Boe: Context-aware Global Power Management for Mobile Devices Balancing Battery Outage and User Experience

Chenren Xu*, Vijay Srinivasan†, Jun Yang†, Yoshiya Hirase†, Emmanuel Munguia Tapia†, Yanyong Zhang*
*{lendlice, yyzhang}@winlab.rutgers.edu, †{v.srinivasan, j3.yang, y.hirase, e.tapia}@samsung.com
*WINLAB, Rutgers University, North Brunswick, NJ, USA
†Samsung Research America, Silicon Valley, San Jose, CA, USA

***Abstract*—Energy conservation on mobile devices is now more important than ever due to the increasing benefits that smartphones and tablets provide to our daily life. However, most existing power management approaches either focus narrowly on a particular sub-system of the mobile device such as the sensor system, the LCD display, or the communication system, or use heuristic approaches to maximize energy efficiency at the cost of user experience. In this paper, we present *Boe*, a context-aware global power management scheme for mobile devices *B*alancing battery *o*utage and user *e*xperience. To meet the mobile device's expected battery life while sacrificing end user experience as little as possible. *Boe* takes into account the users' phone usage patterns and activities to dynamically adjust the device's global power management policy to minimize outage time and maximize user experience. We demonstrate our proposed technique by controlling display brightness level and GPS sampling rate on smartphones. We evaluate our approach through real world smartphone data from 10 users over two months. Compared to the best fixed user experience policies, we show that: (i) *Boe* eliminates *all* frustrating battery outage events for light, moderate, and heavy phone users, and (ii) *Boe* improves user experience by 20% for light users, maintains the same user experience for moderate users, and degrades user experience by 23% for heavy smartphone users.**

***Index Terms*—Context, Power Management, User Experience, Energy Efficiency**

I. Introduction

Modern mobile devices such as smartphones and tablets support a wide range of applications that benefit end users, including applications for telephony, gaming, web browsing, location based services, health, email, office, and social networking. The wealth of these applications on smartphones, improvements in computing power, network bandwidth, and overall user experience in modern smartphones and tablets has resulted in a situation where these mobile devices are used more heavily than earlier generations of cell phones. However, battery technology and energy efficiency has not yet advanced to the point where users can make heavy use of their mobile devices, and still expect the battery charge to last during the entire day.

Existing research has explored numerous approaches to maximize energy efficiency of smartphones and mobile devices. Most approaches focus exclusively on certain subsystems of the smartphone, such as the communication system [14], the sensing and classification system [6], [13], the phone display [8] or the GPS [21], [15]. While these approaches provide valuable techniques to improve the energy efficiency of individual sub-components on mobile devices, the main drawback of these approaches is that they provide no *global* framework to prevent battery outage for end users while maximizing user experience.

In particular, two problem scenarios are common among smartphone and tablet users: (i) the device's battery charge is depleted during the middle of the day and there is no power outlet or charger available to charge the device, or (ii) users receive a low battery warning, and immediately start turning off services, sensors, radios, and applications to save power for emergency situations. In this work, we focus on the problem of reducing or completely eliminating such frustrating battery outage events, while reducing the impact of the energy management on overall user experience. One of our main goals is to mitigate the likelihood of users facing problems (i) and (ii) above, since these scenarios leave the smartphone or tablet unavailable to perform important tasks for a significant portion of the day.

In this paper, we present *Boe*, a smart context-aware global power management scheme for ***B***alancing Battery ***O***utage and User ***E***xperience), designed to meet the mobile device's expected battery life while compromising end user experience as little as possible. *Boe* takes into account users' typical phone usage and energy consumption patterns and user experience preferences for various smartphone components. *Boe* uses a Markov Decision Process model to control the energy consumption of various components on smartphones to minimize battery outage for end users while maximizing end user experience as much as possible.

We show the effectiveness of *Boe* in eliminating battery outage while maximizing user experience by jointly controlling the energy consumption of two major components on smartphones: the display and GPS location sensing components. We evaluate the energy savings of *Boe* using real world smartphone data collected from 10 users over two months. Compared to the best fixed user experience policies, we show that: (i) *Boe* eliminates *all* frustrating battery outage events

This work was performed while Chenren Xu was working as a research intern at Samsung Research America in 2013.

for light, moderate, and heavy phone users, and (ii) *Boe* improves user experience by 20% for light users, maintains the same user experience for moderate users, and degrades user experience by 23% for heavy smartphone users. Moreover, we show how *Boe* could easily scale to include new context information about user status and new power saving actions for controlling additional hardware.

We make the following key contributions in this paper:

- We designed a context-aware global power management scheme called *Boe* to minimize battery outage and maximize user experience.
- We evaluate our energy management scheme using real-world smartphone data from 10 users.
- We demonstrate that *Boe* improves user experience and eliminates battery outage events while jointly controlling energy consumption of display and GPS sensing system.
- We show how our proposed approach is scalable enough to handle additional hardware components and power saving actions, new context information about end users, and can also account for individual user preferences for user experience from different components on the phone.

II. Related Work

Several existing researchers have focussed on identifying the major energy consumers on mobile devices. In particular, Shye et al observe that the powerdown time of a mobile device is highly dependent upon the individual user, but the screen and CPU tend to dominate the active power consumption [17]. Carroll et al. [5] did a circuit level measurement on different hardware components on smartphones and demonstrated that GSM is the dominating energy drain, followed by CPU and graphics. Thus, several existing researchers have focussed on improving the energy efficiency of individual sub-components on smartphones. Particular focus has been on saving energy for the phone screen, the location sensing system, the CPU, and the communication system, since these four components are the major energy consumers in modern smartphones.

Anand et al. [3] proposed to use tone mapping function to dynamically brightening the image or content and dimming the display, and save up to 47% of the display power with no perceptible quality loss. Chameleon [8] reduces system power consumption of OLED smartphones by 41% for web browsing without introducing any user noticeable delay through power-optimized color schemes design. Similarly, multinets [14] proposes automatically choosing the network radio to optimize energy consumption, while Chu et al [6] propose techniques to balance energy, latency, and accuracy for mobile context classification. Trading off location sensing accuracy versus energy efficiency on GPS also have been well studied, such as achieving rate-adaptive sensing through other modality (bluetooth and celltower) [15], or software architecture optimization via substitution, suppression, piggybacking of applications' location-sensing requests [21].

Other researchers have focussed on providing feedback to end users to save energy. CABMAN [16] presented context-aware prediction algorithms such as next charging opportunity and battery life for battery management. Ferreira et al. [9] studied the battery charging pattern of 4000 participants and provide some suggestions to avoid energy waste and opportunistic processing based on users' charging habits.

However, the existing approaches discussed above suffer from one or more of the following three drawbacks: (i) They focus narrowly on individual sub-components of smartphones and miss opportunities for optimizing energy consumption of the smartphone as a whole. For example, if the screen is rarely used, a higher sampling rate may be used for the location sensing to take advantage of the increased energy available from the battery. (ii) Also, the existing approaches do not take into account different user preferences for user experience from different hardware components. For example, some users may care about consistent screen brightness, while other users may care about smooth operation without any glitches due to CPU energy optimization. (iii) Also, many approaches are open loop approaches that maximize energy efficiency even when there is no need to do so based on the battery lifetime expected from end users. For example, if the battery level is 80% and the user typically charges the phone within 5 hours every day there may be no need to save energy on many hardware components on the phone.

Our proposed approach *Boe* addresses all 3 drawbacks above. It provides an optimization framework to optimize energy consumption of the mobile device as a whole; many of the techniques proposed in related work such as changing sampling rate of sensors, adjusting screen brightness, or proving battery lifetime reminders may be incorporated as part of our system as long as the tradeoff between energy efficiency and user experience may be measured. *Boe* can take into account different user preferences for user experience on different hardware components, and also take into account the expected battery lifetime while determining the need to save energy, thus addressing both drawbacks (ii) and (iii) above.

III. Boe Dynamic Energy Management System Design

Boe dynamically adapts the power consumption of smartphone components such as the GPS unit and the LCD display, to achieve maximum user experience while ensuring that the battery is not depleted before the expected user charging time. *Boe* uses a Markov Decision Process (MDP) that takes as input a target battery lifetime and the user's daily smartphone usage patterns. The MDP assigns a dynamic power level to each smartphone component at each time instant to maximize user experience while ensuring that the target battery lifetime is reached. *Boe* has a learning phase where the user's longitudinal phone usage patterns are used to train the MDP model.

A. Markov Decision Process Overview

A Markov Decision Process [4] (MDP) is represented as a 4-tuple $(S, A, P_{\cdot}(\cdot,\cdot), R_{\cdot}(\cdot))$, where

- S is a finite set of states.
- A is a finite set of actions.

- $P_a(s,s') = Pr(s_j = s' | s_i = s, a_i = a)$ is the probability that action a in state s will lead to state s'.
- $R_a(s)$ is the immediate reward received after taking action a in state s.

The value of a state s under policy π is the expected sum of the discounted rewards by following policy π under s, defined as

$$V_\pi = R_a(s) + \gamma \sum_{s' \in S} P_{\pi(s)}(s,s') V_\pi(s').$$

Once the parameters are all known, the optimal policy is computed by solving the Bellman Equation:

$$V^*(s) = \underset{a}{\operatorname{argmax}} \left[R_a(s) + \gamma \sum_{s' \in S} P_a(s,s') V(s') \right].$$

B. Problem Formulation

We describe below how we formulate the energy management problem on mobile devices as a Markov Decision Process.

1) State and Action Space: In our problem, the objective is to learn the appropriate power management actions (dim screen, increase GPS sampling rate) to take in different states (morning, user indoors) to ensure that we achieve high user experience while maintaining enough battery charge to last through the day.

In our model, the composite state is defined as:

$$s := (t, c, e),$$

where t is the time of the day, c is the context, and e is the battery level.

Time of the day t. We discretize the total amount of time T left before the phone is charged into N_T ticks. For example, we have 24 ticks per day if the time interval N_T is set to one hour. The transition of t strictly follows a degenerate Markov chain:

$$P(t_j|t_i) = \begin{cases} 1 & \text{if } t_j = t_i + 1 \\ 1 & \text{if } t_j = t_i = N_T \\ 0 & \text{otherwise} \end{cases}$$

Context c contains the user's rich contextual information, which can be defined as a composite state with different modalities. In this paper, we define c as:

$$c := (u, l, m), \text{where}$$

- u is a binary variable indicating whether the user is using a foreground application, 0 (display off) and 1 (display on).
- l is a binary variable indicating whether the smartphone is indoors (0) or outdoors (1), which can be potentially inferred from ambient light level, cellular signal strength, or other information, as demonstrated in [20];
- m indicates the device's mobility level: 0 (none, e.g. sitting or standing), 1 (low, e.g. walking or running), and 2 (high, e.g. driving).

The state transition probability matrix is obtained from each user's historical context data used for training the MDP model.

Battery level e denotes the remaining energy budget. We divide the total battery energy E_{total} into L uniform levels and each battery level thus contains energy $E_0 = E/L$. The transition of battery levels is determined by user actions, which is defined in a composite form as well.

In our use case, an action a in the action space A is defined as:

$$a := (a_{display}, a_{gps}),$$

where $a_{display}$ is a set of predefined display brightness levels and a_{gps} is a set of GPS sampling intervals. Higher display brightness levels and shorter GPS sampling intervals will draw energy faster from the battery. We denote the power consumption for each action a as

$$power(a) = power_{display}(a) + power_{gps}(a).$$

Similar to the Jigsaw system [12], the probability of the battery level changing from the current level to the next level from time tick t_i to t_{i+1} is calculated as

$$p(a) = \frac{power(a)}{E_0} \frac{T}{N_T}.$$

We model the transition probability of e as a function of action $a \in A$:

$$P_a(e_j|e_i) = \begin{cases} 1 & \text{if } e_j = e_i = 1 \\ p(a) & \text{if } e_j = e_i - 1 \\ 1 - p(a) & \text{if } e_j = e_i = l, l \in [2, L] \\ 0 & \text{otherwise} \end{cases}$$

To reduce the complexity of the MDP, we assume that the states t_i, u_i, l_i, m_i e_i are independent of each other. Thus, we define the overall system transition probability as,

$$P_a(s_j|s_i) = P(t_j|t_i) \cdot P(c_j|c_i) \cdot P_a(e_j|e_i), \text{where}$$

$$P(c_j|c_i) = P(u_j|u_i) \cdot P(l_j|l_i) \cdot P(m_j|m_i).$$

2) Reward Function: The reward function is a key component of the MDP as it determines the balance between user experience and energy conservation; it encourages better higher user experience when sufficient energy is available, and encourages power-saving actions when necessary to guarantee the target battery lifetime is reached. The ideal result for the MDP would be to run out of battery right at the end of the day by maximizing user experience with a bright display or high GPS sampling rate.

We define the reward function for each action a as:

$$R_a(s_i) = \begin{cases} -R_{outage} & \text{if } e_i = 1, t_i < T, \\ f(a, c_i) & \text{otherwise.} \end{cases}$$

The term R_{outage} rewards power-saving actions when necessary to ensure battery lifetime is reached while the term $f(a, c_i)$ rewards high user experience.

Energy Saving actions to achieve the target battery lifetime is achieved by R_{outage}, which penalizes the system by providing a high negative reward to cancel the cumulative positive rewards gained from previous instances of high user experience. Larger values of R_{outage} result in a lower likelihood of battery outage, and result in more conservative power settings that reduce user experience, as we will see in Section IV.

User Experience term $f(a, c_i)$ quantifies the user experience for the GPS and display based on the current context c_i and action a. We define:

$$f(a, c_i) = K_{display} U_{display}(a, c_i) + K_{gps} U_{gps}(a, c_i).$$

The weights $K_{display}$ and K_{gps} are used to adaptively weight the relative priority between display and GPS user experience for different users.

For the display, it is straightforward to directly relate the user experience $U_{display}$ to the display brightness level. However, under strong environmental ambient light, such as in outdoor environments, the display brightness needs to be increased to improve user experience. The location information enriches our model and refines the definition of user experience under different light conditions. As defined in Table I(a), we set 60% as the minimum brightness level outdoors [1] and set -∞ for the reward when the brightness is below 60% to avoid choosing these actions.

For GPS, U_{gps} denotes the user experience for location-based applications or services, such as navigation, location log for fitness, socialness, etc. Thus, U_{gps} is negatively correlated to positioning error, which is determined by a user's velocity v and the GPS sampling interval τ. During the sampling interval, the expected positioning error is the integral of all the differences between the last known position to each unknown position over the sampling interval, which is computed in Table II under different mobility levels. Even though GPS is well-known to have positioning error, we assume here that GPS gives us accurate locations to approximate the localization error due to higher sampling intervals for the purpose of defining user experience. We assume a 10-m average positioning error will be sufficient for most applications [10] such as daily life logging or location-based services. We define the user experience for each GPS sampling interval in Table I(b) under different mobility levels. For example, we assign the same user experience (5) to those GPS sampling intervals that are no more than 10 seconds (i.e., 1 sec, 5 sec, and 10 sec) when the user's mobility level is low. The reason is that a 10-m accuracy is sufficient for most location services under low mobility [10]. When the user has high mobility, to ensure sufficient location accuracy, the reward is set to -∞ for sampling intervals larger than 10 seconds and the highest (5) for a 1 second sampling interval.

C. Extensibility

As we mentioned earlier, *Boe* can be easily extended to incorporate more hardware components by augmenting new power-saving actions corresponding to the hardware to the action space and new context states if necessary into the MDP framework; the key requirement is that the action space contain different levels of power consumption, each with a different user experience which can be quantified in the reward function. Similarly, we can incorporate user feedback to the MDP by associating a potential energy reduction from each user reminder or feedback and a corresponding drop in user experience.

[1]We surveyed 10 participants who all had trouble reading text from Google maps or email outdoors when the display brightness level was below 60%.

(a) $U_{display}$

Brightness level	Indoor space	Outdoor space
20%	1	-∞
40%	2	-∞
60%	3	1
80%	4	3
100%	5	5

(b) U_{gps}

Sampling interval	Low mobility	High mobility
60 secs	1	-∞
30 secs	3	-∞
10 secs	5	1
5 secs	5	3
1 sec	5	5

TABLE I
USER EXPERIENCE (A) FOR DIFFERENT DISPLAY BRIGHTNESS LEVELS UNDER INDOOR/OUTDOOR WHEN THE DISPLAY IS ON AND (B) FOR DIFFERENT GPS SAMPLING INTERVALS UNDER DIFFERENT MOBILITY LEVELS WHEN USER IS NON-STATIONARY.

	GPS sampling interval				
Mobility level	1 sec	5 secs	10 secs	30 secs	60 secs
Low (1-3 m/s)	1 m	5 m	10 m	30 m	60 m
High (10-30 m/s)	10 m	50 m	100 m	300 m	600 m

TABLE II
AVERAGE POSITIONING ERROR UNDER DIFFERENT GPS SAMPLING INTERVALS.

IV. EVALUATION

A. Datasets

We evaluate the performance of *Boe* through real-world smartphone data traces collected from 10 users over 2 months. We collected timestamped data of user interaction with the display and foreground applications, GPS locations, and battery levels. We deployed a data collection Android app called EasyTrack to collect this data from the subjects in our study. EasyTrack is based on the Funf Open Sensing Framework [2], with the customized data probes and sampling rates detailed in Table III.

B. Preprocessing

To acquire meaningful context data to train our Markov decision process model, we preprocess the raw data and reorganize it into daily context tables for each user. Each table is indexed by time of day in minutes (1440 entries per day) with context information such as discretized battery level, display on/off, indoor/outdoor, and mobility level. When preprocessing the raw data, we make several simplifying assumptions: (1). We use simplifying assumptions to compute context data between sampling intervals, such as linear interpolation of battery levels between 15 minute sampling intervals, and assuming constant velocity between two locations during 10

Probe	Description	Sampling
Battery	Battery level and charging indicator.	Every 15 mins
Foreground Applications	Foreground running apps with starting point and duration in secs	Always on, passively listen
Location	Location in longitude and latitude	Every 10 mins

TABLE III
DESCRIPTION AND SAMPLING STRATEGIES OF RELEVANT DATA PROBES.

(a) $Power_{display}$

Brightness level	Measured power consumption
20%	246 mW
40%	356 mW
60%	466 mW
80%	576 mW
100%	686 mW

(b) $Power_{gps}$

Sample interval	Scaled power consumption
60 secs	12 mW
30 secs	24 mW
10 secs	72 mW
5 secs	144 mW
1 sec	370 mW

TABLE V
POWER CONSUMPTION OF DISPLAY AND GPS UNDER DIFFERENT SETTINGS.

minute sampling intervals for GPS[2].(2). We use the estimated velocity to infer motion state/mobility level. (3). We identified being indoors or outdoors by checking device velocity: if the velocity is less than 0.1 m/s, we assume indoors, and if greater, outdoors. We note that these simplifications may not be accurate; however, since our main goal is to evaluate our global energy management approach, these simplifications provide a good approximation for real-world context data that we might observe. Existing approaches for indoor/outdoor detection [20] and motion state detection [12] could be integrated into a more complete system for future work.

C. Power Profiling

We also profile the power consumption of the display and GPS components for model building and evaluation. Our measurement is based on Samsung Galaxy S3 model. The total battery energy is $2100mAh \times 3.8V = 28728J$.

Display: We developed a simple Android app to manually set different brightness levels and use the Monsoon power meter [1] for power measurements. We first measure just the baseline CPU power consumption by shutting down all background services and turn off all other hardware components such as WiFi, GPRS, Bluetooth, GPS, etc. We then measure the power consumption of different brightness levels by subtract the baseline CPU power value. We report the resulting power consumption measurements in Table V(a).

GPS: We also develop a simple Android app to turn on the GPS for positioning and turn off GPS immediately. We use the PowerTutor [19] software to measure the power consumption for GPS sampling. We observed that if we continuously stream GPS locations, the power consumption is about 370 mW, which is similar to that reported in [15]. However, if we turn off the GPS immediately after it returns the location, the whole process took about 4 seconds, and the average power consumption is 144 mW, which agrees with circuit-level measurements [5]. Thus, we mark 144 mW as the power consumption for 5-second sampling interval, and simply calculate the power consumption for longer sampling intervals assuming that the total energy consumption remain the same while only the sampling duration increases. The power consumption results are listed in Table V(b).

D. Evaluation Methodology

We follow a training-testing procedure for each user to evaluate the performance of *Boe*: we use the first month of data collected for training and the second month of data collected for testing.

Training: The goal of training is to estimate user's context transition probabilities as well as other system parameters to jointly determine the overall state-action transition probabilities. We set $T = 12$ hours beginning from 8 am to 20 pm to ensure that users' battery lasts the entire day when they are typically away from home. We note that T can be learned automatically in the future according to the individual's daily schedule. We set each time tick to be 15 minutes and each energy level as 25% of E_{total}; $K_{display}$, K_{gps}, and K are all set to 1. Using these parameters, we obtain all context transition probabilities using frequency counts and apply the Policy Iteration algorithm [11] to learn the optimal policy (display brightness level and GPS sampling rate). The output is a 5-tupled lookup table. Each entry represents a state-action pair which includes the state tuple $< t, u, l, m, e >$ and the optimal policy (action) tuple $< a_{display}, a_{gps} >$. We also empirically set $R_{outage} = 75$ to maximize the user experience while guaranteeing that no outage time occurs across the training data. Since we demonstrate our approach by controlling only two hardware components, we set the total energy budget for the display and GPS to be 20% of the fully charged battery energy.

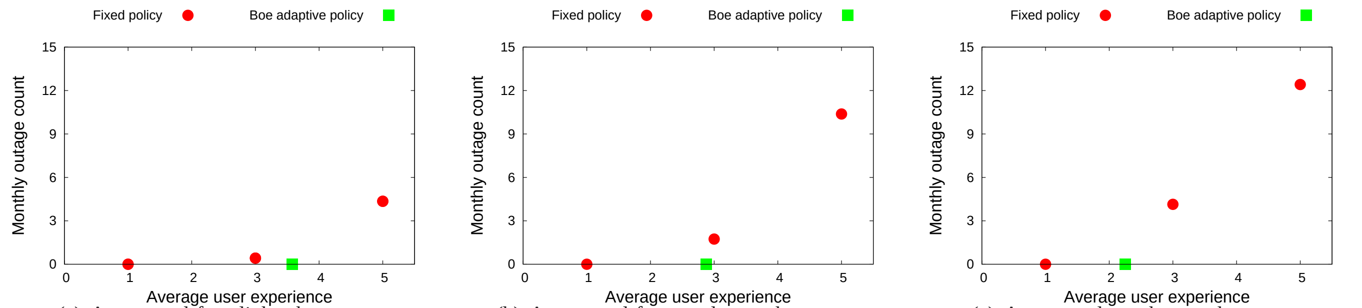
Testing: For each user and each day, we first extract the segments of time when the battery is discharging, and lookup the MDP policy table based on the observation tuple $< t, u, l, m, e >$ for every minute based on the learned the policy π. π denotes the power setting for the GPS and display for each state during the discharging period. We replay the power setting of the MDP policy, resulting in a specific average user experience and a gradual reduction in battery energy; when the battery energy drained by GPS and display exceeds a predefined budget for these two components, we define that an outage has occurred and record the timestamp when the battery outage first occurred.

E. Evaluation Metric

To evaluate the performance of *Boe*, we use the following metrics:

- *User experience* is defined as the average user experience based on Table I for when either the display or the GPS sensors are being used.

[2]We use standard Haversine formula to compute the distance between two GPS locations



(a) Aggregated four light phone users (b) Aggregated four moderate phone users (c) Aggregated two heavy phone users

Fig. 1. *Boe* outperforms fixed user experience policies as it guarantees no outage count while maximize user experience from the aggregated 10 users' data divided into three categories.

Users' phone usage level	User experience				Average outage time (min)				Maximum outage time (min)			
	Fixed max	Fixed mid	Fixed min	Adaptive *Boe*	Fixed max	Fixed mid	Fixed min	Adaptive *Boe*	Fixed max	Fixed mid	Fixed min	Adaptive *Boe*
Light users	5	3	1	3.6	82.6	5.0	0	0	275	5	0	0
Moderate users	5	3	1	2.9	323.3	59.0	0	0	684	96	0	0
Heavy users	5	3	1	2.3	705.6	185.5	0	0	750	475	0	0

TABLE IV

Boe OUTPERFORMS FIXED USER EXPERIENCE POLICIES AS IT GUARANTEES NO OUTAGE TIME WHILE MAXIMIZE USER EXPERIENCE FROM THE AGGREGATED 10 USERS' DATA DIVIDED INTO THREE CATEGORIES.

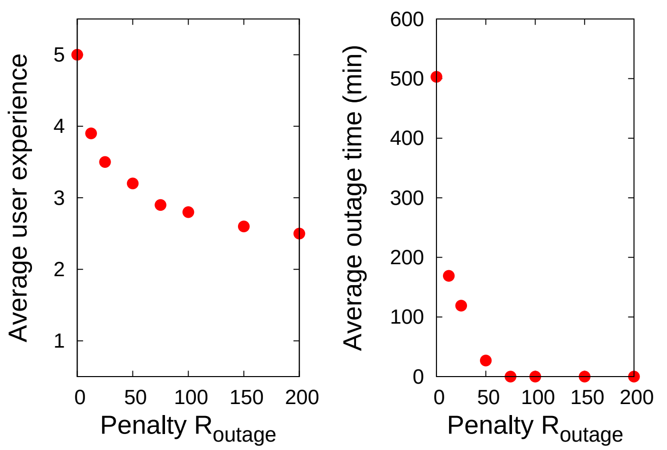


Fig. 2. We can control the outage time and user experience by varying the penalty term R_{outage}.

- *Outage count* is calculated as the number of *outage events*, where each outage event happens whenever the energy budget for the GPS and display is exceeded before the end of the battery discharging period. We compute the outage count on a monthly basis.
- *Outage time* for each outage event is calculated as the time duration between when an outage event happens and the user charges the battery next. For example, if a battery budget outage happens at 7 PM and the user charges the phone only at 8:30 PM, the outage time for the event is 90 minutes. We measure both the average and maximum outage time for each user.

F. Results

We evaluate the performance of *Boe* by replaying its policy on real-world smartphone traces from 10 users. We use our evaluation metrics of user experience, outage count, and outage time to compare the performance of *Boe* with a fixed user experience policy that sets three possible fixed, user experience values: min (1), mid (3) and max (5) with the corresponding actions on display brightness and GPS based on current context information (indoor/outdoor and motion state). We categorize our 10 users into three groups, namely light, moderate and heavy users based on thresholding the average outage time under fixed max user experience (5); heavy users tend to use the phone more and have higher motion states compared to light users. In this experiment, we set an equal user experience preference for the display and GPS: $K_{display} = K_{gps} = 1$.

Figure 1 shows the monthly outage count for light, moderate and heavy users for the three fixed user experience policies and *Boe*. Table IV shows the corresponding average and maximum outage time for outage events for the three categories of users. Firstly, in Figure 1(a), we see that light users have almost no outage events with mid level user experience of 3. However, *Boe* achieves zero outage as well but also improves the average user experience to 3.6, an improvement of 20% in user experience. For a higher user experience of 5, we have more than 4 outage events per month; Table IV shows that even light users with a high user experience of 5 see outages averaging 82.6 minutes, with a maximum outage time of more than 4.5 hours per user. Secondly, for moderate phone users in Figure 1(b), *Boe* achieved almost the same average user experience as the fixed mid policy while eliminating the 2 outage events that happen per month; these outages

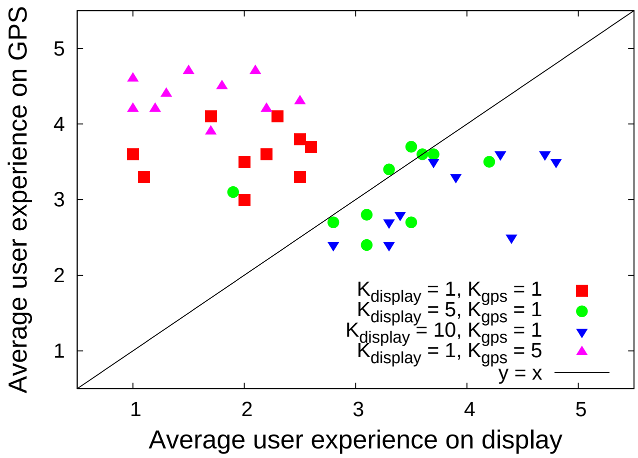
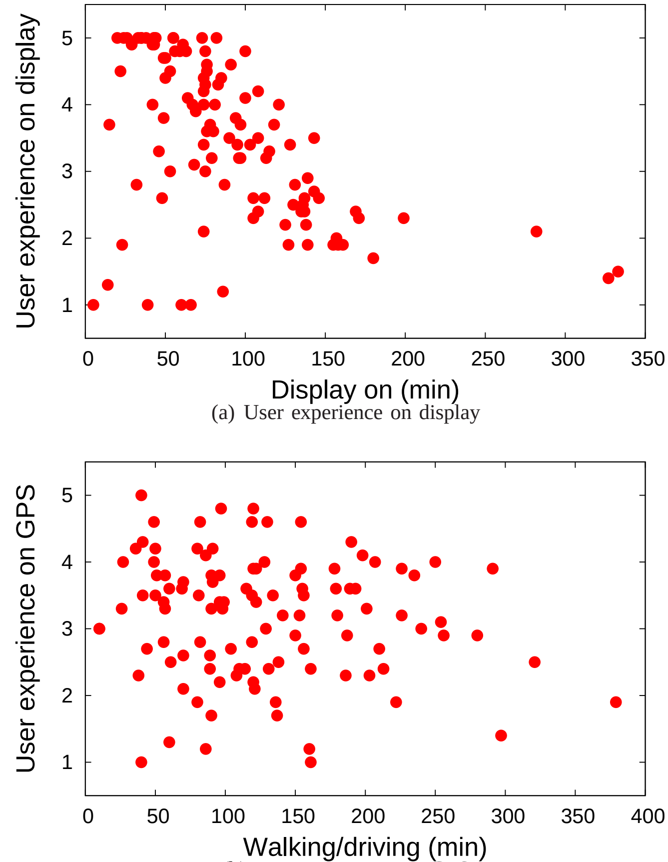


Fig. 3. We can control the user experience on different components by assigning different weights for each component.



(a) User experience on display

(b) User experience on GPS

Fig. 4. User experience for each component as the user spends more time on that component.

are quite frustrating since they range from 1 to 1.5 hours as seen in Table IV. Finally, for heavy users in Figure 1(c), *Boe* eliminates all the 4 outage events seen with the fixed mid policy but achieves an average user experience of 2.3 which is 23% lower than the fixed mid policy; however, we argue that the loss in user experience could be more tolerable to users compared to the frustratingly high outage times of 3-8 hours seen for heavy users in Table IV. Overall, *Boe* eliminates all outage events and tries to achieves as high a user experience as possible while satisfying the battery lifetime requirements. Overall, *Boe* does a better job than the three fixed user experience policies because it is aware of how much energy the user typically consumes compared to how much energy is left and adjusts user experience accordingly; we noticed that especially for the cases when the phone is charged multiple times during the day, *Boe* maximizes user experience at discharging periods where the battery level was high.

In our system *Boe*, the penalty term R_{outage} for battery outage plays a critical role in controlling the user experience and outage time. Figure 2 shows the relationship between R_{outage}, outage time, and user experience across all users. In Figure 2, we see that when $R_{outage} = 0$, *Boe* degenerates to the fixed max policy with average user experience of 5 and 503 minutes outage time. As R_{outage} increases, we see that both average user experience and outage time decrease. When $R_{outage} = 75$, we achieve the approximate optimal result: we have no outage at all while maximizing the user experience to 2.9.

We show that we are able to assign different user experience preferences for different components in Figure 3, in which each point shows the average user experience on display and GPS for each user. By default, when we assign an equal weight for $K_{display}$ and K_{gps}, users have a higher user experience on GPS than on display because the power consumption of the GPS is less than that of the display. To balance the user experience on both sides, we can assign a larger weight to the display, for example, 5 times as much as GPS. A larger weight ($K_{display}$ = 10) will favor better user experience for the display. In future work, we plan to adjust the user experience weights automatically based on the energy consumption of each component.

Finally, we show the user experience on individual days for all users. We empirically choose $K_{display} = 5$ and $K_{gps} = 1$ from the result of Figure 3. We show the display on-duration vs. user experience on display, and the non-stationary motion duration vs. user experience on GPS for all days across all users in Figure 4 as a scatter plot. In general, we observe that we have lower user experience on the component that is used more heavily. However, we also notice a few interesting points: the points shown in the left bottom area are the rare occasions when the user forgets to charge the phone at night and only has very limited energy at the beginning of the day. In general, we still see some points with high user experience even if the usage duration is high due to the advantage of global power management; *even if one component is used heavily, if the other component is used rarely a high average user experience can still be achieved by taking a global approach to power management*. The slope of GPS plot is smaller than in that of display plot because of the high power consumption of the display and the user experience on GPS is still favored even with our current parameter setting of ($K_{display} = 5$ and $K_{gps} = 1$).

V. Limitations and Future Work

In this section, we discuss the limitations and a few future directions of this work.

A. User Experience

A key challenge to our power management problem is the predefined user experience value because it trades off the energy consumption for user experience at different stages. In Table I, we propose a simple scheme to quantize the user experience for the phone's display and location service under different contexts. We interviewed 20 participants about our user experience quantization. 18 participants agreed with our quantization, and two of them disagreed because they believed that human perception does not follow a linear scale. However, they did agree that in general, human gains higher user experience with brighter display and higher accuracy of location estimation. Also, they pointed out that they would prefer a fixed dim level (70% or so) in completely dark environment to avoid stimulating eyes. These lessons learned will collectively help us to improve this context-based optimal user experience design in our future work. Meanwhile, a comprehensive user study will be conducted in the future to estimate the perceived user experience for different power settings.

B. User Model Training

In Section IV, we use the first month of the users' data for training the MDP model and test the model's effectiveness with the second month's data. We achieved good results because in our collected data, the users' behavior is consistent. In other words, their phone usage and mobility patterns do not exhibit large variation across different days. However, this consistent pattern may not work for all the cases. Users could have a progressive behavior changing due to sporadic life events (e.g. paper deadline, vacation), which our current model is not able to capture and adaptive to. In the future, we will implement *Boe* running as a background service to capture this temporally changing behavior. Similar to MobileMiner [18], the service will periodically mine the phone usage patterns and determine the optimal period to update the model based on most recent data. For example, to avoid sacrificing user experience and battery life, the model refresh could happen when the phone is being charged at night. However, we acknowledge that it is still very challenging for history-based models to capture those non-progressive outliers (user pattern significantly deviates from the average case) and often results in battery outage. Nevertheless, if these changes can be predicted by some clues such as from personal calendar, it is still promising to improve the model accordingly to minimize the number of battery outages.

C. Extensibility

In Section III, we mentioned that *Boe* can be easily extended to incorporate more hardware components as long as there are trades-offs exist between energy consumption and user experience. However, there are still some open problems regarding the definition of the user experience. For example, underclocking is the most straightforward way to reduce the energy consumption of the CPU. However, it might only work for non-CPU bound task such as text processing, but will significantly degrade user experience at an unknown rate for gaming or video streaming jobs. Regarding the network interface, it is straightforward to reduce the energy consumption by increasing network polling period for a background service task. However, for streaming tasks, the bandwidth (user experience) is generally affected by the received signal strength and hence there are not any options to be manipulated on the device to trade off energy usage and user experience. As studied in [7], energy consumption on network interfaces (3G and WiFi) is positively correlated with the network bandwidth but negatively with its wireless signal strength. Therefore, it is desirable to add a module which can automatically help the user navigate to the places with stronger signal strength to simultaneously reduce energy consumption and gain higher user experience.

D. Battery Outage and Opportunistic Charging

As mentioned earlier, *Boe* is designed to minimize the number of battery outage events. However, we note that it cannot guarantee the absence of outage events. Scenarios such as watching long online videos watch and long phone conversations are hard to optimize for. Rather than notifying the remaining battery level when there is only 10% or 20%, it will be more useful to provide an estimation of when the battery will run out to keep the users aware of the remaining battery life through this feedback channel. On the other hand, it will be valuable to incorporate a opportunistic charging remainder module into the system. This module will collect the locations where all the phone charging events took place, and remind users to proactively charge their phones when they visit the same places. By doing so, it will be particularly useful for the cases when the system predicts that the battery will deplete before the end of day.

VI. Conclusions

In this paper, we presented *Boe*, a Markov Decision Process (MDP) based global power management scheme balancing user experience and energy saving for mobile devices. We evaluated *Boe* through a field study with smartphone data from 10 users over 2 months and demonstrate that it outperforms fixed policies by eliminating battery outage time and also achieving as high a user experience as possible. Compared to the best fixed user experience policies, we show that: (i) *Boe* eliminates *all* frustrating battery outage events for light, moderate, and heavy phone users, and (ii) *Boe* improves user experience by 20% for light users, maintains the same user experience for moderate users, and degrades user experience by 23% for heavy smartphone users. Moreover, we show how *Boe* could easily scale to include new context information about user status, new power saving actions for controlling additional hardware.

REFERENCES

[1] Monsoon power monitor, http://www.msoon.com/LabEquipment/PowerMonitor.

[2] N. Aharony, W. Pan, C. Ip, I. Khayal, and A. Pentland. Social fmri: Investigating and shaping social mechanisms in the real world. *Pervasive and Mobile Computing*, 7(6), 2011.

[3] B. Anand, K. Thirugnanam, J. Sebastian, P. G. Kannan, A. L. Ananda, M. C. Chan, and R. K. Balan. Adaptive display power management for mobile games. In *ACM MobiSys*, 2011.

[4] R. Bellman. A markovian decision process. *Journal of Applied Mathematics and Mechanics*, 6, 1957.

[5] A. Carroll and G. Heiser. An analysis of power consumption in a smartphone. In *USENIX ATC*, 2010.

[6] D. Chu, N. D. Lane, T. T.-T. Lai, C. Pang, X. Meng, Q. Guo, F. Li, and F. Zhao. Balancing energy, latency and accuracy for mobile sensor data classification. In *ACM SenSys*, 2011.

[7] N. Ding, D. Wagner, X. Chen, Y. C. Hu, and A. Rice. Characterizing and modeling the impact of wireless signal strength on smartphone battery drain. In *ACM SIGMETRICS*, 2013.

[8] M. Dong and L. Zhong. Chameleon: a color-adaptive web browser for mobile oled displays. In *ACM MobiSys*, 2011.

[9] D. Ferreira, A. K. Dey, and V. Kostakos. Understanding human-smartphone concerns: a study of battery life. In *Pervasive*. 2011.

[10] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *ACM MobiSys*, 2003.

[11] R. A. Howard. Dynamic programming and markov processes. 1960.

[12] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *ACM SenSys*, 2010.

[13] S. Nath. Ace: exploiting correlation for energy-efficient and continuous context sensing. In *ACM MobiSys*, 2012.

[14] S. Nirjon, A. Nicoara, C.-H. Hsu, J. Singh, and J. Stankovic. Multinets: Policy oriented real-time switching of wireless interfaces on mobile devices. In *IEEE RTAS*, 2012.

[15] J. Paek, J. Kim, and R. Govindan. Energy-efficient rate-adaptive gps-based positioning for smartphones. In *ACM MobiSys*, 2010.

[16] N. Ravi, J. Scott, L. Han, and L. Iftode. Context-aware battery management for mobile phones. In *IEEE PerCom*, 2008.

[17] A. Shye, B. Scholbrock, and G. Memik. Into the wild: studying real user activity patterns to guide power optimizations for mobile architectures. In *ACM MICRO*, 2009.

[18] V. Srinivasan, S. Moghaddam, A. Mukherji, K. K. Rachuri, C. Xu, and E. M. Tapia. Mobileminer: Mining your frequent behavior patterns on your phone. In *ACM UbiComp*, 2014.

[19] L. Zhang, B. Tiwana, Z. Qian, Z. Wang, R. P. Dick, Z. M. Mao, and L. Yang. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In *ACM CODES+ISSS*, 2010.

[20] P. Zhou, Y. Zheng, Z. Li, M. Li, and G. Shen. Iodetector: A generic service for indoor outdoor detection. In *ACM SenSys*, 2012.

[21] Z. Zhuang, K.-H. Kim, and J. P. Singh. Improving energy efficiency of location sensing on smartphones. In *ACM MobiSys*, 2010.