Environment-Aware Estimation of Battery State-of-Charge for Mobile Devices

Liang He  
University of Colorado—Denver, CO  
liang.he@ucdenver.edu

Youngmoon Lee, Eugene Kim, Kang G. Shin  
University of Michigan—Ann Arbor, MI  
{ymoonlee,kimsun,kgshin}@umich.edu

ABSTRACT
Reliable operation of mobile devices, such as smartphones and tablets, has become essential for great many users around the globe. Mobile devices, however, have been reported to suffer from frequent, unexpected shutoffs — e.g., shutting off even when their batteries were shown to have up to 60% remaining state-of-charge (SoC) — especially in cold environments. Their main cause is found to be the inability of commodity mobile devices to account for the strong dependency between battery SoC and the environment temperature. To remedy this problem, we design, implement, and evaluate EA-SoC, a real-time Environment-Aware battery SoC estimation service for mobile devices. EA-SoC estimates the battery SoC by predicting the end-of-discharge battery resistance, grounding on (1) a thermal circuit model that describes the interactions among the battery’s discharge current, temperature, and the environment, and (2) an empirically validated data-driven model on the relations between battery temperature and resistance. We have conducted 35 experimental case-studies with two Nexus 5X smartphones to evaluate EA-SoC. EA-SoC is shown to report an average of 3% SoC when the phones shut off even in a $-15^\circ$C environment, while that reported by the phones’ built-in fuel-gauge chips could be over 90%.

ACM Reference format:
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION
Mobile devices have become essential to great many people, and their reliable operation depends strongly on the timely and accurate estimation of their battery state-of-charge (SoC). Mobile users, however, have frequently complained about the unexpected shutoffs of their devices due to inaccurate/untimely SoC estimation [2, 4, 5] — i.e., the devices were shut off even when their remaining power was shown to be up to 60% of full power [8], especially in a cold environment [6]. Fig. 1 illustrates such an unexpected shutoff of Xperia Z smartphone:

(i) video streaming in a $-15^\circ$C environment, the phone shut off prematurely when it was shown to have 30% state-of-charge (SoC); (ii) the phone, after shutting off, was moved to and kept in a room-temperature environment for 2 hours; (iii) the phone, without charging it, was turned back on again and operated for about 2 more hours. These premature shutoffs of mobile devices are due to the failure of their fuel-gauge chips to capture the environment-dependent battery performance, thus leading to erroneous estimation of their real-time SoCs. Cold temperature increases the internal resistance of batteries, thus degrading batteries' ability to deliver both the stored energy (i.e., the energy that can be delivered to operate the devices) and power (i.e., the maximum discharge power the battery can supply) [13, 15, 25, 30, 32]. When moving a mobile device from a warm to a cold environment, the fuel-gauge chip of the device...
cannot sense the change of the environment and thus cannot accurately predict the battery’s end-of-discharge condition. This, in turn, leads to the over-estimation of battery’s remaining power supply, i.e., SoC, and thus unexpected shutoffs. Note that unexpected shutoffs also risk deep-discharging and accelerate battery degradation [22, 31]. To address this problem, we design and implement EA-SoC, an environment-aware battery SoC estimation service for mobile devices that achieves accurate SoC estimation even in a cold environment. EA-SoC compensates the environment’s impact on battery SoC by predicting the battery’s end-of-discharge temperature in real time, and then estimating the end-of-discharge battery resistance according to an empirically captured resistance–temperature relationship of device batteries.

The end-of-discharge battery temperature, however, is affected by both device operation and ambient temperature. To meet this challenge, we analyze the thermal behavior of device batteries with a thermal circuit model, capturing the interactions among the battery’s discharge current, steady-state temperature, and ambient temperature. Specifically, the steady-state analysis of the thermal model facilitates prediction of battery temperature based on its discharge current and ambient temperature, and its transient-state analysis allows for estimation of ambient temperature based on battery’s recent discharge history. Combining these results, EA-SoC learns and updates the thermal characteristics of device battery, and then estimates, in real time, the battery SoC with environment-awareness.

We evaluate EA-SoC with two Nexus 5X smartphones and via 35 experimental case-studies, with [-15, 25]°C ambient temperature. The results show that EA-SoC reliably captures temperature effect on the remaining usable capacity of batteries, thus achieving accurate SoC estimation even in an environment as cold as −15°C — phones shut off when EA-SoC concludes they have only ≈3% remaining SoC. On the other hand, such end-of-discharge SoCs provided by the fuel-gauge chips of the phones are averaged at ≈50% for all of the 35 case-studies, indicating a ≈17x improvement by EA-SoC in accurately predicting phones’ shutoffs.

This paper makes the following main contributions.

- Empirically capturing the battery’s resistance–temperature relationship.
- Characterization of interactions among battery temperature, current, and environment.
- Design of EA-SoC, an environment-aware battery SoC estimation service for mobile devices.
- Evaluation of EA-SoC with 2 Nexus 5X smartphones, showing EA-SoC reports ≈3% end-of-discharge battery SoC even in a −15°C environment, while that reported by the phones’ built-in fuel-gauge chips could be over 90%.

The paper is organized as follows. The background and motivation are presented in Secs. 2 and 3, respectively. Secs. 4–6 detail the design, implementation, and evaluation of EA-SoC, respectively. The related literature is discussed in Sec. 7, and the paper concludes in Sec. 8.

2 BATTERY STATE-OF-CHARGE

Here we introduce the necessary background on the estimation of battery SoC.

2.1 Circuit Model of Mobile Devices

Fig. 2 illustrates the power architecture of mobile devices with a circuit model, consisting of the battery, the individual hardware components of the device, and a voltage regulator connecting them. The device battery is represented as a series connection of an ideal voltage source and its internal resistance $r$. The ideal voltage source provides a voltage that is commonly referred to as the battery’s open circuit voltage (OCV), defined as the voltage between its terminals when no loads/charger is connected. This way, the battery supplies a voltage of

$$V_{bat} = OCV - I \cdot r_{bat} \approx OCV - I \cdot r$$

(1)

Equation (1) is a polarization voltage which represents voltage transient over time (t) when it supplies current (I). Note that polarization voltage does not affect ohmic resistance and instantaneous voltage drop in the mobile application. Also, high-rate measurement in a mobile device can capture instantaneous voltage drop by limiting the influence of voltage transient. This allows for simplifying battery equivalent circuit model without the loss of voltage estimation accuracy. The voltage regulator then converts $V_{bat}$ to a required level to power various device modules, such as screen, CPU, etc.

The voltage regulator, however, needs an input voltage (i.e., $V_{bat}$ in Fig. 2) no less than a pre-defined level $V_{shutoff}$; otherwise, it will be unable to provide the required voltage to the device, and thus the device will shut off. According to Eq. (1), a mobile device will shut off when its battery OCV decreases to $V_{shutoff} + I \cdot r$, which is commonly referred to as the end-of-discharge OCV, i.e., $OCV_{end}$.

2.2 Estimation of Battery SoC

SoC quantifies the remaining capacity of a battery, defined as the ratio of battery’s remaining usable capacity to its full charge capacity, i.e.,

$$SoC = \frac{C_{remain}}{C_{full}} \times 100\%.$$  

(2)

Commodity mobile devices estimate their battery SoCs mainly based on the OCV–DoD relationship of their batteries. This is
because Lithium-ion batteries, the most widely used batteries for mobile devices, demonstrate a monotonic relationship between their OCVs and DoDs as shown in Fig. 3. This relationship is tested to be stable for batteries of the same chemistry and does not vary much with manufacturer (e.g., < 5mV variances in OCV with given DoD [1, 18]). We use $d = D(v)$ to refer to the mapping from battery OCV $v$ to DoD $d$ in the rest of the paper, and will elaborate on how to obtain the OCV–DoD relationship in Fig. 3 in Sec. 5.

Based on this OCV–DoD relationship, mobile device estimates its battery SoC by predicting its end-of-discharge OCV according to

$$OCV_{end} = V_{shutoff} + I_{end} \cdot r_{end}, \tag{3}$$

and then estimating the battery SoC as

$$SoC(t) = \frac{D(OCV_{end}) - D(OCV(t))}{D(OCV_{end})} \times 100\%. \tag{4}$$

Combination of Eqs. (1), (3), and (4) reveals the battery resistance $r$ (both current value and the one at the time at end-of-discharge) is needed to estimate battery OCV. Battery resistance is measured on mobile devices based on the principle of $r = \Delta V / \Delta I$ [26, 34], as illustrated in Fig. 4 with a battery of the Galaxy S3 phone. Switching the discharge current from 1A to 0.5A, the battery voltage recovers instantly from 3.61V to 3.66V, and then reduces gradually again due to continuous discharge. This way, a current change of $|1 - 0.5| = 0.5A$ causes the battery a voltage response of $|3.66 - 3.61| = 0.05V$, indicating a $0.05V / 0.5A = 0.1 \Omega$ battery resistance according to Ohm’s law. However, the two stable current levels (i.e., 1A and 0.5A in the above example) are needed to obtain a reliable $dI$, which are not always available on mobile devices due to their dynamic usage patterns, degrading both the availability and accuracy of real-time resistance.

Moreover, the battery’s internal resistance depends strongly on the temperature, complicating its estimation further. To shed more light on this, we log the battery resistance of a Nexus 5X smartphone after putting it in a freezer and until it shuts off. These experiments are repeated 3 times and a total number of 1, 263 pairs of battery resistance and temperature are collected, as plotted in Fig. 13: the temperature, complicating its estimation further. To shed more light on this, we log the battery resistance of a Nexus 5X smartphone after putting it in a freezer and until it shuts off. These experiments are repeated 3 times and a total number of 1, 263 pairs of battery resistance and temperature are collected, as plotted in Fig. 13: the resistance increases dramatically when its temperature drops below 5°C.

## 3 UNEXPECTED DEVICE SHUTOFFS IN COLD ENVIRONMENTS

Knowing the basics of batteries and the estimation of their SoC, we next examine why the mobile devices tend to prematurely shut off in cold environments.

Cold temperature slows down the chemical reactions inside the batteries, impeding the batteries to produce the same current and
deliver the same capacity as with warmer temperature. This chemical degradation is observed physically as the increase of battery’s internal resistance \([15, 25, 32]\), as empirically validated in Fig. 5. Also, such an increase of battery resistance in cold environments occurs gradually. Fig. 6 plots the resistance and temperature of a Nexus 5X phone’s battery after putting an idle Nexus 5X phone into a \(-12^\circ\text{C}\) environment: (i) the battery temperature reduces gradually to \(-11.8^\circ\text{C}\), and during the same process, (ii) the battery resistance gradually increases to about 0.66\(\Omega\). This gradually changing battery information, albeit intuitive, makes the estimation of end-of-discharge battery condition (i.e., Eq. (3)) non-trivial. Let us consider the scenario in which a mobile user is moving from a warm to a cold environment. The fuel-gauge chip of the mobile device will estimate the end-of-discharge condition of the device battery based on its current information, i.e., a resistance measured in a warm environment \(r_{\text{warm}}\) \([34]\). Such an estimation, however, is clearly inaccurate because the battery resistance will increase when it is moved to the cold environment, denoted as \(r_{\text{cold}}\) and \(r_{\text{cold}} > r_{\text{warm}}\). Thus, the device’s fuel-gauge chip will underestimate \(T_{\text{end}}\) in Eq. (3), leading to under-estimation of \(OCV_{\text{end}}\). This will, in turn, cause over-estimation of battery SoC according to Eq. (4), thus causing unexpected device shutoffs.

4 ENVIRONMENT-AWARE ESTIMATION OF BATTERY SOC

To mitigate the unexpected shutoffs of mobile devices in cold environments, we design a novel method called EA-SoC to estimate the battery SoC for mobile devices with the awareness of environment temperature.

4.1 EA-SoC Overview

EA-SoC estimates the battery SoC by (i) predicting the end-of-discharge battery temperature, (ii) estimating the end-of-discharge battery resistance based on the thus-predicted temperature, and then (iii) estimating battery SoC based on Eqs. (3) and (4). Fig. 7 provides the flow chart of EA-SoC.

4.2 Predicting the End-of-Discharge Battery Temperature

EA-SoC predicts the end-of-discharge battery temperature by exploiting the interplays between battery temperature and that of the ambient environment. Note that same as with phone’s built-in fuel-gauge chips, EA-SoC assumes the recent discharge current of the battery will be kept stable until shutoff \([34]\).

- **Battery’s Thermal Model.** Battery temperature \(T_{\text{bat}}\) is jointly determined by device operation and ambient environment \(T_{\text{amb}}\), as shown in Fig. 8 with a thermal circuit model. According to Fourier’s law, a thermal circuit is analogous to an electric circuit where voltage represents temperature \((T_{\text{bat}})\), current source represents heat generation \((P_{\text{bat}})\), and thermal resistance and capacitance \((R_{\text{th}}, C_{\text{th}})\) describes how battery dissipates heat. Since heat conserves, heat generation from the battery must equate to the heat dissipation through thermal capacitance and resistance:

\[
C_{\text{th}} \cdot \frac{dT_{\text{bat}}(t)}{dt} + \frac{T_{\text{bat}}(t) - T_{\text{amb}}}{R_{\text{th}}} = P_{\text{bat}},
\]

where the left-hand side is heat dissipation and the right-hand side is heat generation from the battery. Eq. (5) can be further divided into two cases based on \(dT_{\text{bat}}(t)/dt\): the battery will be in a stable thermal state when \(dT_{\text{bat}}(t)/dt = 0\), and in a transient thermal state otherwise. Fig. 6 highlights the two-state thermal behavior of batteries. EA-SoC estimates, and update in real time, the (to-be-converged) stable-state battery temperature based on recent battery information, and use it as the battery temperature at the end-of-discharge to estimate the battery SoC.

- **Estimating Battery Temperature via Stable-State Analysis.** Eq. (5) facilitates the identification of battery’s steady-state temperature, i.e., the equilibrated battery temperature when \(dT_{\text{bat}}(t) = 0\).
0. Specifically, given stable device operation and ambient environment, battery’s steady-state temperature $T_{bat}$ can be calculated as

$$\frac{dT_{bat}(t)}{dt} = 0 \iff T_{bat}(t) - T_{amb} = \frac{P_{bat}}{R_{th}}$$
$$\iff T_{bat} = T_{amb} + \frac{P_{bat}}{R_{th}}. \quad (6)$$

We can see that (i) $T_{bat}$ is linear to battery’s heat dissipation $P_{bat}$, and (ii) the ambient temperature $T_{amb}$ contributes to $T_{bat}$ as an offset.

Battery’s heat generation $P_{bat}$ can be modeled by the Joule’s law, and according to Eq. (11), we know

$$P_{bat} = I^2 \cdot r_{bat} = I^2 \cdot (a_1 \cdot e^{b_1 \cdot T_{bat}} + c_1 \cdot e^{d_1 \cdot T_{bat}}). \quad (7)$$

Substituting Eq. (7) into Eq. (6), we get

$$T_{bat} = T_{amb} + R_{th} \cdot I^2 \cdot (a_1 \cdot e^{b_1 \cdot T_{bat}} + c_1 \cdot e^{d_1 \cdot T_{bat}}), \quad (8)$$

which can be used to estimate the stable-state battery temperature based on that of the ambient environment.

Next, we empirically validate these observations with a Nexus 5X smartphone. Specifically, we use an Android app called BatteryDrainer to regulate the phone operation and thus achieving a (relatively) stable phone operation but with a controllable intensity. The phone, with the thus-regulated operation, is then put in a thermal chamber to achieve controllable ambient temperature, as shown in Fig. 9. We conducted 27 such experiments with $[−13, 30]°C$ ambient temperature, during which the battery temperature and current are logged at 1Hz. Each experiment lasts at least 1 hour which is observed to be long-enough for battery temperature to equilibrate. Fig. 10 plots the experiment results, showing (i) clear linearity between $T_{bat}$ and $I^2$ (i.e., the key factor in battery’s heat generation) and (ii) $T_{amb}$’s contribution to $T_{bat}$ as a temperature offset, validating the analytical observations in Eq. (8). Also note that all other parameters in Eq. (8) can be identified on mobile devices in practice: $R_{th}$ and $<a_1, b_1, c_1, d_1>$ can be identified based on collected training samples, and $I$ can be accessed from the devices’ fuel-gauge chips.¹

¹Current sensing is now pervasively supported by the fuel-gauge chips of mobile devices, although not so a few years ago [33].

---

¹To the best of our knowledge, only a few existing device models support ambient temperature sensing, such as Galaxy S4 and Note 3.
of battery’s thermal behavior, which, when combined with Eq. (8), facilitates the estimation of $T_{amb}$. From Eq. (5), the transient-state battery temperature $T_{bat}(t)$ can be described as

$$T_{bat}(t) = (T_0 - (T_{amb} + R_{th} \cdot P_{bat}))) \cdot e^{\frac{-t}{t_{bat}}} + T_{amb} + R_{th} \cdot T_{bat}.$$  

(9)

where $T_0$ is battery’s initial temperature, and the thermal-time constant, $R_{th} \cdot C_{th}$, represents the time duration required for the battery temperature to stabilize. Note that the second term captures the steady-state temperature while the first term corresponds to the effects of initial temperature, which decays exponentially over time.

Eq. (9) implies the equilibrating process of battery temperature conforms to an exponential process, i.e.,

$$T_{bat}(t) = a_2 \cdot e^{b_2 \cdot t} + c_2,$$  

(10)

for certain $<a_2, b_2, c_2>$. As a validation, Fig. 11 plots the results when fitting an empirically collected temperature equilibrating process according to Eq. (10), observing an excellent match. Fig. 12 plots the CDF of the goodness-of-fit (in terms of AdjRsquare) when exponentially fitting 34 such temperature equilibrating processes, validating the soundness of the exponential model as a close-to-1 AdjRsquare indicating promising fitting goodness.

Such an exponential equilibrating process allows EA-SoC to predict the steady-state battery temperature based on its recent temperature traces, if the device has operated stably for a certain amount of time. EA-SoC determines the stable device operation based on its recent current. Specifically, EA-SoC smooths the recent (e.g., 10 minutes) current trace via moving average, and then applies linear fitting on the smoothed current trace, concluding a stable device operation if the resultant slop factor is small, e.g., $<0.1$. In case of stable device operation, EA-SoC trains the exponential model in Eq. (10) based on the recent 10-minute battery temperature trace, i.e., identifying $<a_2, b_2, c_2>$, and using it to predict the steady-state battery temperature, i.e., when $dT_{bat}/dt \leq 1^\circ C/5min$. The thus-estimated steady-state battery temperature is then used to estimate the ambient temperature according to Eq. (8). We will experimentally explore the required duration of stable device operation, e.g., 10 minutes in the above explanations, in Sec. 6.

### 4.3 Estimating the End-of-Discharge Battery Resistance

EA-SoC then estimates the end-of-discharge battery resistance based on the above predicted end-of-discharge battery temperature. As shown in Fig. 5, batteries’ resistance increases with the decrease of temperature, reducing their usable capacity. This explains why commodity mobile devices operate reasonably well in warmer environments (i.e., with relatively stable resistance), but frequently suffer from unexpected shutoffs in a cold environment (i.e., due to increased resistance). Moreover, these empirically-collected samples reveal a 2-term exponential relationship between battery resistance $r_{bat}$ and temperature $T_{bat}$, i.e.,

$$r_{bat}(T_{bat}) = a_1 \cdot e^{b_1 \cdot T_{bat}} + c_1 \cdot e^{d_1 \cdot T_{bat}}.$$  

(11)

Inspired by such a resistance–temperature relationship, EA-SoC identifies $<a_1, b_1, c_1, d_1>$ in Eq. (11) based on empirically collected samples of $<r_{bat}, T_{bat}>$, and then estimates the end-of-discharge battery resistance based on the previously predicted the end-of-discharge battery temperature. Four pairs of different $<r_{bat}, T_{bat}>$ are needed to determine $<a_1, b_1, c_1, d_1>$. To enhance reliability and reduce fluctuation, EA-SoC determines $<a_1, b_1, c_1, d_1>$ based on the most recent $n (n \geq 4)$ different pairs of $<r_{bat}, T_{bat}>$ — calculating $<a_1, b_1, c_1, d_1>$ based on each of $C_n^4$ possible combinations of logged samples, and describing the resistance–temperature relationship with the averaged $<a_1, b_1, c_1, d_1>$. Also, EA-SoC only updates $<a_1, b_1, c_1, d_1>$ when the device is fully charged and the charger is kept connected, which (i) offers reliable conditions to estimate battery resistance [16] and (ii) does not incur any additional overhead in battery energy consumption due to the connected charger and thus the existence of external power supply.

Note that we implicitly assume a uniform temperature distribution across the battery, i.e., describing battery temperature with a single value $T_{bat}$. This assumption is reasonable as most mobile devices use single-cell batteries which are of much smaller form factors than large multi-cell battery packs, e.g., those for EVs, although Lithium-ion batteries are known to have complex thermal behaviors due to their non-uniform temperature distributions and non-linear chemical reactions. Such a simplification is also widely used in practice — to the best of our knowledge, all commodity mobile devices use a single thermal sensor to monitor their battery temperature, and thus represent their battery temperature with a single temperature reading.
Algorithm 1: Pseudocode of EA-SoC.

1. initializing \(<a_1, b_1, c_1, d_1>\) and \(T_{amb}\);
2. while true do
   3. estimate steady-state battery temperature according to Eq. (8) with \(J\) and \(T_{amb}\);
   4. estimate end-of-discharge battery resistance according to Eq. (11);
   5. estimate end-of-discharge OCV according to Eq. (3);
   6. estimate remaining usable capacity and thus SoC;
   7. estimate SoC via Coulomb counting;
   8. if stable operation for the past 10 minutes then
      9. predict steady-state battery temperature \(T_{bat}\) according to Eq. (10);
      10. update \(T_{amb}\) according to Eq. (8);
   end if
   11. if phone is fully charged and the charger is still connected
       12. update \(<a_1, b_1, c_1, d_1>\);
   end if
   13. if \(time = time + 1\);
end while

4.4 Estimating Battery SoC with Temperature Awareness

At last, EA-SoC estimates the end-of-discharge battery OCV based on the predicted \(\hat{r}_{end}\) according to Eq. (3), and then further estimates the battery SoC based on Eqs. (1) and (4).

4.5 EA-SoC Summary

Alg. 1 summarizes EA-SoC in the form of pseudocode. EA-SoC estimates the steady-state battery temperature based on the discharge current and ambient temperature with current and ambient temperature (i.e., Eq. (8) and line 3), which is then used to predict the end-of-discharge battery resistance according to Eq. (11) (line 4). This way, the end-of-discharge OCV can be calculated as in Eq. (3) (line 5). EA-SoC then estimates battery SoC based on the OCV–DoD relationship as in Eq. (4) (line 6). EA-SoC updates the knowledge of (i) ambient temperature \(T_{amb}\) when the device has been working reliably for, e.g., 10 minutes (line 9-10), and (ii) the resistance-temperature model after the device has been fully charged and the charger is still connected (line 13).

5 IMPLEMENTATION

Given below are a few details of EA-SoC’s implementation.

5.1 Logging of Battery Information

EA-SoC monitors and logs real-time information on battery current and temperature, based on which the device’s SoC is estimated with environment-awareness. This battery information can be obtained from the devices’ fuel-gauge chips, e.g., the system files of current_now and temp under directory
/sys/class/power_supply/battery/ for Nexus 5X and Nexus 6P. The real-time logging of such information, however, incurs energy consumption to the device. So, EA-SoC is desired to collect battery information at a low frequency while ensuring sufficient SoC estimation accuracy. Note that the fuel-gauge chips of mobile devices easily support up to 1 Hz (or higher, depending on specific device models) sampling rates of battery information, which, for example, can be adjusted on the Android platform by configuring the fuel-gauge driver at drivers/power/qpnp-fg.c.

We empirically identify such proper sampling rates on battery temperature and current. First, we examine the frequency for battery temperature to change based on the 34 temperature equilibrating processes as used in Fig. 12, which are originally collected at 1 Hz sampling frequency. This way, we find that the battery temperature changes every 8.9 s, on average, based on a total of 17,212 samples. Second, the current information is needed to quantify the device’s energy consumption. To identify the necessary current sampling rate for accurate fuel gauging, we collected a 12-minute current trace from a Galaxy S5 phone with the Monsoon power monitor running at 5 kHz, as shown in Fig. 14, during which 114 mAh capacity is discharged. Using this trace, we calculate the total energy consumption when emulating different current sampling rates of 1 Hz, 1/10 Hz, 1/30 Hz, and 1/60 Hz, achieving the discharged capacity of 115 mAh, 113 mAh, 109 mAh, and 98 mAh, respectively — a 1/10 Hz sampling rate is able to achieve over 99% fuel-gauge accuracy, as shown in Fig. 15. Combining the above empirical observations, EA-SoC monitors and logs battery temperature and current once every 8 s.
5.2 Identifying the OCV–DoD Relationship

EA-SoC needs the OCV–DoD relationship of device battery to estimate its DoD, which is collected via offline training. Given a specific model of mobile device, e.g., Nexus 5X, we use the battery tester as in Fig. 16 to discharge its battery with 200mA current and log the process at 1Hz, collecting traces on the relationship between the battery terminal voltage and its DoD. We then perform resistance compensation on the thus-collected traces based on Eq. (1) to derive its OCV–DoD table. The small current of 200mA is to reduce the $I \cdot r_{bat}$ voltage and thus improve the accuracy of the derived OCD–DoD table. Fig. 17 shows the thus-collected OCV–DoD relationship of a Nexus 5X battery. The OCV–DoD relationship shown in Fig. 3 is collected similarly.

6 EXPERIMENTS

We have evaluated EA-SoC extensively with two Nexus 5X smartphones.

6.1 Accuracy in Predicting Battery Temperature

We first validate the reliability of predicting battery temperature based on its exponential equilibrating process, i.e., Eq. (10). Specifically, we collected 42 temperature equilibrating processes of a
Nexus 5X phone’s battery, while keeping the phone in stable operation conditions. These traces last for [18, 190] minutes and the battery temperature varies in the range of [−9.5, 43]°C. We then use the first x minutes of these traces to train the exponential equi-librating model in Eq. (10), and then use the thus-trained models to predict the battery temperature at the end of their respective traces.

Fig. 18 summarizes the prediction errors when the first 18-minute portion of each trace is used for training, showing within ±2°C errors for all the 42 traces and an average error of 0.366°C. To further examine how long a stable operation is needed to train a reliable model, Fig. 19 plots the average prediction error for the 42 traces when training with the first x-minute traces, together with their 5- and 95-percentiles5 — higher prediction accuracy could be achieved by training with longer traces, while a 10-minute training trace is enough to achieve less than 1°C prediction error.

6.2 Accuracy of SoC Estimation
We next validate EA–SoC’s accuracy in estimating battery SoC with 35 case-studies. The thus-estimated environment-aware SoCs are then compared with those provided by the phones, further validating EA–SoC’s reliability in predicting phones’ shutoffs. Specifically, in each of these case-studies, the phones are used for either Youtubing, or playing an offline video, or kept idle but with screen on, until it shuts off. The experiments start with a [37, 100]% phone battery SoC, and are conducted in a [−15, 28]°C environment. The real-time SoCs estimated by EA–SoC, together with those provided by the phone’s fuel-gauge chips, are logged. Note that a closer-to-0 end-of-discharge SoC indicates higher reliability in predicting the phone’s shutoffs.

Fig. 20 plots the battery voltage, current, temperature, and estimated SoC during one of such case-studies. Specifically, the fully-charged phone is put into a −13°C environment, and operates without human interactions until it shuts off, at which time its fuel-gauge chip provides an SoC estimation of about 67% — i.e., the phone shuts off unexpectedly. This case-study finishes with a 3.66V end-of-discharge battery voltage, which is much higher than the usual level, e.g., [3.2, 3.4]V [17], owing to the increased battery resistance at a low temperature. On the other hand, EA–SoC captures the decreased battery temperature in real time, and compensates the SoC estimation accordingly — the phone shuts off when EA–SoC concludes its battery only has a 3.8% SoC, showing EA–SoC’s ability to predict the phone’s shutoff accurately.

Fig. 21 summarizes the end-of-discharge SoCs collected in all the 35 case-studies, together with, and sorted according to, their end-of-discharge battery temperature. The phones shut off with high and random phone-provided SoCs when their batteries’ end-of-discharge temperature is low — they are high because the low temperature and thus increased battery resistance, and they are random because the phones operate with different currents. By capturing the battery’s resistance–temperature dependency, EA–SoC is able to predict the phones’ shutoffs much more accurately, achieving a 3% averaged end-of-discharge SoC for the 28 case-studies with end-of-discharge battery temperature lower than 1°C. Also, EA–SoC performs well in a warmer environment and achieves similar end-of-discharge SoCs as the phones’ fuel-gauge chips, as observed in the 29th–35th case-studies.

7 RELATED WORK
Many SoC estimation methods have been proposed in the literature [14, 19, 20], including the OCV-based methods [21, 24], Coulomb counting [23], neural network methods [11], and various Kalman filter-based methods [9, 12, 35]. For example, He et al. [18] proposed a model to eliminate the effects of current drift in the current sensor, improving Coulomb counting accuracy. A method to improve re-initialization of Coulomb counting is developed in [29]. A multi-cell battery pack SoC estimation considering cell imbalance is proposed in [37]. Trinh [28] explored ways to validate the SoC estimation accuracy. An empirical study of comparison of various SoC estimation methods can be found in [10].

Of these existing SoC estimation methods, combining the OCV-based method and Coulomb counting to estimate real-time battery SoC is the most widely used for commodity mobile devices.

5 Only the traces with a duration longer than x minutes are used for validation.

---

**Figure 21**: Environment-aware SoC estimation with 35 case-studies: EA–SoC achieves (i) much more reliable SoC estimation in a cold environment, and (ii) comparable results in a warmer environment, compared to the existing SoC estimation on Nexus 5X smartphone.
thanks to its simplicity and reasonably good accuracy. Examples of such deployment include TI’s Impedance Track [34] and Maxim’s MAX17847/17850 fuel-gauge chips [3], to name a few. However, the dependency between battery performance and temperature is still not covered well for commodity mobile devices, causing them to shut off unexpectedly in a cold environment [6].

To address such deficiency, we have proposed EA-SoC, an environment-aware SoC estimation service for mobile devices, achieving accurate SoC estimation even in cold environments. To the best of our knowledge, the closest to EA-SoC is [27], which also considers battery temperature in its SoC estimation. The method proposed therein, however, requires various fundamental battery properties such as solid phase diffusion coefficient and electrolyte diffusion coefficient, which are not available on mobile devices due to the limited hardware support.

8 CONCLUSIONS

In this paper, we have designed, implemented, and validated EA-SoC, an environment-aware battery SoC estimation service for mobile devices. EA-SoC captures the resistance–temperature relationship of device batteries with an empirically established regression model, and estimates battery SoCs based on the thermal interactions among battery’s discharge current, steady-state temperature, and ambient temperature. Such interactions are uncovered by a thermal circuit model of the battery and validated experimentally. We have evaluated EA-SoC with two Nexus 5X smartphones, achieving ∼3% end-of-discharge SoC even in a −15°C environment and improving reliability in predicting device shutoffs by 17x.

REFERENCES

[3] [n. d.]. MAX17847/MAX17850 ModelGauge m3 Fuel Gauge.
[26] Liang He and Youngmoon Lee, Eugene Kim, Kang G. Shin. 2014. For commodity mobile devices, causing them to shut off unexpectedly in a cold environment [6].