An Adaptive Power Saving Mechanism in IEEE 802.11 Wireless IP Networks

Sangheon Pack and Yanghee Choi

Abstract: **Reducing energy consumption in mobile hosts (MHs) is one of the most critical issues in wireles/mobile networks. IP paging protocol at network layer and power saving mechanism (PSM) at link layer are two core technologies to reduce the energy consumption of MHs. First, we investigate the energy efficiency of the current IEEE 802.11 power saving mechanism (PSM) when IP paging protocol is deployed over IEEE 802.11 networks. The result reveal that the current IEEE 802.11 PSM with a fixed wake-up interval (i.e., the static PSM) exhibits a degraded performance when it is integrated with IP paging protocol. Therefore, we propose an adaptive power saving mechanism in IEEE 802.11-based wireless IP networks. Unlike the static PSM, the adaptive PSM adjusts the wake-up interval adaptively depending on the session activity at IP layer. Specifically, the MH estimates the idle periods for incoming sessions based on the exponentially weighted moving average (EWMA) scheme and sets its wake-up interval dynamically by considering the estimated idle period and paging delay bound. For performance evaluation, we have conducted comprehensive simulations and compared the total cost and energy consumption, which are incurred in IP paging protocol in conjunction with various power saving mechanisms: The static PSM, the adaptive PSM, and the optimum PSM. Simulation results show that the adaptive PSM provides a closer performance to the optimum PSM than the static PSM.**

Index Terms: **Adaptive power saving mechanism, dynamic wake-up interval, IEEE 802.11, IP paging protocol, performance evaluation.**

I. INTRODUCTION

In wireless/mobile networks, since mobile hosts (MHs) are free to change their attachments to the networks, location management scheme is required to provide MHs with continuous mobile services. The location management scheme is divided into two procedures: *Location update* and *terminal paging*. Location update is an operation wherein a MH informs the network of its current location. On the other hand, terminal paging is a procedure that determines the exact location of the MH. In the current cellular networks such as GSM and IS-95, location management is supported by the specialized protocol suite for each network standard [1]. Hence, it is difficult to apply a location management protocol used in a network to another network. Furthermore, the next-generation wireless/mobile networks will be heterogeneous networks that various access networks are incorporated. Therefore, it is necessary to design a unified location management protocol, which can be adapted to any types of wireless access networks. To solve these problems, IP (Internet protocol)-based location management schemes were proposed by Internet engineering task force (IETF). In terms of location update, mobile IP [2] and its variants were proposed and several IP paging protocols [3]–[5] were proposed to support terminal paging in IP layer.

The main advantage of the terminal paging is to minimize energy consumption of MHs by avoiding frequent location update procedures [3], [6]. However, it is impossible to obtain a suitable power saving gain by a simple state transition in IP layer without any help of power saving mechanisms supported in lower layers [6]. Hence, when IP paging protocols are deployed in IEEE 802.11 wireless networks, they should be integrated with power saving mechanism (PSM) supported in the IEEE 802.11 standard [7]. In the IEEE 802.11 PSM, a MH goes to sleep state when it is not actively sending or receiving data packets. The MH wakes up periodically every fixed time interval to check whether there are some packets destining it.

However, the existing IEEE 802.11 PSM with the fixed wake-up interval shows a low performance in terms of energy consumption [8]. For example, if the wake-up interval is too short, a MH may wake up too frequently even though there are no incoming packets. On the other hand, too long wake-up interval results in a longer packet reception delay. In [9], the performance of IP paging protocol deployed in IEEE 802.11 networks were investigated. In [9], Pack *et al.* have suggested a simple analytical model and investigated the effects of the session arrival rate and the wake-up interval for both delay-sensitive and delay-insensitive sessions. These results indicate that it is necessary to find the optimal wake-up interval required to minimize MH's energy consumption, while satisfying the given paging delay constraints [10].

To improve the performance of the current IEEE 802.11 PSM, a number of studies have been conducted in the literature [8], [11], [12], and [13].

Recently, Krashinsky *et al.* proposed the bounded slowdown (BSD) protocol to overcome long delay problems occurred when the IEEE 802.11 PSM is used for web-like transfers [8]. In the BSD protocol, the wake-up interval is dynamically adapted to network activity. BSD is an optimal solution to the problem of minimizing energy consumption while guaranteeing that a connection's round trip time (RTT) does not increase by more than a factor p over its base RTT, where p is a protocol parameter that exposes the trade-off between minimizing energy and reducing latency. However, since the BSD protocol was designed for web-like transfers, it is difficult to apply BSD for general IP data sessions, which may have different traffic characteristics from web-like transfers.

Manuscript received February 14, 2005.

The authors are with the School of Computer Science and Engineering, Seoul National University, Korea, email: shpack@mmlab.snu.ac.kr, yhchoi@snu.ac.kr.

The preliminary version of this paper was presented at IEEE International Workshop on Mobile Multimedia Communications (MoMuC) 2003, Munich, Germany, October 2003.

In [11], Anand *et al.* proposed the self-tuning power management (STPM) that adapts its behavior to the access patterns of the network interface and the energy usage of the platform. Compared to the current IEEE 802.11 PSM, STPM reduces the total energy usage of a MH running the Coda distributed file system [14] by 21% while also reducing interactive file system delay by 80%. Furthermore, STPM adapts to diverse operating environments (e.g., both laptops and handhelds) with substantially different characteristics, and performs well across a range of application network access patterns. However, STPM was validated only for distributed file systems, streaming audio, and thin-client applications rather than general Internet applications.

In [12], Zheng *et al.* analyzed the characteristics of energy consumption, delay, and loss rate of the IEEE 802.11 PSM as a function of the traffic load, buffer size, and other protocol specific parameters. The analytical results reveal that the IEEE 802.11 PSM is not very energy efficient for light traffic load. On the other hand, a dynamic PSM (DPSM) in the IEEE 802.11 was introduced by Jung *et al.* in [13]. DPSM dynamically adjusts the size of the fixed time interval called *ad-hoc traffic indication map (ATIM) window*. In the IEEE PSM with ad-hoc mode, each MH must stay awake for ATIM window at the start of each beacon interval. The ATIM window is used to announce any packets pending transmission to nodes in doze state. However, these two works [12], [13] consider an ad-hoc mode operation not infrastructure mode operation. Note that IP paging protocol will be used in infrastructure-based wireless networks, so that we focus on infrastructure mode.

Intuitively, the ideal power save mechanism should allow a MH to sleep continuously during all of the idle period and awake the MH just before the arrival of the paging request message (e.g., the first packet of a session). The ideal power save mechanism, however, is not feasible because it is impossible to predict the time when a paging request message will arrive at paging agent (PA) [15] in future. Therefore, it is necessary to estimate the length of the idle period in order to find the optimal wake-up interval minimizing energy consumption.

In this paper, we propose an adaptive power saving mechanism extending the current IEEE 802.11 PSM. The adaptive PSM adjusts the sleep period in the IEEE 802.11 PSM adaptively, depending on the traffic patterns. Namely, a MH uses a constant, called *adjustment constant*, in order to adjust the wake-up interval during the idle period. The idle period length is estimated by a simple estimation scheme using the previous traffic patterns. To determine the adjustment constant, we formulate the optimal wake-up interval selection problem and present its solution.

The remainder of this paper is organized as follows. Section II introduces the power saving mechanism defined in the IEEE 802.11 standard. Section III describes an adaptive power saving mechanism to adjust the wake-up interval during the idle period. In Section IV, we formulate wake-up and paging delay costs. In addition, we propose a problem to find out the optimal wake-up interval satisfying the given paging delay constraint and introduce an estimation scheme for the idle period using the previous traffic patterns in Section V. Section VI shows simulation results and Section VII finally concludes this paper.

II. OVERVIEW: IEEE 802. 11 POWER SAVING MECHANISM

In wireless/mobile communication, it is preferable to turn MHs to lower power consuming power saving (PS) mode, while they are not actively communicating, and wake them up just before when data packets destined for them arrive at the networks. However, since MHs in PS mode cannot know when packets arrive for them, it is impossible to follow this rule exactly. Therefore, the IEEE 802.11 standard supports the PSM as follows.

First of all, the IEEE 802.11 PSM can be operated in infrastructure mode or ad-hoc mode. The detailed operations of two modes are quite different. Since IP paging protocols are deployed in infrastructure networks, this work focuses on the power saving mechanism in infrastructure mode. In infrastructure mode, there is an access point (AP) to monitor current mode of each MH. The IEEE 802.11 PSM supports two types of modes: *Active* and *power saving (PS)* modes. Compared with PS mode, a MH in active mode is fully powered and thus may send and receive packets at any time, while the MH in active mode consumes extremely higher energy.

On the other hand, a MH in PS mode only wakes up periodically to check whether there are any incoming packets from the AP. The MH always notifies its AP when changing modes. Periodically, the AP transmits beacon frames spaced by a fixed beacon interval (BI). An MH in PS mode should monitor these frames. Once every beacon interval, which is typically set to 100 msec, the AP sends a beacon frame containing a traffic indication map (TIM), which contains the identifiers (ID) of those MHs in PS mode for which there are buffered unicast packets waiting in the AP. Upon hearing its ID, a MH in PS mode should stay awake for the remaining beacon interval.

During the contention period (i.e., *distributed coordination function (DCF)*), an awakened MH in PS mode can issue a PS-POLL to the AP to retrieve the buffered packets. On the other hand, during the contention-free period (i.e., *point coordination function (PCF)*), a MH in PS mode waits for the AP to poll it. Typically, a MH listens to every beacon, but the MH can also be configured to skip several beacons between listen times by setting the listen interval. Spaced by a fixed number of beacon intervals, the AP will send deliver TIMs (DTIMs) within beacon frames to indicate the presence of buffered broadcast packets. Immediately after the DTIMs, the buffered broadcast packets will be sent. Whenever the AP sends data to a MH, it indicates whether or not there are more outstanding data, and the MH goes to sleep only when it has retrieved all pending data from the AP. If the MH has data to send, it can wake up in order to send the data without waiting for a beacon frame.

Power saving mechanism in ad-hoc mode, where the packet store and forward and the timing synchronization has to be done in a distributed manner, is more complex. Details of the power saving protocol used in ad-hoc networks can be found in [16].

III. ADAPTIVE POWER SAVING MECHANISM

The adaptive power saving mechanism is based on the IEEE 802.11 PSM. However, in the adaptive power saving mechanism, the wake-up interval during the idle period is adjusted as ρ (i.e., adjustment constant) times of beacon interval, which is

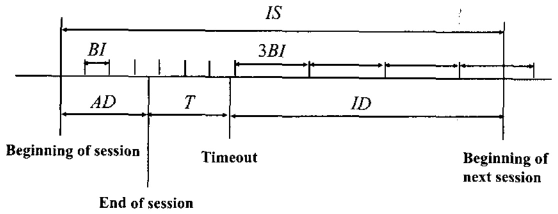


Fig. 1. Timing diagram in the IEEE 802.11 PSM with the adaptive wake-up interval ($\rho = 3$).

usually set to 100 ms. ρ is an integer value to adaptively adjust the length of the wake-up interval. Fig. 1 shows a timing diagram in the adaptive power saving mechanism with an adjustment constant of 3. Let AD and ID be the session active duration time and the idle duration time, respectively. IS and BI denote the inter-session arrival time and the beacon interval, respectively. T is the active timer value to separate data sessions in connectionless IP networks [3]. The operation from the beginning of a session to the expiration of the active timer is the same as that of the IEEE 802.11 PSM. In other words, a MH wakes up every beacon interval before the active timer expiration. However, the MH determines the optimal adjustment constant (ρ^*) when the idle period begins (at τ_0), and it wakes up only per ρBI during the idle period. If a paging request message is arrived at τ_1, the idle period is ended and the time period from τ_1 to the next wake-up epoch (τ_2) becomes the paging delay (D_p).

Intuitively, there is a trade-off relationship between the wake-up cost ($C_{\text{wake-up}}$) and the paging delay cost (C_{delay}). Namely, as the wake-up interval increases, the paging delay also increases whereas the wake-up cost decreases. On the other hand, as the wake-up interval decreases, the paging delay decreases and the wake-up cost increases. Based on the relationship, we formulate the optimal wake-up interval selection problem as follows. We assume that each duration time (i.e., AD, T, and ID) is synchronized with the beacon interval of the IEEE 802.11 specification [7]. P_{th} is a threshold value for the paging blocking probability (P_B). The paging blocking probability refers to the probability that a paging request is failed due to the long paging delay. How to calculate $C_{\text{wake-up}}$, C_{delay}, and P_B will be elaborated in Section IV.

$$\begin{aligned} &\text{Minimize} \quad C_{\text{total}} = C_{\text{wake-up}} + C_{\text{delay}}, \\ &\text{such that} \quad P_B \le P_{th}. \end{aligned}$$

IV. ANALYTICAL MODELING

As mentioned before, we define two types of costs: *Wake-up cost* and *paging delay cost*. In the IEEE 802.11 PSM, a MH in sleep mode wakes up periodically to check whether there are some incoming packets. In the current IEEE 802.11 PSM, the wake-up interval is the same as the beacon interval (BI). However, when there exists a long idle period, waking up every beacon interval is unnecessary. The wake-up cost ($C_{\text{wake-up}}$) is formulated by considering the effect of the wake-up interval. We assume that the minimum wake-up interval is the beacon interval.

In the IEEE 802.11 PSM, since a MH wakes up only at the beginning of the wake-up interval, if a paging request packet arrives during the sleep period then the MH can receive the paging request packet only at the next wake-up time. In other words, when IP paging is utilized over IEEE 802.11 wireless networks supporting the PSM, the paging delay may occur. To represent this paging delay, we define the paging delay cost (C_{delay}). Also, if the delay associated with a paging request packet is longer than the given delay constraint (D_{const}), the session may become blocked. Therefore, the paging delay is related to the session blocking probability (P_B).

The total cost is the sum of the wake-up cost and the paging delay cost incurred during the inter-session time, as shown in

$$C_{\text{total}} = C_{\text{delay}} + C_{\text{wake-up}}. \tag{1}$$

The wake-up cost ($C_{\text{wake-up}}$) is proportional to the number wake-ups which occur during the inter-session time.

$$C_{\text{wake-up}} = \alpha \cdot K, \tag{2}$$

where K denotes the number of wake-up events and α is a weighting factor for a wake-up event.

On the other hand, the paging delay cost, C_{delay} is related to the arrival time of a paging request packet during a wake-up interval. The paging delay may vary depending on the length of the wake-up interval. In this paper, we assume that one-step uniform paging scheme [10] is used for terminal paging. Therefore, to locate a MH in a paging area, only one paging procedure is performed. In (3), D denotes the paging delay and β is a weighting factor used to take into account the sensitivity of session. For example, β of a session, which requires a strict session setup delay bound (e.g., instant message applications), is larger than that of a session in elastic applications. The effect of β has been investigated in [9].

$$C_{\text{delay}} = \beta \cdot D. \tag{3}$$

A. IEEE 802.11 PSM with Fixed Wake-up Interval

First, we formulate the wake-up cost and paging delay cost corresponding to the IEEE 802.11 PSM with the fixed wake-up interval. Fig. 2 shows a timing diagram when the IEEE 802.11 PSM is associated with the fixed interval. The meaning of each notations was presented in Section III. Then, the paging delay can be calculated as follows

$$D = \left\lceil \frac{ID}{BI} \right\rceil \cdot BI - ID. \tag{4}$$

In the case of the IEEE 802.11 PSM with the fixed interval, although a MH is in idle state for a long time, the MH wakes up every beacon interval. Therefore, the number of wake-ups during inter-session time is given by

$$K = \left\lfloor \frac{IS}{BI} \right\rfloor. \tag{5}$$

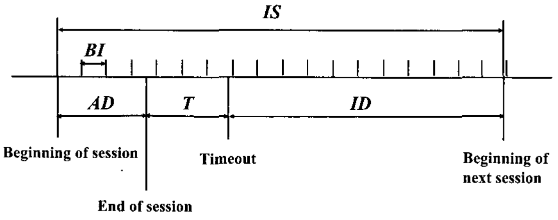


Fig. 2. Timing diagram in the IEEE 802.11 PSM with the fixed wake-up interval.

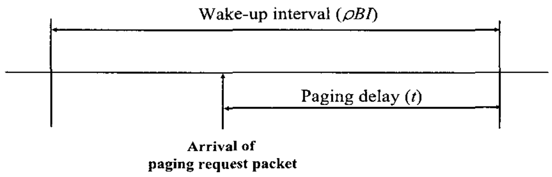


Fig. 3. Uniform arrival pattern of the paging request packet.

B. IEEE 802.11 PSM with Adaptive Wake-up Interval

Second, we consider the power saving mechanism which allows a larger wake-up interval during an idle period. According to the specification of the IEEE 802.11, the wake-up interval can be adjusted by applying *listen interval* during the idle period. When the listen interval is set to ρ times of the beacon interval, the MH wakes up only every ρBI without any intermediate wake-ups. Fig. 1 shows the timing diagram in the IEEE 802.11 PSM with an adaptive wake-up interval. ρ is an adjustment constant, which is larger than 1. Then, the paging delay in the adaptive wake-up interval is given by

$$D = \left\lceil \frac{ID}{\rho BI} \right\rceil \cdot \rho BI - ID. \tag{6}$$

In the IEEE 802.11 PSM with the adaptive wake-up interval, the original wake-up interval (i.e., beacon interval) is used before the expiration of the active timer. After the expiration of the active timer, a MH switches from active state to idle state. During the idle period, the MH wakes up only every ρBI, because the MH usually remains in the idle state for a long time. In other words, the wake-up interval in idle state is ρBI. Therefore, the total number of wake-ups during inter-session time can be represented by

$$K = \left\lfloor \frac{AD + T}{BI} \right\rfloor + \left\lfloor \frac{ID}{\rho BI} \right\rfloor. \tag{7}$$

C. Session Blocking Probability

In wireless/mobile networks supporting the paging function, the paging delay affects the session blocking probability, which is one of the most important QoS related performance factors. If a paging request packet is delayed more than the given delay constraint, the session will be blocked. In IP paging protocols deployed over IEEE 802.11 networks, the paging delay is related to the wake-up interval used in the IEEE 802.11 PSM. Fig. 3 shows a timing diagram corresponding to the case where a paging request packet arrives during a wake-up interval.

If we assume that a paging request packet arrives at the paging agent in a uniformly distributed manner during a wake-up interval, the probability density function of the paging delay is $1/\rho BI$ (note that ρ is 1 for the IEEE PSM with the fixed wake-up interval). If the delay of a paging request is larger than the delay constraint (D_{const}), the session becomes blocked. Therefore, the session blocking probability (P_B), is calculated as follows

$$P_B = \Pr(t > D_{\text{const}}) = \int_0^{\rho BI - D_{\text{const}}} \frac{1}{\rho BI} dt, \tag{8}$$

where t is a random variable that refers to the paging delay.

Fig. 4 depicts the session blocking probability as a function of the adjustment constant. Intuitively, it would be expected that as the delay constraint decreases, the session blocking probability would increase. Using Fig. 4, we can find the largest adjustment constant which keeps the blocking probability below a specific threshold blocking probability. For example, let's assume that the delay constraint is 100 msec and the upper bound of the blocking probability is 0.8. In this case, if the adjust constant is 5, then the session blocking probability is 0.8. On the other hand, the session blocking probability is 0.83 when the adjustment constant is 6. Therefore, the adjustment constant should be equal to or less than 5 in order to satisfy the given delay constraint. As mentioned before, the session blocking probability is one of the most important factors in wireless/mobile networks, because it determines users' quality of service and the buffer requirement at the paging agent. However, the result shown in Fig. 4 does not consider the total cost, i.e., it does not include the wake-up cost. Therefore, it is necessary to find the optimal adjustment constant to minimize the total cost while satisfying the given delay constraint, which will be elaborated in the next section.

V. NEAR OPTIMAL WAKE-UP INTERVAL SELECTION

It is simple to find the optimal adjustment constant in the problem formulated in Section III if the idle duration time (ID) is given. First, using the threshold blocking probability (P_{th}), we can obtain a candidate set of adjustment constants that satisfy the condition, $P_B \leq P_{th}$. After then, we can find an optimal ρ, that minimizes the total cost, by substituting ρ in the total cost function with each ρ in the candidate set. In other words, since the total cost is a function of ρ and ID in the selection problem (refer to Section IV), we should know the length of the idle duration time in order to find the optimal wake-up interval. However, it is impossible to determine how long a MH stays in idle state at the beginning of the idle period. Therefore, we propose a simple estimation scheme for the idle period length and a procedure to find a near optimal adjustment constant (ρ_n) minimizing the total cost when there is no information about the exact idle period length.

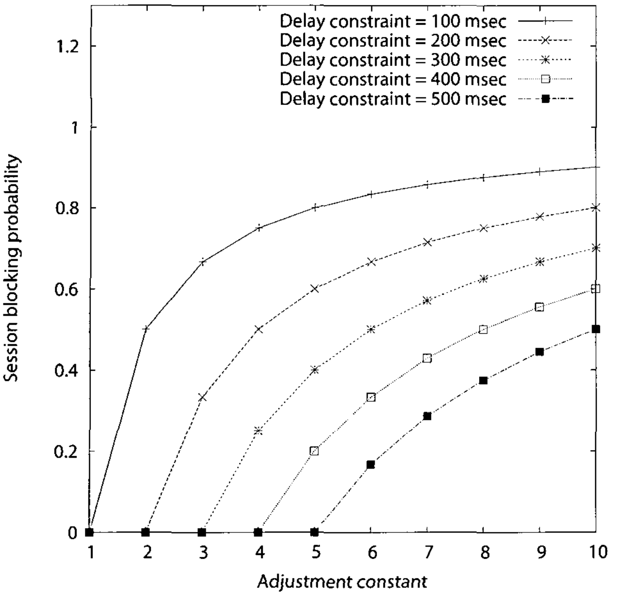


Fig. 4. Session blocking probability distribution.

A. A Simple Estimation Scheme for Idle Period

The proposed estimation scheme uses the previous communication patterns of the MH in order to estimate the idle period length. Let ID_{init} be an initial estimated length of the idle period. $ID_e(k)$ and $ID(k)$ denote the k-th estimated idle period length and the k-th measured idle period length, respectively. Using the initial value and the previous idle period lengths, we estimate the next idle period length as the below equation, which is based on the *exponentially weighted moving average (EWMA)* scheme. ω is a weighting factor ($0 < \omega < 1$). $E_{k-1}(ID)$ is the average idle period length from 0-th to the $(k-1)$-th idle duration time.

$$i)\ k = 1,\ ID_e(1) = ID_{\text{init}}$$
$$ii)\ k \geq 2,\ ID_e(k) = \omega \cdot E_{k-1}(ID) + (1-\omega) \cdot ID(k-1),$$

$$\text{where } E_{k-1}(ID) = \left(\sum_{i=1}^{k-1} ID(i)\right) / (k-1).$$

B. Procedure to Find Near Optimal Wake-up Interval

Using the estimation scheme for the idle period length, we can find a near optimal adjustment constant (ρ_n). Following pseudocode shows the algorithm to find ρ_n. The weight value (ω) and the initial value (ID_{init}) are given as input parameters. T_{start} and T_{end} are variables to record the start time and end time of the idle period, respectively.

```
Algorithm  FindNearOptimalWakeupInterval ($\omega$, $ID_{\text{init}}$)
k := 1;
while(true){
    After the active timer expiration {
        $T_{\text{start}}$ := Current time;
        If(k = 1)  $ID_e(k) := ID_{\text{init}}$;
        elseif(k > 2)
            $ID_e(k) := \omega \cdot E_{k-1}(ID) + (1-\omega) \cdot ID(k-1)$;
    }
    Find $\rho_n$ to minimize $C(\rho, ID_e(k))$
    The MH wakes up every $\rho_n \cdot BI$ interval
    if(Arrival of paging request message){
        $T_{\text{end}}$ := Current time;
        $ID(k) := T_{\text{end}} - T_{\text{start}}$;
        k++;
    }
}
```

Fig. 5. A procedure for finding near optimal wake-up interval.

This algorithm, which is invoked when the active timer expires, is implemented in between IP layer and MAC layer. Of course, various prediction schemes can be applied in order to estimate the idle period length [17]. However, since these prediction schemes should be able to be implemented in network interface cards (NIC) with the limited computation capability, the schemes should be as simple as possible. The proposed scheme based on the exponentially weighted moving average algorithm does not require any complex operations. Hence, the proposed scheme is a more appropriate solution than any other prediction schemes in terms of easy implementation. However, the performance of the EWMA scheme may not be satisfied under heterogeneous traffic environments. Even if our simulations have been conducted over reasonable traffic models, more efficient estimation schemes should be investigated.

VI. PERFORMANCE EVALUATION

To evaluate the adaptive power saving mechanism, we performed comprehensive simulations in terms of total cost and energy consumption. In simulations, we compare our adaptive power saving mechanism with the ideal power saving mechanism and current power saving mechanism with the fixed wake-up interval.

A. Simulation Environment

To investigate the effectiveness of the adaptive power saving mechanism, we have simulated and calculated the total cost and energy consumption for 100 data sessions. In this simulation, we assume that the session arrival process follows a Poisson distribution with rate of λ. Therefore, the inter-session time (IS) follows an exponential distribution with mean of $1/\lambda$. In general, the session active duration time of an IP session does not follow an exponential distribution. Therefore, it is assumed that the session active duration time follows a Pareto distribution with shape parameter α and scaling parameter k, which determines the minimum value [18]. The Pareto distribution has a heavy-tailed property reflecting the characteristics of IP sessions. According to previous works [19], [20], it was proved that a data session in wireless networks has similar characteristics to a data session in wired networks. This is because mobile users usually use similar types of applications to those in wired networks (e.g., web browsing, email, and so on). (9) and (10) show the probability density functions (PDFs) of the inter-session time and the session active duration time, respectively.

$$f_{IS}(t) = \lambda e^{-\lambda t}, \tag{9}$$

$$f_{AD}(t) = \frac{a}{k}\left(\frac{k}{t}\right)^{a+1}. \tag{10}$$

In terms of the distribution of the session active duration time, a is set to 0.78 and k is set to 10 (short session active duration time) or 30 (long session active duration time), based on the reference values in [20]. In addition, P_{th} is set to 1 % in our simulations. To evaluate the adaptive power saving mechanism in more different environments, we calculated the wake-up cost and paging delay cost when the session arrival rate is changed. The low and high session arrival rates are set to 10 sessions/hour and 30 sessions/hour, respectively. In this simulation, the weight values, α and β are set to 0.01 and 0.01, respectively, and the paging delay constraint is set to 1000 msec.

In the adaptive PSM, we used the exponentially weighted moving average scheme to estimate the next idle period length. To predict the idle period length more exactly, the initial value and the weight value should be carefully determined. In addition, sufficient previous patterns are required. In our simulation, the initial idle period length is set to the predefined paging delay constraint value (D_{const}) to avoid unnecessary paging delay. In terms of weight value, a set of weight values (i.e., 0.8, 0.6, 0.4, and 0.2) is evaluated and they are denoted by $A1$, $A2$, $A3$, and $A4$, respectively. The active timer is set to 18 sec.

We compared our adaptive power saving mechanism (A) with three different power saving mechanisms.

- *Fixed scheme (F)*: This scheme is based on the current IEEE 802.11 PSM with the fixed wake-up interval. The default wake-up interval is 100 ms, which is the same as the beacon interval.
- *Power optimal scheme (P)*: The goal of the power optimal scheme is to minimize energy consumption, but does not consider any paging delay. Namely, in this scheme, a MH remains in sleep state as long as possible without any intermediate wake-ups.
- *Ideal scheme (I)*: This scheme is an optimal power saving mechanism to minimize power consumption while meeting the paging delay constraint. Unlike power optimal scheme, a MH often wakes up not to violate paging delay constraint. To do this, the wake-up interval of the ideal scheme is set to $\min\{\lfloor ID \rfloor, D_{\text{const}}\}$.

Since P and I assume that a MH knows the future idle period length before entering the sleep state in advance, they cannot be supported in real network system.

For the cost comparison, we defined a performance parameter named cost gain (G_X). The cost gain for X scheme is defined as follows

$$G_X = \frac{\text{The cost of ideal scheme}}{\text{The cost of } X \text{ scheme}}.$$

Since the cost of the ideal scheme is always smaller than those of other schemes, the cost gain is smaller than 1.0. Also, as the cost gain of the scheme X increases, the performance of the scheme X becomes better.

B. Simulation Results

B.1 Effect of Session Arrival Rate

As shown in Figs. 6 and 7, a higher cost is required as the session arrival rate decreases. This is because the low session arrival rate refers to longer inter-session time. If the inter-session time is long, the MH requires a higher cost for wake-up events. As shown in Fig. 6, F shows the largest total cost among various power saving mechanisms. However, as the session arrival rate increases, the total cost gain of F also increases. In other words, the total cost gain of F is 0.31 when the session arrival rate is low. However, the total cost gain becomes 0.61 when the session arrival rate is high. In short, when the session arrival rate is low, the inter-session time is long. Therefore, the fixed wake-up interval results in a higher wake-up cost. However, the inter-session time and the wake-up cost decrease as the session arrival rate increases. Consequently, the total cost gain of F is proportional to the session arrival rate. A similar trends can be observed in P and A.

In both Figs. 6 and 7, P shows the lowest wake-up cost among various power saving mechanisms and I exhibits almost the same wake-up cost as P. However, F shows the highest wake-up cost whereas the adaptive PSMs show less wake-up costs than F. Since the wake-up cost has a larger portion of the total cost than the paging delay cost in most cases, the adaptive PSM also shows a higher total cost gain than F. Specifically, in the case of low session arrival rate, the total cost gain of F is 0.31 and the total cost gains of adaptive PSMs (i.e., $A1$–$A4$) are about 0.82–0.84.

In terms of weight value used in the exponentially weighted moving average scheme, the differences among adaptive power saving mechanisms with different weight values are not significant. This is because the inter-session time is much longer than 1000 msec for most cases whereas the paging delay constraint is set to 1000 msec. This means that the maximum wake-up interval is determined by the paging delay constraint value rather than the estimated idle period length. Consequently, each adaptive PSM shows almost the same cost when the paging delay constraint is equal to 1000 msec.

Although P shows the lowest wake-up cost, P has the highest paging delay cost. Therefore the total cost gain of P is 0.81, which is slightly lower than that of A. As mentioned above, this is because P dose not consider any paging delay constraint so that P sleeps as long as possible during the idle period. In contrast, in the case of F, since a MH wakes up every 100 msec, the paging delay cost is very small and the paging delay cost gain is about 1.0.[1] On the other hand, the total cost gain of the adaptive PSM is about 0.82–0.84, which is a larger value than F and P.

B.2 Effect of Session Duration Time

Figs. 8 and 9 show the cost comparison when the session active duration is changed. As depicted in Figs. 8 and 9, the total cost gains of F, P, and A increase as the average session active duration time increases. As the session active duration time increases, the idle period length and the number of wake-ups are

---

[1] Note that the paging delay constraint used in the simulation is 1000 msec, which is much longer than the beacon interval, 100 msec.

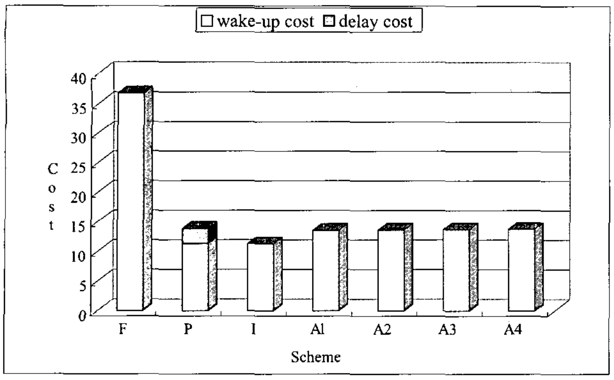


Fig. 6. Effect of session arrival rate – low session arrival rate.

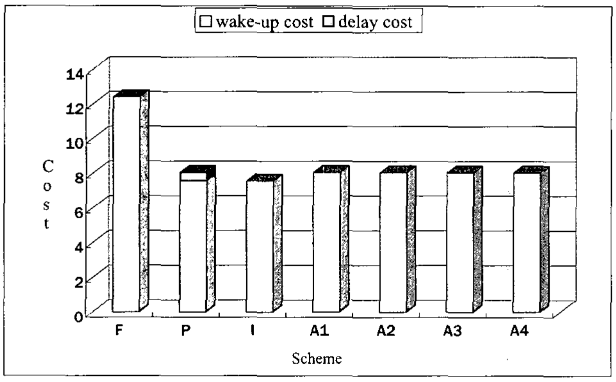


Fig. 7. Effect of session arrival rate – high session arrival rate.

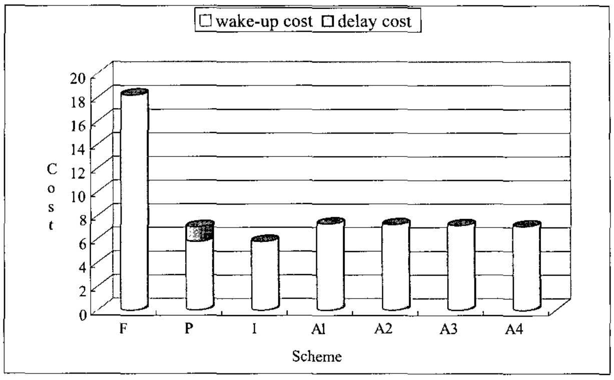


Fig. 8. Effect of session active duration time – short duration time.

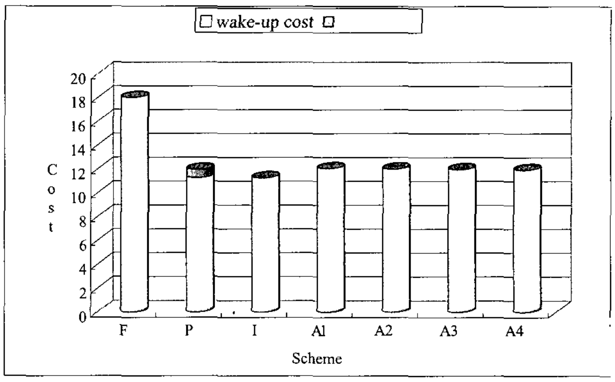


Fig. 9. Effect of session active duration time – long duration time.

Table 1. Two power consumption models for WLAN cards.

State	Model 1 (W)	Model 2(W)
Active	1.5	0.85
Standby	1.15	0.033
Sleep	0.045	0.005

reduced. In other words, since there are less wake-ups during the idle period when the session active duration time is long, the effectiveness obtained by using PS mode is reduced. Because of the same reason, the cost gain of F drastically increases as the active duration time increases, compared with P and A. In short, if the active duration time is long, there is not a sufficient idle period requiring a longer sleeping supported in I and P. In addition, P shows a slightly larger total cost than I and A, because it has a larger paging delay cost. On the other hand, F shows the largest total cost among various PSMs for all cases.

B.3 Energy Consumption

To analyze energy consumption in different PSMs, we use two different power consumption models used in [13] and [21]. The model 1 represents the power consumption model of typical WLAN cards, whereas the model 2 shows the low power consumption model that has been recently developed. The active state refers to a state that a MH actively sends or receives some packets. As shown in Table 1, a MH in the active state consumes 1.5 W and 0.85 W in the models 1 and 2, respectively. In the standby state, a MH does not send or receive any packets but it remains awaken state, so that the MHs in the models 1 and 2 consume 1.15 W and 0.033 W, which is less than power consumption in the active state. On the other hand, a MH in the sleep state does not perform any interactions (e.g., sending or receiving) with other hosts. Thus, a MH in the sleep state consumes an extremely low energy, i.e., 0.045 W and 0.005 W in the models 1 and 2. In the IEEE 802.11 PSM, a MH must stay awake for a period of time (e.g., TIM or DTIM) after each beacon. This time period is typically set to 5 msec.

Fig. 10 shows the energy consumption for high and low session arrival rates (i.e., λ =10 and 30) when the model 1 is used. It is assumed that the initial energy of a MH is 20000 J. In the case of low session arrival rate, the remaining energies after 100 sessions of F, P, and I are 1093.86 J, 2498.82 J, and 2358.85 J, respectively. Since a MH wakes up periodically even though during a long idle period in the current PSM with the fixed interval, it wastes an amount of energy due to frequent wake-ups. On the other hand, in the case of P, a MH stays in the sleep state as long as possible, so that the remaining energy is higher than that of I. In I, since a MH often wakes up not to violate the paging delay constraint, it consumes more energy than P. The remaining energies of $A1$, $A2$, $A3$, and $A4$ are 2516.00 J, 2477.03 J, 2438.01 J, and 2411.92 J, respectively.[2] In other

---

[2] To clarify the presentation, the results of $A2$, $A3$, and $A4$ are omitted in Figs. 10 and 11.

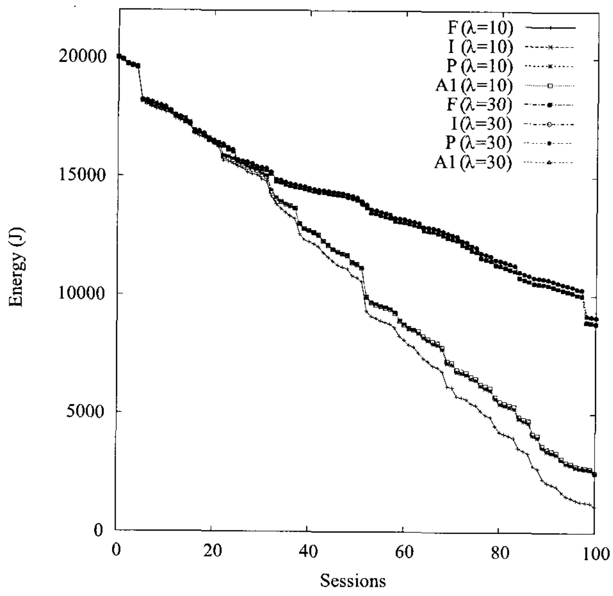


Fig. 10. Effect of session arrival rate.

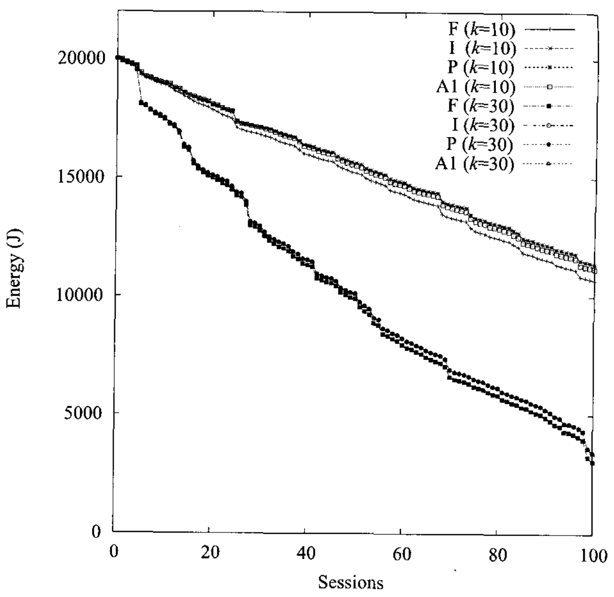


Fig. 11. Effect of session active duration time.

words, $A2$, $A3$, and $A4$ consume more energy than P while less energy than I. Especially, $A1$ consumes the least energy among different schemes. This is because the adaptive PSM sometimes over-estimates the idle duration time. When the idle duration time is over-estimated, a higher paging delay cost occurs even if the energy consumption is lowered. Compared with F, the adaptive PSMs show much lower energy consumption.

When the session arrival rate is high (i.e., $\lambda = 30$), similar results are observed. However, the difference between P (or I) and A (or F) is reduced as the session arrival rate increases. As mentioned above, the inter-session time is inversely proportional to the session arrival rate. Therefore, when the session arrival rate is high, the inter-session time is short and the idle duration time is also short. Therefore, the gains of longer sleeping in P and I are reduced when the session arrival rate is high. Specifically, the difference between P and F decreases from 1404.96 J to 266.96 J as the session arrival rate increase. In addition, the difference between P and A also decreases from 17.17 J $\sim$ 86.89 J to 15.53 J $\sim$ 26.07 J, as the session arrival rate increases.

Fig. 11 depicts the energy consumption pattern in different session active duration times (i.e., k=10 and 30). As shown in Fig. 11, more energy is consumed when the average session active duration time is long. This is because the energy consumption in the active state is much higher than that in the standby or sleep state. Also, the difference between P (or I) and A (or F) is not notable when the average session active duration time is long. This is because the effectiveness of a longer sleeping in I and P is low when the session active duration time is long.

Tables 2 and 3 compare the energy consumption in the models 1 and 2. Since the model 2 consumes much lower energy in the standby and sleep states, the remaining energy is higher than that of the model 1. In addition, the energy saving gain of the adaptive PSM is not remarkable in the model 2. This is because, in the model 2, the difference between power consumptions of the standby and sleep states is not high. However, the low power consumption models are available only for limited interfaces such as secure digital input/output (SDIO) type. Therefore, the adaptive PSM is sufficiently effective in current dominant WLAN cards.

VII. CONCLUSION AND FUTURE WORKS

In this paper, we proposed an adaptive power saving mechanism for wireless IP networks based on the IEEE 802.11. Simulation results show that the adaptive power saving mechanism can significantly reduce the energy consumption compared with the current IEEE 802.11 PSM. In addition, the adaptive PSM outperforms the IEEE 802.11 PSM in terms of the total cost, which consists of the wake-up cost and the paging delay cost. The key scheme to reduce the energy consumption is how to estimate the idle period length. In this study, we used a simple estimation scheme, called exponentially weighted moving average scheme. However, the performance of this scheme is highly dependent on the initial estimated value and the weight value. To enhance the performance of the estimation scheme, many different schemes are available. Especially, Bayes' theorem [22] is widely used to estimate a value based on the previous values. Hence, in the future work, we will propose a new estimation scheme based Bayes' theorem. However, it is required to reduce the computation overhead to allow Bayes' theorem to be implemented in network interface cards. Consequently, we will conduct a comparative study of each estimation scheme.

ACKNOWLEDGMENT

This work was supported in part by the Brain Korea 21 project of the Ministry of Education and in part by the National Research Laboratory project of the Ministry of Science and Technology, 2004, Korea.

REFERENCES

[1] I. F. Akyildiz, J. Ho, and Y. B. Lin, "Movement-based location update and selective paging for PCS networks," *IEEE/ACM Trans. Networking*, vol. 4, no. 4, pp. 629–638, Aug. 1996.

Table 2. Remaining energy percent (%): Model 1.

Case	F	P	I	$A1$	$A2$	$A3$	$A4$
$\lambda = 10$ sessions/hour	5.47	12.49	11.74	12.58	12.39	12.19	12.05
$\lambda = 30$ sessions/hour	44.11	45.45	45.32	45.32	45.33	45.34	45.37
$k = 10$ sec	53.54	56.97	56.63	55.93	56.12	56.31	56.53
$k = 30$ sec	15.31	17.17	16.98	17.01	17.11	17.22	17.35

Table 3. Remaining energy percent (%): Model 2.

Case	F	P	I	$A1$	$A2$	$A3$	$A4$
$\lambda = 10$ sessions/hour	58.41	58.59	58.57	58.65	58.63	58.61	58.60
$\lambda = 30$ sessions/hour	75.12	75.15	75.15	75.18	75.18	75.18	75.18
$k = 10$ sec	82.37	82.46	82.45	82.38	82.40	82.41	82.43
$k = 30$ sec	59.28	59.32	59.32	59.57	59.58	59.59	59.60

[2] IETF mobile IP Working Group, http://www.ietf.org/html.charters/mobileip-charter.html.

[3] X. Zhang, J. Castellanos, and A. Campbell, "P-MIP: Paging extensions for mobile IP," *ACM Mobile Networks Appl.*, vol. 7, no. 2, pp. 127–141, Apr. 2002.

[4] R. Ramjee *et al.*, "IP paging service for mobile hosts," *ACM Wireless Networks*, vol. 8, no. 5, pp. 427–441, Sept. 2002.

[5] C. Castelluccia, "Extending mobile IP with adaptive individual paging," *ACM Mobile Computing Commun. Review*, vol. 5, no. 2, pp. 14–26, Apr. 2001.

[6] V. Kafle, S. Pack, and Y. Choi, "Performance evaluation of IP paging with power saving mechanism," in *Proc. IEEE LCN 2003*, Bonn, Germany, Oct. 2003.

[7] IEEE 802.11b WG, Part 11, *Wireless LAN medium Access Control (MAC) and Physical Layer (PHY) Specification: High-Speed Physical Layer Extension in the 2.4 GHz Band*, IEEE, 1999.

[8] R. Krashinsky and H. Balakrshnan, "Minimizing energy for wireless web access with bounded slowdown," in *Proc. ACM MOBICOM 2002*, Atlanta, USA, Sept. 2002.

[9] S. Pack, V. Kafle, and Y. Choi, "Performance analysis of IP paging protocol in IEEE 802.11 networks," in *Proc. IEEE WLN 2003*, Bonn, Germany, Oct. 2003.

[10] W. Wang, I. F. Akyildiz, and G. L. Stuber, "Effective paging schemes with delay bounds as QoS constraints in wireless systems," *ACM Wireless Networks*, vol. 7, no. 5, pp. 455–466, Sept. 2001.

[11] M. Anand, E. B. Nightingale, and J. Flinn, "Self-tuning wireless network power management," in *Proc. MOBICOM 2003*, San Diego, USA, Sept. 2003.

[12] R. Zheng, J. Hou, and L. Sha, "Performance analysis of the IEEE 802.11 power saving mode," in *Proc. CNDS 2004*, San Diego, USA, Jan. 2004.

[13] E. Jung and N. H. Vaidya, "An energy efficient MAC protocol for wireless LANs," in *Proc. IEEE INFOCOM 2002*, New York, USA, June 2002.

[14] J. Kistler and M. Satyanarayanan, "Disconnected operation in the coda file system," *ACM Trans. Computer Sys.*, vol. 10, no. 1, pp. 3–25, Feb. 1992.

[15] J. Kempf *et al.*, "Requirements and functional architecture for IP host alerting protocol," *IETF RFC 3154*, Aug. 2001.

[16] Y. Tseng, C. Hsu, and T. Hsieh, "Power saving protocols for IEEE 802.11-based multi-hop ad hoc networks," in *Proc. IEEE INFOCOM 2002*, New York, USA, June 2002.

[17] W. Soh and H. Kim, "QoS provisioning in cellular networks based on mobility prediction techniques," *IEEE Commun. Mag.*, vol. 41, no. 1, pp. 86–92, Jan. 2003.

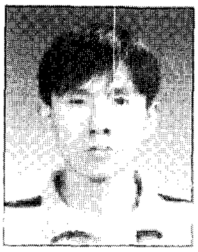
[18] D. Mcdysan, *QoS and Traffic Management in IP and ATM Networks*, McGraw-Hill, 1999.

[19] A. Balachandran *et al.* "Characterizing user behavior and network performance in a public wireless LAN," *ACM SIGMETRICS Performance Evaluation Review*, vol. 30, no. 1, pp. 195–205, June 2002.

[20] D. Staehle, K. Leibnitz, and K. Tsipotis, "QoS of Internet access with GPRS," *ACM Wireless Networks*, vol. 9, no. 3, pp. 213–222, May 2003.

[21] SD-Link11b specification, available at http://www.c-guysusa.com/SD-Link%2011b.html.

[22] T. Roos *et al.* "A probabilistic approach to WLAN user location estimation," *Int. J. Wireless Inform. Networks*, vol. 9, no. 3, pp. 155–164, July 2002.



Sangheon Pack received the B.S. (2000, magna cum laude) and the Ph.D. (2005) degrees from Seoul National University, both in computer engineering. He is currently a post-doctor fellow in the School of Computer Science and Engineering at the Seoul National University, Korea. He is a member of the IEEE and ACM. During 2002–2005, he was a recipient of the Korea Foundation for Advanced Studies (KFAS) Computer Science and Information Technology Scholarship. He has been also a member of Samsung Frontier Membership (SFM) from 1999. He received a student travel grant award for the IFIP Personal Wireless Conference (PWC) 2003. He was a visiting researcher to Fraunhofer FOKUS, German in 2003. His research interests include mobility management, QoS provision, and scalability issues in the next-generation wireless/mobile networks. Specifically, he is researching the performance of IP-based mobility protocols such as mobile IP, Hierarchical mobile IP, and IP-layer paging protocol.

Yanghee Choi received the B.S. in electronics engineering from Seoul National University, M.S. in electrical engineering from Korea Advanced Institute of Science, and Doctor of Engineering in Computer Science from Ecole Nationale Superieure des Telecommunications (ENST) in Paris, in 1975, 1977, and 1984, respectively. Before joining the School of Computer Engineering, Seoul National University in 1991, he has been with Electronics and Telecommunications Research Institute (ETRI) during 1977–1991, where he served as director of Data Communication Section, and Protocol Engineering Center. He was research student at Centre National d'Etude des Telecommunications (CNET), Issy-les-Moulineaux, during 1981–1984. He was also Visiting Scientist to IBM T. J. Watson Research Center for the year 1988–1989. He is now leading the Multimedia Communications Laboratory in Seoul National University. He is also director of Computer Network Research Center in Institute of Computer Technology (ICT). He was editor-in-chief of Korea Information Science Society journals. He was chairman of the Special Interest Group on Information Networking. He has been associate dean of research affairs at Seoul National University. He was president of Open Systems and Internet Association of Korea. His research interest lies in the field of multimedia systems and high-speed networking.