International Journal of Advances in Computer Science and its Applications – IJCSIA
Volume 4 : Issue 1 *[ISSN 2250 – 3765]*
Publication Date : 09 January 2014

Joint Design of Delay Constraint and Energy Saving for Efficient Packet Transmissions in Smartphones

Jenn-Wei Lin and J. Morris Chang

***Abstract*—In this paper, delay constraint and energy saving are jointly studied for efficiently transmitting packets in smartphones. We propose a new energy-saving transmission mechanism. The proposed mechanism first utilizes the lazy transmission concept to estimate the last transmission interval of each packet. Before transmitting a packet *p*, we also adopt the accompanied transmission concept to involve more other packets to be transmitted with packet p. The well-known knapsack problem can be applied to solve the selection of accompanied packets under a given bandwidth capacity. Finally, simulation experiments are performed to demonstrate the effectiveness of the proposed mechanism in the delay performance and energy conservation.**

***Keywords*—energy saving, delay constraint, smartphone, knapsack problem.**

I. Introduction

In the past decade, there has been rapid growth in the use of smartphones. In a smartphone, most of the applications need to exchange data over wireless interfaces. The smartphone has been equipped with a number of wireless interfaces for data communication, namely, 3G, WiFi, Bluetooth, etc. The wireless interface drains up to 50% of the total energy spent by a mobile device [1]. Nowadays researchers have proposed a variety of energy-saving approaches for smartphones to reduce the energy consumption while data communication.

The popular IEEE 802.11 standard [2] provides a power-saving mode (PSM) that periodically wakes up the wireless interface of a smartphone to synchronize with the AP for downloading packets. The 802.11 PSM belongs to a static power-saving algorithm, which does not adapt the sleep and awake durations to the degree of network activity. Recently, several works [3]-[5] highlighted that PSM may cause un-acceptable transfer delays for certain types of applications (e.g., delay-sensitive applications), and proposed the enhanced energy-saving approaches. These previous approaches do quite well in saving energy while satisfying the desired delay performance in some circumstances. However, they do not consider the following characteristics of smartphone applications:

- Multiple delay-sensitive applications may concurrently run on a smartphone. These applications have different delay constraint requirements for their respective data transmissions.
- For a smartphone application, it usually has upstream and downstream data transmissions. Most of the previous energy-saving approaches often focus on the downstream packet transmissions from the access point (AP) to the smartphone. The energy conservation is not done on the upstream packet transmissions from the smartphone to the AP.
- There is a bandwidth (throughput) limit on the wireless interface of a smartphone. When the smartphone wakes up to transmit packets in a traffic burst, the allowed bandwidth on the wireless interface may not sustain the delay constraint requirements of the packets in the traffic burst.

To take the above characteristics in the energy conservation of the smartphone, we design a new energy saving mechanism that considers the *multiple delay constraints* and *limited bandwidth provisioning*. The energy saving mechanism is abbreviated as MDLB. In the MDLB mechanism, each upstream (downstream) packet of a smartphone application is characterized as a status entry to be stored in the upstream (downstream) status list. For a status entry, it contains the important information about the last transmission interval of the corresponding packet. If the transmission of a packet cannot be completed before its last transmission interval, the required delay constraint of the packet cannot be ensured. • Basically, the estimation of the last transmission interval is for achieving the lazy transmission concept, which can aggregate more upstream and downstream packet transmissions together without violating the required delay constraints. After transmitting more packets, the smartphone can stay in a longer sleep sate to save energy consumption. However, the smartphone has limited bandwidth capacity. This introduces a packet transmission scheduling problem, i.e. how to select the upstream and downstream packets under the supplied bandwidth limit of the smartphone. The packet transmission scheduling problem can be mapped into the well-known *knapsack* problem [6]. Solving the knapsack problem is known as NP-hard [6]. In the MDBC mechanism, we also present a greedy packet scheduling algorithm to select the transmitted packets in a more efficient manner.

The rest of the paper is organized as follows. Section II reviews the related work of our concerned issue. Section III presents our delay-aware energy saving mechanism. Section IV evaluates the performance of the proposed mechanism. Finally, section V concludes the paper.

II. Related Work

A comprehensive review of energy saving methodologies and techniques is presented in [7]. The author surveys various energy-saving approaches from the physical layer to the application layer. In addition, several system models are also developed to predict the energy cost of an application.

For the energy saving in the data transmission aspect, the 802.11 standard has defined the PSM to let a smartphone (mobile device) in the active mode only for the time necessary

---

Jenn-Wei Lin is with the Department of Compter Science and Information Engineering, Fu Jen Catholic Unversity, New Taipei City, 24205, Taiwan, R.O.C. email: jwlin@csie.fju.edu.tw

J. Morris Chang is with the Department of Electrical and Computer Engineering, Iowa State Unisersity, Ames, IA. 50010 USA email: morris@iastate.edu

to exchange data. In the PSM of 802.11 [2], the AP periodically broadcast a beacon message with the traffic indication map (TIM) information. The smartphone wakes up periodically to synchronize with the AP for receiving the beacon message. From the received beacon message, if the TIM indicates that there are buffered packets in the AP for the smartphone, the smartphone will be set in the active mode to receive the buffered packets. Otherwise, the smartphone is put into the sleep mode. The energy saving of PSM will increase the packet transmission delay. There is a tradeoff between the energy conservation and delay *reduction*.

To balance the two conflicting two factors, the authors of [3] propose a bounded slowdown (BSD) protocol. It can use minimum possible energy necessary to guarantee that the round-trip times (RTTs) of Web access applications do not increase by more than a given factor *p*. To accomplish this, the mobile station stays awake for a short period of time after a request is sent, and adaptively listens to fewer beacons when the wireless link remains idle. With the BSD, the mobile station listens for the beacon with decreasing frequency during idle times to save energy. However, in the BSD, the time instants of beacon listening are still statically determined. In addition, if the tolerant delay of the packet transmission is very low, the BSD consumes more energy cost than the PSM.

To further enhance the BSD, the authors of [4] propose a new energy-saving mechanism called smart power saving mode (SPSM). Unlike the BSD, the SPSM dynamically estimates the time instants when the mobile stations wake up to listen for beacons. The SPSM can take less energy cost than the BSD. Compared to the PSM, its energy efficiency is also worse in some cases.

Unlike the previous energy-saving techniques, the work [5] uses knowledge (hints) of application traffic behavior to save energy. In the web-browsing application, the hints describe the download times of web pages and the think times of users.

The above previous approaches can both save energy consumption and satisfy the desired delay performance. Compared to our approach, some characteristics of smartphone applications are not concerned by them.

III. Proposed Energy Saving Mechanism

In this section, we will present a new energy saving mechanism which joins the multiple delay constraints and the limited bandwidth provisioning. We call the energy saving mechanism as the MDLB mechanism that consists of four components: queue division, transmission rate estimation, status list formation, and packet scheduling. The four components are initiated by three driven events (beginning, packet, and transmission), as shown in Fig. 1. The execution time unit of a smartphone application is a time frame. Each application can run in a number of time frames. At the beginning of a time frame, a beginning event is triggered to run the queue division component for determining the occupied queue size of each running application. Then, the component of transmission rate estimation is executed to estimate the transmission rate of each application in order to satisfy the required delay constraint and avoid packet losses. Next, whenever a packet is generated by a running application, a packet event is generated to invoke the component of the status list formation for storing the important information of the packet, especially the last transmission interval. For the transmission event, it occurs when the last transmission interval of a packet p is up. In such case, a state transition is performed to wake up the wireless interface of the smartphone from the sleep state. In addition, the packet scheduling component is called to transmit the packet p and also some other packets together in a duration of time. After transmitting the scheduled packets, the wireless interface of the smartphone can go into the sleep mode. Based on the above the four components, the MDLB mechanism achieves the energy saving by utilizing the lazy transmission and accompaniment transmission concept to make the wireless interface of the smartphone with longer sleep durations and few state transitions. The details of each component will be described as follows.

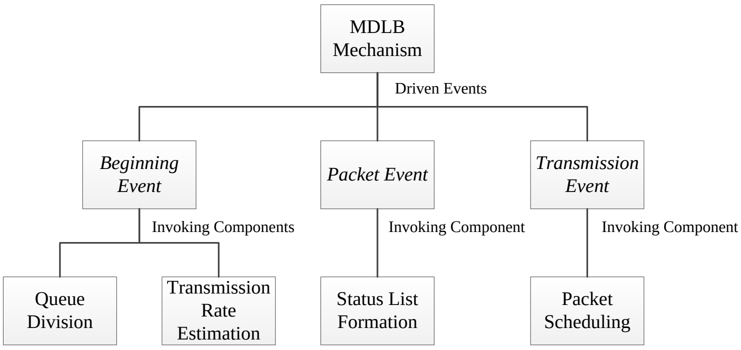


Fig. 1. The components of the MDLB mechanism and their driven events.

TABLE I Notations

Notation	Description
n	The total number of smartphone applications run concurrently in a time interval.
a_p_i	The average packet size of the application *i*
s	The size of a normalized packet.
$sq(aq)$	The size of the whole smartphone(AP) queue
$\lambda_i^u(\lambda_i^d)$	The mean number of upstream (downstream) packets transmitted in a time interval with respect to the application *i*.
$\lambda_i^{u'}\left(\lambda_i^{d'}\right)$	The effective mean number of normalized upstream (downstream) packets transmitted in a time interval with respect to the application *i*.
$\mu_i^u(\mu_i^d)$	The average transmission rate of upstream (downstream) packets in a time interval with respect to the application *i*.
$\mu_i^{u'}\left(\mu_i^{d'}\right)$	The average transmission rate of normalized upstream (downstream) packets in a time interval with respect to the application *i*.
$dc_i^u(dc_i^d)$	The required delay constrain of a normalized upstream (downstream) packet with respect to the application *i*.

A. Definitions and Notations

Before elaborating the four components of the MDLB mechanism, we first make the following definitions

- The execution time of a smartphone application can be divided into a number of fixed time intervals. The time interval is a basic execution time unit. In a time interval, there may have several delay-sensitive applications executed concurrently in the smartphone. These applications have different delay constraints among them.

- For a delay-sensitive smartphone application, its delay constraint limits the service time of a packet no greater than a specified value Δ. The service time consists of two parts. One is the waiting time of an application packet in the queue. The other is the transmission time for sending the packet from the smartphone to the AP, or from the AP to the smartphone. The sum of the waiting and transmission time cannot exceed Δ.
- A normalized packet with a fixed size *s* is also assumed here. Utilizing the normalized packet, if the average packet size of an application $\mathrm{a_i}$ is $\mathrm{s_i}$, the packet size will be normalized as $\frac{s_i}{s}$.
- The Poisson distribution is used to model the number of packets transmitted from (to) an application [x find references]. This means that packets follows the Poisson distribution to get into the transmission queue
- Without loss generality, the transmission time of an application packet is assumed to follow an exponential distribution.

In addition to the above assumptions, we also define the notations used in the proposed MDLB mechanism, as shown in Table 1.

B. Queue Division

With possible multiple applications concurrently running in the smartphone, the transmission queue of a smartphone (smartphone queue) cannot be fully occupied by only one application. In the MDLB mechanism, the smartphone queue is virtually divided based on the traffic patterns of running application at the beginning of each time interval. For the application *i*, the size qs_i^u of its occupied smartphone queue can be estimated using the following equation.

$$qs_i^u = sq \times \frac{\lambda_i^{u'}}{\sum_{k=1}^{n} \lambda_k^{u'}} \quad (1)$$

The smartphone queue is virtually divided based on the $\lambda_i^{u'}$ of each application. Unlike λ_i^u, $\lambda_i^{u'}$ considers the applications with different packet sizes and execution time in a time interval. The relationship between λ_i^u and $\lambda_i^{u'}$ can be represented as $\lambda_i^{u'} = r_i \times \lambda_i^u \times \frac{a\_p_i}{s}$.

Based on Eq. (1), we can further estimate the maximum number $\max\_\mathrm{p}_i^u$ of normalized upstream packets allowed to be stored in the smartphone queue with respect to the application i, as follows.

$$\max\_\mathrm{p}_i^u = \frac{\mathrm{qs}_i^u}{s} = \frac{1}{s} \times (sq \times \frac{\lambda_i^{u'}}{\sum_{k=1}^{n} \lambda_k^{u'}}) \quad (2)$$

Similar to the smartphone queue, the AP queue can be also virtually divided by smartphone applications. In the following, we give the maximum number $\max\_\mathrm{p}_i^d$ of normalized downstream packets allowed to be stored in the occupied AP queue with respect to the application i, as follows.

$$\max\_\mathrm{p}_i^d = \frac{1}{s} \times (aq \times \frac{\lambda_i^{d'}}{\sum_{k=1}^{n} \lambda_k^{u'} \lambda_k^{d'}}) \quad (3)$$

C. Transmission Rate and Time Estimations

After deriving the occupied smartphone queue space of each application, a queuing model is further applied to estimate the upstream packet transmission rate of each application. The packet transmission rate is an important factor to calculate the transmission interval of a packet without violating the required delay constraint. In section 3.1, the Poisson distribution and exponential distribution have been used to model the number of packets transmitted and the transmission time of a packet. Based on these two distributions, we can adopt the *M/M/1* queuing model [8] to estimate the upstream packet transmission rate, as follows.

$$\frac{1}{\mu_i^{u'} - \lambda_i^{u'}} \le dc_i^u \text{ and } \frac{(\lambda_i^{u'})^2}{\mu_i^{u'}\left(\mu_i^{u'} - \lambda_i^{u'}\right)} \le \max\_\mathrm{p}_i^u \quad (4)$$

In Eq. (4), the two well-derived equations [8] are used to represent the average transmission time of a normalized upstream packet and the average number of normalized upstream packets in the buffer, respectively. For $\mu_i^{u'}$, it is used instead of μ_i^u due to considering the applications with different packet sizes. The relationship between μ_i^u and $\mu_i^{u'}$ is $\mu_i^{u'} = \mu_i^u \times \frac{a\_p_i}{s}$. By setting two upper bound values in Eq. (4), we can estimate the upstream packet transmission rate with the delay constraint and packet lossless satisfaction.

In Eq. (4), we have estimated the smallest $\mu_i^{u'}$ with the delay constraint and packet lossless satisfaction. Therefore, the $[\mathrm{t}+\frac{1}{\mu_i^{u'} - \lambda_i^{u'}} - \frac{1}{\mu_i^{u'}},\ \mathrm{t}+\frac{1}{\mu_i^{u'} - \lambda_i^{u'}}]$ can be regarded as the last transmission interval of the packet p.

For each downstream packet of the application *i*, we can also adopt the *M/M/1* queuing model to determine its last transmission interval.

D. Packet Status List Formation

In addition to considering the delay constraint and packet losses, the energy conservation is strongly concerned in the proposed MDLB mechanism. To efficiently performing the energy conservation, two packet status lists are formed to store the status information of application packets. The data structure of the packet status is a triple (packet identifier, packet size, packet transmission interval). In a time interval, each packet in the smartphone queues will be extracted the above triple information to be put in the upstream status list. By modeling the downstream packet distribution, the traffic patterns of downstream packets can be predicated. Similarly, the status information of each downstream packet is put in the downstream status list.

E. Packet Transmission Scheduling

Based on the upstream and downstream status lists, the delay-limited packet transmissions can be scheduled as follows. In the upstream or downstream status list, if the upstream packets *p* of application *i* is with the transmission interval $[t_{p_1}, t_{p_2}]$, the smartphone will begin to transmit packet p at time t_{p_1} using the transmission rate μ_i^u. However, this packet scheduling manner does not consider energy conservation. To achieve energy efficiency, the *transmission accompaniment* is applied to reduce the number of *sleep-to-active state transitions*. Whenever a smartphone would like to transmit a packet *p*, some packets are selected to accompany the transmission of packet *p*. For these packets, their corresponding transmission intervals are moved ahead, which are called the accompanied packets of packet *p*. As a result, the smartphone can have longer active and sleep durations to reduce the number of sleep-to-active state transitions.

To maximize the benefit of the transmission accompaniment in energy consumption, accompanied packets would like to be selected as many as possible. However, the smartphone has limited bandwidth capacity. In addition, different applications have different packet sizes. To obtain the optimal solution of the transmission accompaniment problem, we transform the problem into the minimum knapsack-like problem, as follows.

- For the smartphone, its own bandwidth capacity is corresponding to the capacity of a knapsack.
- In the upstream and downstream status lists, each packet is regarded as an item to be possibly put into a knapsack.
- The packet size is corresponding to the item size.
- Each packet is associated with an *accompanied transmission cost* corresponding to the item cost of an item. The accompanied transmission cost is determined as follows. Assume that packet p has the earliest transmission interval $[t_{p_1}, t_{p_2}]$ than other packets. If the packet *q* with the transmission interval $[t_{q_1}, t_{q_2}]$ is one accompanied packet of packet *p*, its accompanied transmission cost is defined as follows:

$$[t_{q_1} - t_{p_1}] + [t_{q_2} - t_{q_1}] = t_{q_2} - t_{p_1} \qquad (5)$$

The well-known knapsack greedy algorithm [6] can be used to find the approximation solution of the transmission accompaniment in polynomial time.

IV. Performance Evaluation

A. Simulation Setup

We used MatLab [9] to perform simulation experiments for evaluating the performance of the proposed MDLB energy-saving mechanism. We use the energy consumption of PSM as the energy comparison benchmark. To measure the PSM performance, the beacon interval is set to 100ms. The non-energy saving (NES) transmission scheme can achieve the best delay performance. In this scheme, when there is a packet to or from the smartphone, the smartphone is in the awake state to immediately send or receive the packet. Its delay performance is regarded as the delay comparison benchmark.

In simulation experiments, the used energy consumption and application parameters are set as follows. There are different power usages for the wireless interface in transmit, receive, idle, and sleep states, respectively. They are set to 1.4 W, 0.9 W, 0.7 W, and 0.06 W by referring to [10]. For the state transition, its power usage is set to 1.85 W for waking up the wireless interface from the sleep state. The state transition takes a short of duration of time [11]. However, it consumes significantly higher power than being in the steady awake state [12]. In the aspect of application parameters, we assume there are five applications concurrently running in the smartphone. For simplicity, the upstream (downstream) packet size of each application is fixed to 1024 byres. The delay constraints for the five applications are given as [10, 30, 50, 70, 90] for upstream packets, and [20, 40, 60, 80, 100] for downstream packets. The bandwidth capacity of the wireless interface is 11Mpbs.

After the above simulation parameter settings, 10 simulation runs are performed. In each simulation run, the total number of upstream and downstream frames for all five applications is set from 100 to 1000 in a step of 100, respectively. The following simulation results give the mean of 10 simulation runs in the following concerned metrics: average energy consumption, and average delay transmission time.

B. Simulation Results

Fig. 2 plots the average energy consumption comparisons among the NES, PSM and MDLB . In Fig. 2(a), the spent power consumption for sending and receiving packets is included. Without applying the energy saving method, the NES has the highest amount of energy consumption. The total energy consumption is dominated by the power consumption of packet transmissions from and to the smartphone. However, the MDLB can still improve the energy costs of the NES and PSM to 37% and 17% on average, respectively. Basically, the packet power consumption cannot be saved by any energy-saving approaches. If this essential power consumption cost is not counted, the MDLB can improve more energy consumption cost than NES and PSM. As shown in Fig. 2(b), the improved average ratios are 91% and 79% by comparing with the NES and PSM, respectively. The main reason is that MDLB can aggregate more packet transmissions to make longer sleep durations and fewer state transitions for smartphone.

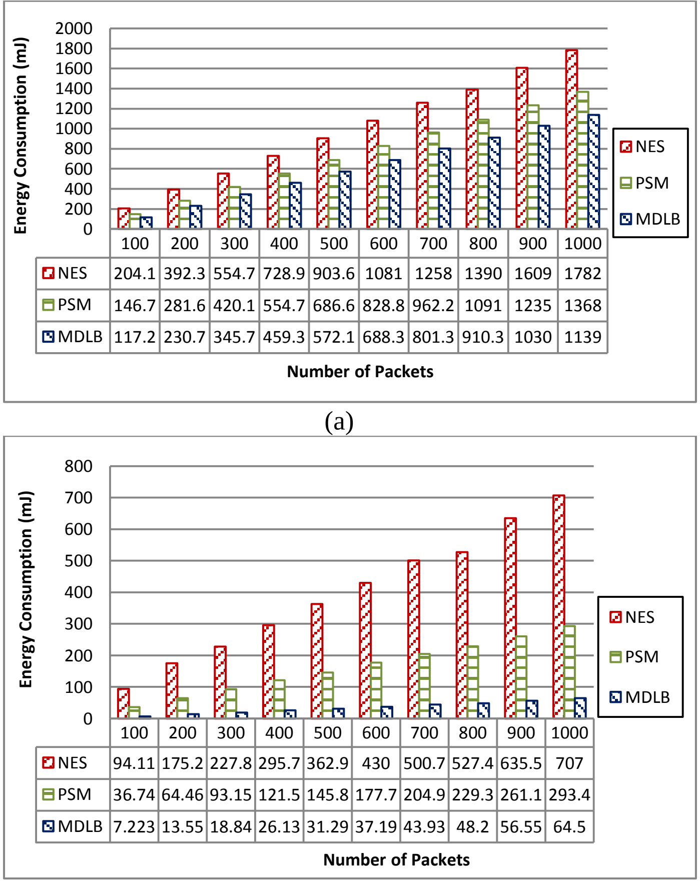
To save energy saving, the basic idea is to accumulate a number of packets to send or receive in a certain time period. This will defer the transmission (sending or receiving) times of some packets. To measure this deferring impact, the packet delay performance is defined as: *packet transmission time – packet generation time*.

Without applying any energy-saving mechanism in the NES, this approach has the best packet delay performance.

Fig. 3 gives the comparison of packet delay performance. The NES can immediately send or receive packets. The MDLB can aggregate packet transmissions to transmit some packets before their last transmission intervals. Due to aggregating packet transmissions, some packets have better packet delay performance. As shown in Fig. 3, the packet delay performance of the MDLB is about 8.62 times larger than that of NES. For the PSM, it statically defer the transmission times of downstream packets at the beginning time of next beacon interval. On average, the packet delay performance of PSM is 4.3 times larger than that of MDLB 50ms. Compared to the MDLB, it increase about x times in the packet delay performance.

V. Conclusions

We have investigated the joint problem of delay constraint and energy saving for smartphones. In addition to the delay constraint, the bandwidth limit of the smartphone is also considered in the proposed mechanism. In the proposed mechanism, we utilize the queuing model to estimate the last transmission interval of each packet without violating its delay constraint. To consider the energy consumption, if a packet is p is transmitted at [t1, t2], it also would like to involve more other packet transmissions at the same time duration. As a result, the accompanied transmission problem is introduced, i.e., how to find accompanied packets as many as possible under a given bandwidth limit. In the proposed mechanism, the problem is transformed to the well-known knapsack problem. Finally, the simulation results showed that the proposed replication algorithms can perform energy-efficient packet transmissions with delay-bound guarantees.



(a)

(b)

Fig. 2. Average energy consumption comparison (a) Total energy cost. (b) Excluding packet energy cost.

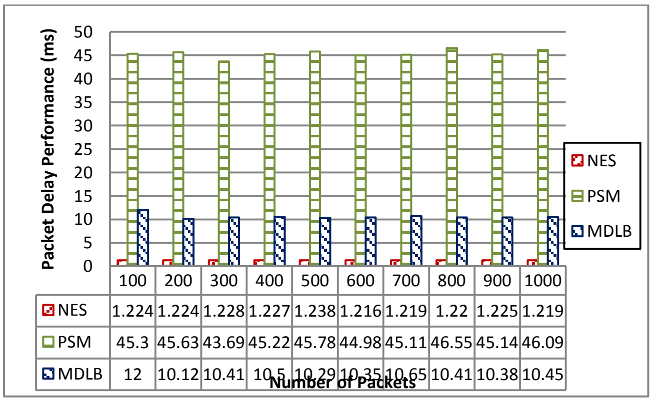


Fig. 3. Average packet delay performance comparison.

Acknowledgment

This research was supported by the National Science Council, Taiwan, R.O.C., under Grant NSC 99-2221-E-030-007-MY3.

References

[1] R. Kravets and P. Krishnan, “Power Management Techniques for Mobile Communication,” in *Proc. of the 4th Annual ACM/IEEE Int. Conference on Mobile Computing and Networking.* (MobiCom ’98), Dallas, TX, 1998, pp. 157–168.

[2] IEEE Computer Society LAN MAN Standards Committee, *Wireless LAN Medium Access Control and Physical Layer Specifications*, August 1999.

[3] R. Krashinsky and H. Balakrishnan, “Minimizing Energy for Wireless Web Access with Bounded Slowdown,” *Wireless Networks*, Vol. 11, No. 1-2, Jan. 2005, pp. 135-148.

[4] D. Qiao and K.G. Shin, “Smart Power-Saving Mode for IEEE 802.11 Wireless LANs,” *Proc. of the 24th Annual Joint Conference of the IEEE Comput. and Commun. Societies* (INFOCOM 2005), Miami (FL), March 13-17, 2005.

[5] M. Anand, E. Nightingale, and J. Flinn, “Self-Tuning Wireless Network Power Management,” *Wireless Networks*, Vol. 11, Issue 4, Jul. 2005, pp. 451-469.

[6] S. Dasgupta, C. H. Papadimitriou, and U. V. Vaziran, “Algorithms,” *McGraw-Hill*, 2006.

[7] K. Naik, “A Survey of Software Based Energy Saving Methodologies for Handheld Wireless Communication Devices,” *Technical Report 2010-13, Department of Computer and Electrical Engineering*, Oct.2010.

[8] D. Gross and C. M. Harris, Fundamentals of Queuing Theory, 3rd ed. New York, NY: Wiley, 1998.

[9] (2013), MathWorks - MATLAB and Simulink for Technical Computing. [Online]. Available: http://www.mathworks.com.

[10] Y. Xie, X. Luo, and R. Chang, “Centralized PSM: An AP-centric power saving Mode for 802.11 infrastructure networks,” IEEE *Sarnoff Symposium*, Mar. 2009, pp.1-5.

[11] A. Kameman and L. Monteban. “WaveLAN-11: A High-Performance Wireless LAN for the Unlicensed Band,“ *Bell Labs Technical Journal*, pp. 118-133, Summer 1997.

[12] P. J. Havinga and G. J. Smit, “Energy-Efficient TDMA Medium Access Control Protocol Scheduling,” in *Proc. Asian International Mobile Computing Conference*. Nov. 2000