleep state because the packet can only be transmitted after a DL map. The state transition equations are given as follows.

$$k=1, \qquad P_I(k) = P_S(k)$$
$$1<k<Q, \qquad P_I(k) = P_I(k+1)P(T,0)$$
$$k=Q, \qquad P_I(k) = P_A(1)P(T,0)$$

[1]Under the 802.16, fixed-size MAC (medium access control) frames repeat over time.

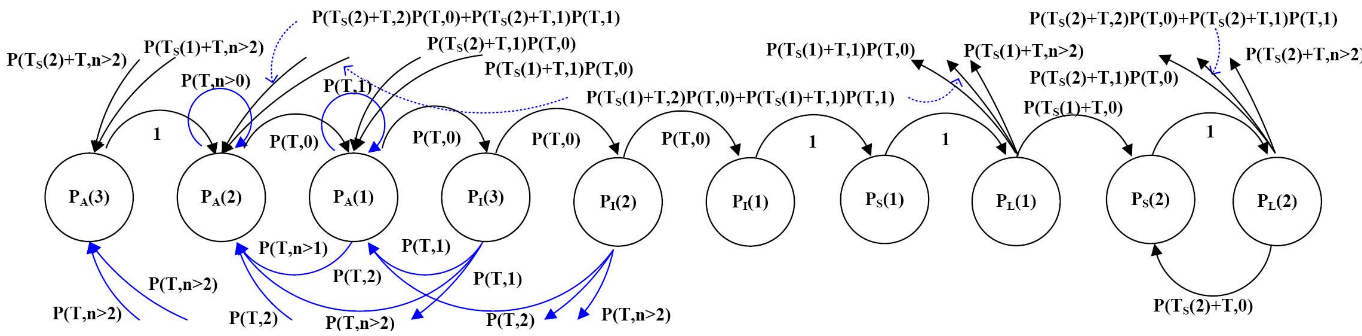


Fig. 2. An example of state transition diagram

For the sleep state case, the state transition is exactly the same as the idle state case except for the last sleep state and the listening interval. When the MSS stays in the last sleep state, it remains in the same state if there is no packet destined to it. The followings are the state transition equations.

$$\begin{aligned} k=1, &\quad P_S(k)=P_I(k)P(T,0)\\ 1<k<M, &\quad P_S(k)=P_L(k-1)P(T_S(k-1)+T,0)\\ k=M, &\quad P_S(k)=P_L(k-1)P(T_S(k-1)+T,0)\\ &\quad\quad +P_L(k)P(T_S(k)+T,0)\\ \\ 1\le k\le M, &\quad P_L(k)=P_S(k) \end{aligned}$$

In addition, the sum of the entire state probability should be one. Using these equations, we can obtain the steady state probabilities by the standard Markov chain analysis. Now, using the derived steady state probabilities, we can derive both delay and power consumption performance metrics.

B. Delay Modeling

The delay consists of *frame delay* and *queuing delay*. The frame delay is caused by the discrete behavior of the Markov chain. The queuing delay is the waiting time to be served in the queue, which is related to the number of the queued packets. Packets which arrive during an active or idle state suffer the same amount of frame delay in average, and the average value is given by $T/2$ which is a half of the frame duration. However, queuing delay is slightly different because the queuing delay in the active state depends on the current queue size. The delay can be expressed as follows.

$$\begin{aligned} D=&\frac{P_A(N-1)}{P_{PA}}P(T,n>0)D_A(N,1)\\ &+\sum_{i=1}^{N-2}\left\{\frac{P_A(i)}{P_{PA}}\left[\sum_{j=1}^{N-i-1}P(T,j)D_A(i,j)\right.\right.\\ &\qquad\left.\left.+P(T,n>N-i-1)D_A(i,N-i)\right]\right\}\\ &+\sum_{i=2}^{Q}\left\{\frac{P_I(i)}{P_{PA}}\left[\sum_{j=1}^{N-1}P(T,j)D_I(j)+P(T,n>N-1)D_I(N)\right]\right\}\\ &+\sum_{i=1}^{M}\left\{\frac{P_L(i)}{P_{PA}}\left[\sum_{j=1}^{N-1}\sum_{k=1}^{j}P(T_S(i)+T,k)P(T,j-k)D_S(i,j)\right.\right.\\ &\qquad\left.\left.+P(T_S(i)+T,n>N-1)D_S(i,N)\right]\right\} \end{aligned}$$

where $D_A(i,j)$, $D_I(j)$ and $D_S(i,j)$ represent the average packet delay of active, idle, and sleep states respectively, and are given by $D_A(i,j)=\frac{1}{j}\sum_{k=1}^{j}\left[(i+k-2)T+\frac{T}{2}\right]$, $D_I(j)=\frac{1}{j}\sum_{k=1}^{j}\left[(k-1)T+\frac{T}{2}\right]$ and $D_S(i,j)=\frac{1}{j}\sum_{k=1}^{j}\left[(k-1)T+\frac{T_S(i)}{2}\right]$, respectively. $T_S(n)$ represents the time duration for which an MSS spends at the n-th sleep state and the conditional packet arrival probability is expressed by $P_{PA}=\sum_{i=1}^{N-1}P_A(i)P(T,n>0)+\sum_{i=1}^{Q}P_I(j)P(T,n>0)+\sum_{k=1}^{M}P_L(k)P(T_S(k)+T,n>0)$.

C. Power Consumption Modeling

We analyze the average power consumption, i.e., the average sum of consumed power at each state. Then, the average power consumption can be modeled simply as follows.

$$P=\frac{\left(\sum_{i=1}^{Q}P_I(i)E_I+\sum_{i=1}^{M}P_A(i)E_A\right)+\sum_{i=i}^{N}(P_S(i)E_ST_S(i)+P_L(i)E_LT)}{\left(\sum_{i=1}^{Q}P_I(i)+\sum_{i=1}^{M}P_A(i)\right)T+\sum_{i=1}^{N}(P_S(i)T_S(i)+P_L(i)T)}$$

where E_I, E_A, E_L and E_S stand for the average power consumption in each of idle, active, listening, and sleep states respectively.

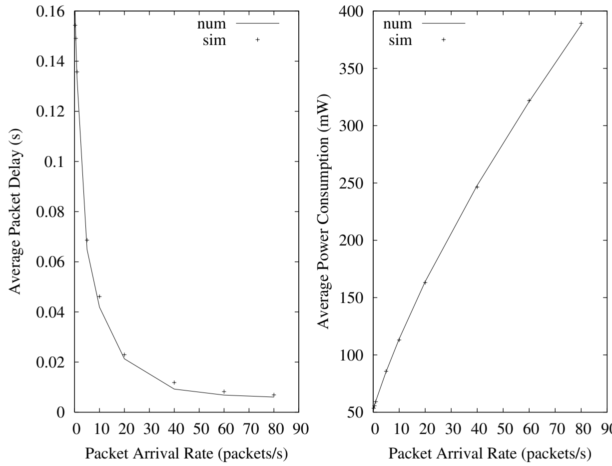
IV. Numerical Evaluation

A. Numerical Modeling Validation

We first validate our analytical model using NS2 simulation. In the simulation, we use the following parameters: $T=5\ ms$, $T_I=10\ ms$, $T_F=160\ ms$, $T_L=5\ ms$ and $T_{TH}=25\ ms$. The packet arrival rate varies according to Poisson distribution with mean λ and the maximum active state is 20. Fig. 3 shows the analytical results and the simulation results of average delay and average power consumption where both are pretty well matched under a reasonable error margin.

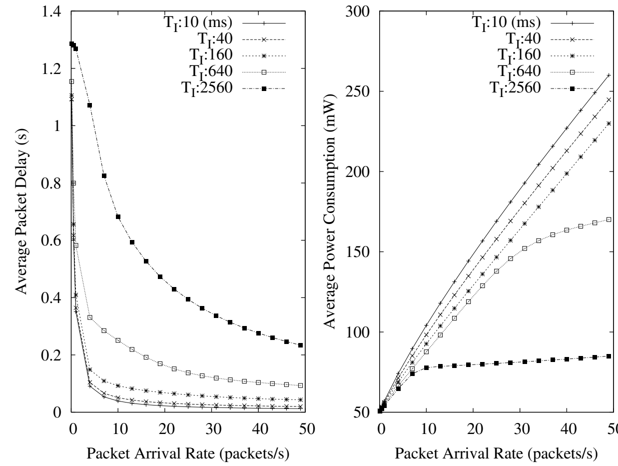
B. The Effects of Operational Parameters

1) Final Sleep Window: Fig. 4 shows the average packet delay (D) and the average power consumption (P), respectively, as λ increases for five different T_F (i.e., 10, 40, 160, 640 and 2560 ms). In order to ignore other parameter effects, we



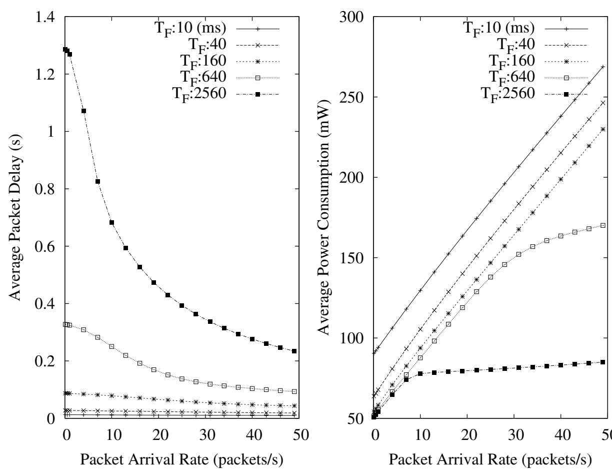
(a) Average delay (b) Average power consumption

Fig. 3. Numerical evaluation vs. simulation



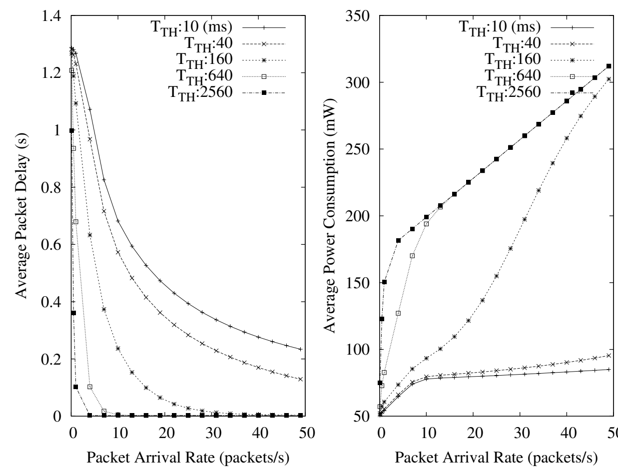
(a) Average delay (b) Average power consumption

Fig. 5. The effects of initial sleep window



(a) Average delay (b) Average power consumption

Fig. 4. The effects of final sleep window



(a) Average delay (b) Average power consumption

Fig. 6. The effect of idle frame threshold

have T_{TH} of 10 ms and T_I of T_F for all T_F. In addition, we borrow the power consumption values from the IEEE 802.11 WLAN because we do not have any power consumption information of IEEE 802.16e currently. Following [7], we assume the following power consumption values in each state: $E_A = 750\ mW$, $E_S = 50\ mW$, and $E_I = E_L = 170\ mW$. 170 mW is determined from the fact that, when we employ the IEEE 802.16e system operating in OFDMA/TDD mode with 5 ms frame, 10 MHz bandwidth, one symbol preamble and 6 symbols map message, the frame overhead for broadcast including the traffic indication message is about 17 percent. Fig. 4(a) shows that D increases proportionally as T_F increases, and it converges to approximately a half of T_F in the low λ region. On the other hand, P decreases as T_F increases. This can be explained that MSS sleeps most of time in the low λ region and sleeps longer relatively if the final sleep window is bigger.

2) Initial Sleep Window: In Fig. 5, we show with the effect of T_I with T_{TH} of 10 ms and T_F of 160 ms. A different T_I means that the granularity of the sleep window growth is different. The smaller T_I is, the more frequent the interval receiving TIM message if T_F is fixed. As T_I increases, D increases and P decreases for the entire λ region. From the both results, we can catch that a smaller T_I can react faster to the change of λ.

3) Idle Frame Threshold: In Fig. 6, the effects of the different T_{TH} values are shown. Both T_F and T_I are set to 160 ms. Note that T_{TH} is the decision value whether the MSS enters the sleep mode or not. As T_{TH} increases, D is shorter, because the decision sensitivity increases and then the chance to enter the sleep mode decreases. For example, if we increase T_{TH} infinitely the system never enters the sleep mode and D is just a half of T. On the other hand, as T_{TH} decreases, P starts to decrease from a small λ.

From the above results, we learn two things. First, D and P have a tradeoff relationship. That is, we cannot improve both of them at the same time. Two, each parameter affects the different λ region. T_F affects the low λ region. On the other hand, the idle window and the initial sleep window give an effect to the MSS which is not in the saturation or idle. Based on this result, we can use these parameters to optimize the system performance when we know the traffic information and the optimization goal.

C. Choosing Appropriate Operational Parameters

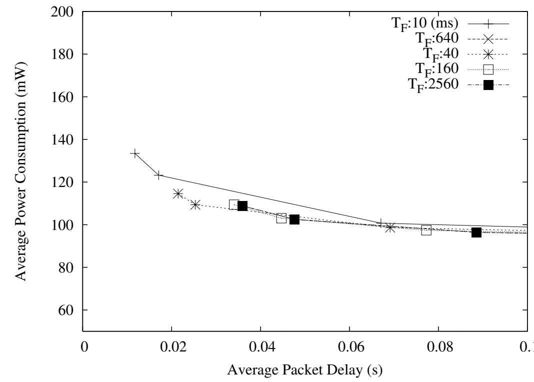
Based on the above results, one can choose appropriate system parameters to optimize the performance by considering both the average delay and the power consumption when the traffic pattern is given. Since the packet arrival rate is the only traffic information for our traffic model, we evaluate the relationship between average delay (D) and average power consumption (P) with a given $\lambda = 10$, and the results are shown in Fig. 7. In Fig. 7(a), the points with the same shape have the same T_F and different T_I values. For each point, T_F and T_I can have one value out of 10, 40, 160, 640 and 2560 ms, and the left-most point corresponds to $T_F = 10$ and $T_I = 10$. For all the cases, T_{TH} value is fixed at 10 ms. In Fig. 7(b), we change T_{TH} values based on the result in Fig. 7(a). Points on the same line have the the same T_F and T_I values while T_{TH} value varies out of 10, 40, 160, 640 and 2560 ms. The result shows that for the same T_F and T_I values, if the T_{TH} increases, D decreases and P increases.

Now, using this result, one can choose some proper points, i.e., the combinations of T_F, T_I, and T_{TH}, if we have an optimization goal. We here consider three different optimization goals.

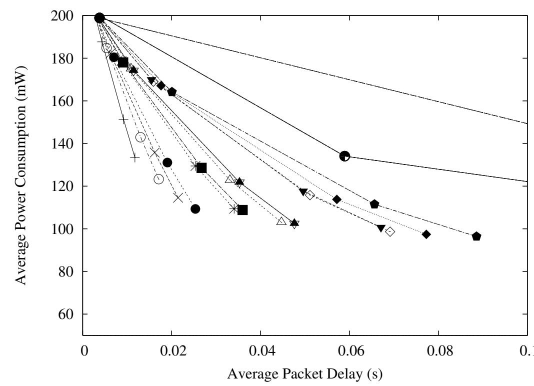
1) If we want to restrict D and P within 0.03 s and 150 mW, respectively, we can choose two operational parameter sets, i.e., $\{T_F = 40,\ T_I = 10,\ T_{TH} = 40\}$ and $\{T_F = 40,\ T_I = 40,\ T_{TH} = 40\}$. In this case, if there is no further restriction, we can use arbitrarily any parameter set out of these two.
2) If we want to restrict only D within 0.03 s (e.g., real-time traffic), we can find over 20 sets in the achievable region. In such a case, we can select the operational parameter set which can achieve additionally the minimum average power consumption in the achievable region, i.e., $\{T_F = 40,\ T_I = 40,\ T_{TH} = 40\}$.
3) On the other hand, for the best-effort traffic case, D is relatively less critical. In such as case, we can limit P within a threshold. For example, if the threshold is 120 mW, about 14 set can achieve this purpose. Now, if we choose $\{T_F = 40,\ T_I = 10,\ T_{TH} = 40\}$, D can be also minimized simultaneously.

V. Conclusion

In this paper, we analyze and evaluate the *sleep mode* operation of IEEE 802.16e using semi-Markov chain. From this, we derive the average delay and the average power consumption under the *sleep mode* operation. After that, we show the effect of the parameters and how to choose the appropriate system parameters when the traffic information is given. In this work, we obtain the following results: (1) numerical and simulation results well match; (2) each operational parameters impose different effects to average delay (D) and average power consumption (P); (3) D and P have a tradeoff relationship; (4) one can get appropriate parameters when the traffic pattern is given.



(a) Varying initial sleep window



(b) Varying idle frame threshold

Fig. 7. Combination of system parameters

Acknowledgment

This research was in part supported by University IT Research Center (ITRC).

References

[1] IEEE 802.16-2004, *Part 16: Air Interface for Fixed Broadband Wireless Access Systems*, Draft Standard for Local and metropolitan area networks, June 2004.

[2] IEEE 802.16e/D12-2005, *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*, October 2005.

[3] IEEE, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, Reference number ISO/IEC 8802-11:1999(E), IEEE Std 802.11, 1999.

[4] J. C. Chen, K. M. Sivalingam, and P. Agrawal “Performance comparison of battery power consumption in wireless multiple access protocols,” *ACM Wireless Networks*, vol. 5, no. 6, pp. 445–460, 1999.

[5] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. C. Chen, “A survey of energy efficient network protocols for wireless networks,” *ACM Wireless networks*, vol. 7, no. 4, pp. 343–358, July 2001.

[6] Shun-Ren Yang and Yi-Bing Lin, “Modeling UMTS discontinuous reception mechanism,” *IEEE Transactions on Wireless Communications*, vol. 4, issue 1, pp. 312–319, Jan. 2005.

[7] Ronny Krashinsky and Hari Balakrishnan,“Minimizing Energy for Wireless Web Access with Bounded Slowdown,” in *Proc. ACM MobiCom’02*, Atlanta, GA, Sep. 2002, pp. 119-130.