Always-On Quick Charging for Mobile Devices

Daeyong Kim, Seunghyeok Jeon, Seokjun Lee, and Hojung Cha
Department of Computer Science, Yonsei University, Seoul, Korea
{dy.kim, sh.jeon, sjlee, hjcha}@yonsei.ac.kr

***Abstract*—Mobile users are always demanding extended availability in their battery use. Together with enlarged battery capacity, fast charging is one approach that provides an improved user experience in battery use. Recently, device manufacturers have been developing a variety of fast charging techniques for mobile devices. However, the existing techniques severely reduce the charging power when the device is in use while charging. We experimentally demonstrate that the primary cause of the reduction in the charge speed during device use is to cope with the performance degradation incurred by heat generation. We then propose an adaptive charging scheme, called Always-on Quick Charging, which enables fast charging especially when the device is in use. The key idea of our approach is to adjust the charging power while ensuring that the heat generated by the charging does not affect the performance. The proposed scheme is implemented in Google's Pixel 2XL smartphone. The experiment with a real-world usage scenario shows that the charging speed of the proposed scheme is up to 2.4 times faster than the default scheme, while preserving device performance.**

***Index Terms*—Mobile devices, battery lifetime, fast charging, thermal management**

I. Introduction

As mobile devices increasingly consume more energy, users experience situations where the battery does not last more than a day. Users often charge their devices several times a day, and even carry a portable power bank for handy charging. As part of a solution for the battery problem, device manufacturers have been offering fast charging technologies, such as Quick Charge [1], Fast Adaptive charging [2], and Power Delivery [3]. Samsung's Fast Adaptive Charging is, for example, reported to charge 50% of the Galaxy S7's 3000 mAh battery within 30 min. Although the fast charging technology helps users cope with their battery problem, mobile users often use their devices aggressively, for example, playing heavily loaded 3D games. The power demand is so high that some users even carry a phone holder to charge the battery while using the device.

Recently, we observed the situation in which the charging speed slows down especially when the device is in use. This issue has not been reported in previous studies or related documents [7-10]. Our preliminary observation was that the existing charging method exercised in commercial devices reduces the charging power excessively when the device is in use. We further analyzed this phenomenon based on our understanding of the charging path in mobile devices, and found that this is caused by the device manufacturer's specific charging control which charges the device with a fixed power. In general, the charging power of a mobile device is controlled to prevent performance degradation caused by the temperature rise of the device [4,5]. However, since the charging power is controlled without information about the device usage, charging is conducted inefficiently. This inefficiency slows down the charging speed even when the device screen is only turned on and not doing any work.

In this paper, we address the inefficiency of the existing charging method and propose a charging scheme, called Always-on Quick Charging, to provide fast charging especially while the device is in use. The key idea is to raise the charging power to the maximum possible level without affecting the device performance. To this end, we should first understand how the charging rate affects the device performance. Adjusting the charging rate has a direct impact on performance; thus, the charge control should not degrade users' quality of experience. We should then consider an actual implementation that regulates the charging power in software. This can be done through kernel driver modification of the power management integrated chip (PMIC) inside the mobile device.

The contributions of our work are summarized as follows:

- Our work is the first to analyze the charging path of mobile device and address the inefficiency of quick charging, especially when device is in use.
- We developed a simple and yet pragmatic charging control mechanism to maximize the charging efficiency of mobile device without performance degradation.

The remaining sections detail the motivational experiments, followed by a description of the proposed Always-on Quick Charging scheme.

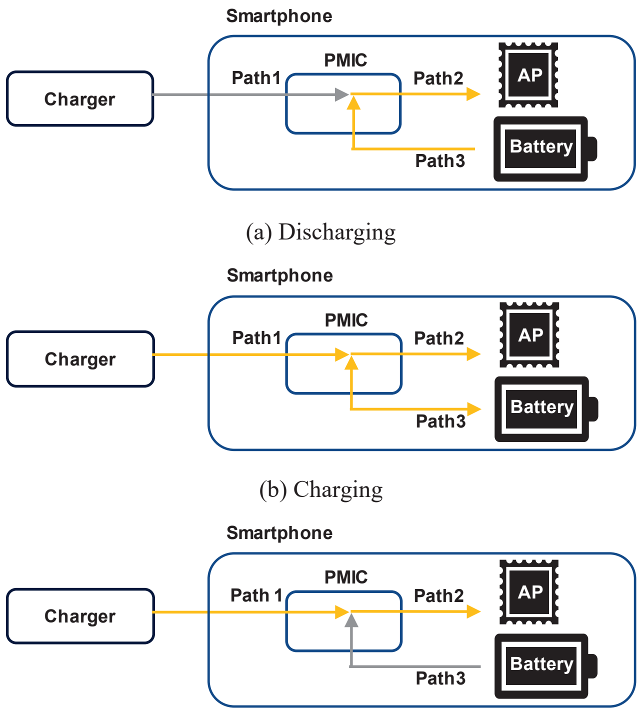
II. PRELIMINARY EXPERIMENTS

We conducted preliminary experiments to understand the charging path of commercial smartphones and show there is room to improve the charging efficiency. We also analyzed the cause of this inefficiency in existing charging mechanisms.

A. Current Paths

Before performing charging-related experiments, we tried to figure out what charging paths are used in smartphones. For this purpose, we identified the hardware parts in mobile devices. Fig. 1 illustrates the current paths of typical smartphones where the yellow line identifies the current flow. When the phone is discharged without charging (Fig. 1(a)), the PMIC draws current from the battery via Path3 and provides it to the application processor (AP) through Path2. When charging proceeds, the power from the external charger flows into the PMIC through Path1. The external power is then consumed by the AP via Path2, while the residual power is flown into the battery through Path3 (Fig. 1(b)). When the kernel blocks battery charging in the case of a full charge, the PMIC does not charge the battery and provides current directly to the AP via Path2 (Fig. 1(c)). In this

978-1-5386-9148-9/19/$31.00 ©2019 IEEE



(a) Discharging

(b) Charging

(c) Skip battery charging

Fig. 1: Current paths inside a smartphone.

paper, we focus on Path3 with which the charging efficiency is determined. Direct control of Path1 is difficult because we cannot control the charging efficiency accurately due to the dynamic power consumed by the AP. Moreover, when the current of Path1 is restricted less than the AP's power consumption, the battery will be discharged. This will severely degrade the user experience because the user does not expect the battery to be discharged while charging.

B. *Charging Speed while Using Device*

We experimentally observed the charging characteristics of the latest smartphones, especially when the device is in use. Five different devices listed in Table 1 were used for the experiment. All the devices use a charging voltage of 9 V and are capable of charging the device with a high power of 10–12 W. We recorded the charging power of Path3 in Fig. 1 by monitoring the PMIC inside the smartphone.

Fig. 2 shows the results for all five devices, when the screen is simply turned on and off. As soon as the screen was turned on, the charging power decreased immediately in all devices. The reduced power was then maintained at a constant level while the screen was on. This charging behavior is not efficient because the device's status does not change immediately after the screen is turned on. The device was doing nothing while the screen was on, consuming less than 2 W; therefore, an additional 5 W of power could have been used for charging in this case. Table 1 summarizes the average charging power for the five devices. In the case of the V20, the charging power for screen-on decreased considerably, about 81% of the screen-off case. The situation was similar in other devices, providing 39–47% of the maximum charging power. This experiment clearly reveals that the fast

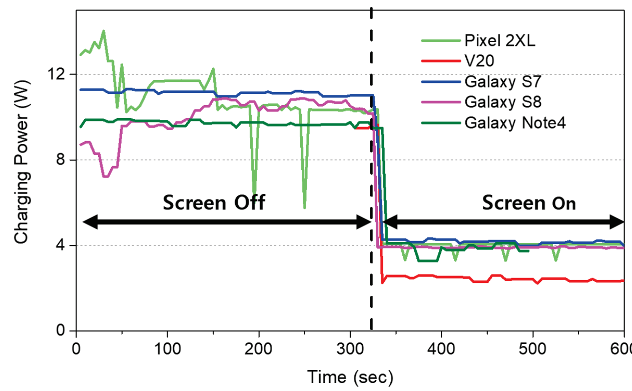


Fig. 2: Trace of charging power for various devices.

TABLE I
CHARGING POWER DIFFERENCES

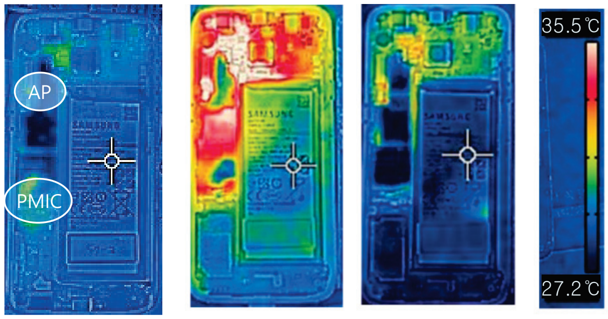
Device	Charging Power (W)		Reduction Rate (%)
	Screen On	Screen Off	
LG V20	2.1	10.8	81
Pixel 2XL	4.3	11	61
Galaxy S7	4.2	10	58
Galaxy S8	4	9.2	57
Galaxy Note 4	4.3	9	53

charging schemes employed in commercial devices are overly inefficient especially when the device is in use.

C. *Device Temperature while Charging*

With the previous experiment, we found that using a device while charging it greatly reduces the charging power. We tried to understand the cause of this phenomenon, based on the previous research claiming that the battery temperature of a mobile device affects the AP temperature [4]. Our hypothesis is that charging increases the temperature of the battery as well as the AP, as hinted in previous studies. Thus, the system reduces the charging power aggressively to control the temperature. We conducted an experiment to confirm this hypothesis. Three kinds of charging schemes were evaluated for the analysis. The first scheme is simply not using the charger (Fig. 1(a)). The second is to charge battery while the charger is plugged into the device (Fig. 1(b)). The third is not charging battery while the charger is plugged into the device (Fig. 1(c)). We just turned on the screen without any workload running on the device and observed how the temperature changed. For the experiment, we used the Galaxy S7 smartphone with the back plate removed. The temperature was recorded using a thermal camera.

Fig. 3 shows the device's surface temperature measured 10 min after the start of experiment. In comparison with Fig. 3(a), the AP temperature increases in Fig. 3(b) where the battery is partly being charged. In fact, the AP temperature rose, but the temperature of the battery did not rise meaningfully. Interestingly, we observed that the temperature of the PMIC component was rising rather than the battery itself. Since the charging voltage is 9V, which is too high for both battery and AP, the regulator inside of PMIC lowers the voltage to drive the components. The battery and other hardware components of AP require different voltages; hence, the regulator generates heat



(a) Discharging (b) Charging (c) Skip battery charging

Fig. 3: Surface temperature of the Galaxy S7.

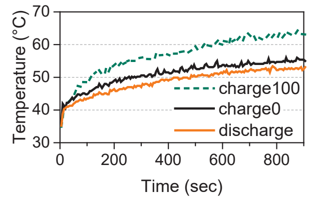
when the current is flown to both AP and the battery simultaneously (i.e., Path2 and Path3 in Fig. 1). This analysis was backed up by the result shown in Fig. 3(c), where the battery is not charged while the charger is plugged in. The heat generation in Fig. 3(c) is decreased, in comparison with Fig. 3(b), and this is because the path from the PMIC to the battery is not used, and the heat generation is reduced accordingly.

In summary, as power passes through two paths of charging the battery and the AP, the PMIC generates heat which affects the overall temperature of the device. Consequently, the AP temperature will rise when the device is in use while charging.

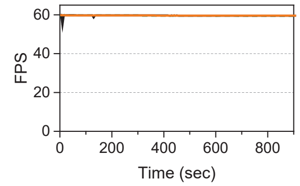
D. Charging Speed vs. Performance

The previous experiment confirmed that using a device during charging generates heat to the PMIC and the AP. In general, heat affects the performance of a mobile device. When the AP's temperature reaches a certain point, the kernel throttles the device's performance to a lower temperature. We designed and conducted an experiment to understand the relationship between the AP's temperature change and the performance of the device. The experiments were conducted with three different charging schemes. The first scheme is charge0, which charges the device while not charging the battery. The second scheme is charge100, which charges the device while charging the battery at full capacity. The third scheme is discharge, which uses only the internal battery without using the charger. The experiment was conducted with the Google Pixel 2XL, where charge100 corresponds to 12 W of charging power. For the application load generator, we used a benchmark program called Aquarium with which the degree of the workload can be controlled. We used two workloads, normal workload and thermal workload. The normal workload is light weight and consumes average 2.9 W for our target device, and maintains low device temperature. The thermal workload consumes about 4.3 W, and increases temperature significantly. The workload leads to thermal throttling and degrades device performance severely. Meanwhile, FPS (frames per second) was used as a metric of the user's quality of experience, because the smoothness of the display is a critical factor of the user experience. Graphic-related work on smartphones uses a lot of computing resources; thus, if the performance deteriorates due to heat, the FPS will drop immediately.

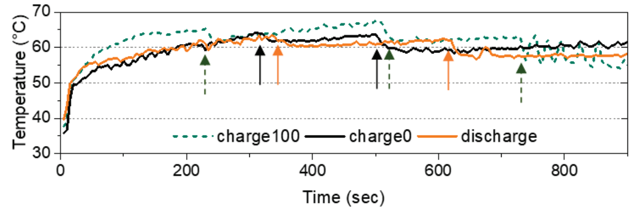
Fig. 4 shows the trace of the temperature and the FPS with both normal load and thermal load. We measured the FPS via



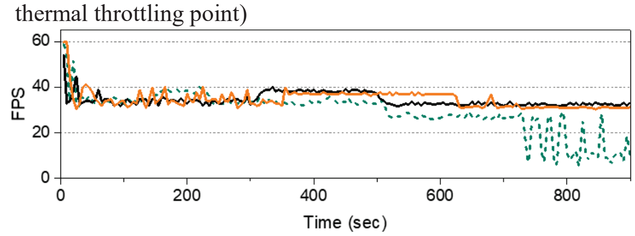
(a) Temperature with a normal workload



(b) FPS with a normal workload



(c) Temperature with a thermal workload (arrows indicate the thermal throttling point)

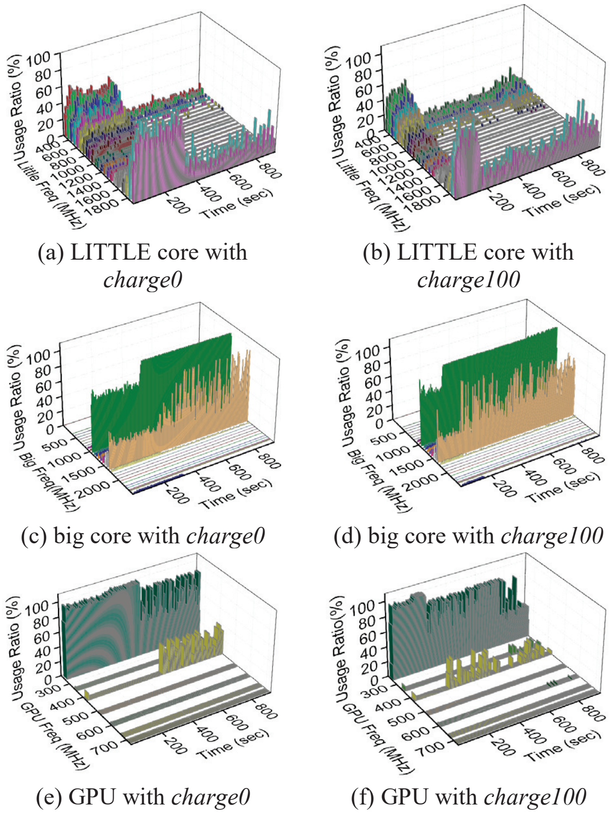


(d) FPS with a thermal workload

Fig. 4: Temperature and FPS trace.

the sys file system (/sys/class/graphics/fb0/measured_fps). In the case of a normal workload, charge0 raised the temperature slightly more than discharge, while the temperature was much higher for charge100. As the workload was light, thermal throttling was not performed in all cases, maintaining the FPS at 60. However, in thermal workload, thermal throttling indicated by the arrows in the figure occurred even in the discharge case. Specifically, in charge100, throttling arose frequently, and the FPS was degraded considerably at the end of experiment. We consider that due to continuous temperature increments, a strong level of throttling was applied at around 700 s.

We further analyzed the CPU and GPU frequency to confirm that thermal throttling was actually activated. Since Pixel 2XL uses Qualcomm Snapdragon 835 AP which is based on the ARM's big.LITTLE processor architecture, we plotted the frequency traces of the big and LITTLE core separately. Fig. 5(a,c,e) and (b,d,f) show the frequency trace of charge0 and charge100, respectively. We observed that CPU and GPU throttling occurred separately. In the case of the CPU, throttling occurred earlier with charge100, degrading the FPS. In both cases, the occurrence of throttling reduced the use of the LITTLE cluster, and the low frequency in the big cluster was used more often. Similarly, GPU throttling in charge100 occurred earlier than charge0. Moreover, it seems that the strong throttling shown in Fig. 4(d) was caused by GPU throttling. Here, we calculated the GPU's workload as $\sum_F (F^3 \times U_F)$ where F and U_F represent the frequency and the usage of frequency F, respectively [5]. The workload was expressed as a relative value with the maximum value of 100. Fig. 6 confirms that GPU throttling degraded the FPS considerably at the end of the experiment. In conclusion, the charging rate influences the



(a) LITTLE core with *charge0* (b) LITTLE core with *charge100*

(c) big core with *charge0* (d) big core with *charge100*

(e) GPU with *charge0* (f) GPU with *charge100*

Fig. 5: Frequency traces with thermal workload.

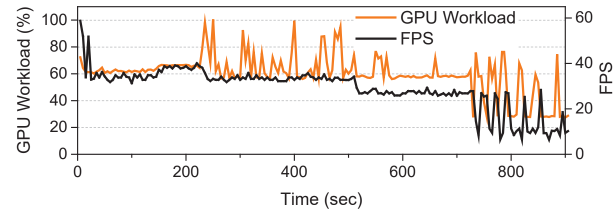
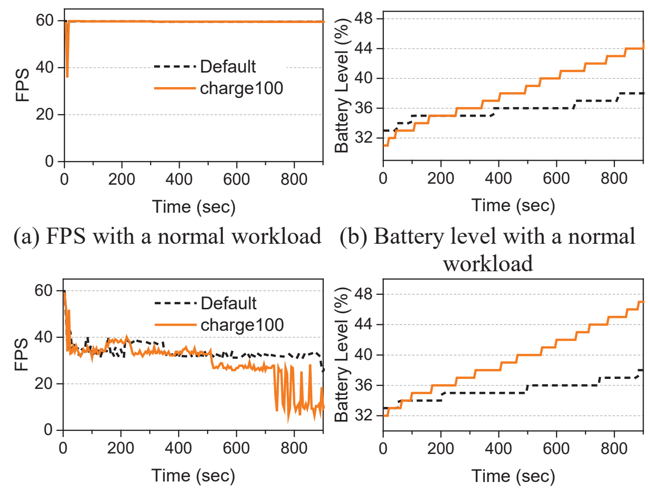


Fig. 6: GPU workload trace.



(a) FPS with a normal workload (b) Battery level with a normal workload

(c) FPS with a thermal workload (d) Battery level with a thermal workload

Fig. 7: FPS and Battery level trace.

temperature of AP, and performance is degraded depending on the temperature due to CPU and GPU throttling.

E. Commercial Devices

We looked at how the performance of a device is affected by the manufacturers' default charging policy. We specifically compared the FPS and the battery level of the default charging with charge100. The experimental setup is the same as in Section 2.4.

Fig. 7(a) and (b) show the results with the normal workload. The amount of increased battery level in the default charging was one third of charge100, while the FPS was maintained at 60. This means that the charging efficiency can be improved up to 3 times without FPS loss. On the other hand, with the thermal workload (Fig. 7(c) and (d)), the battery was charged slightly less than the case in the normal workload because the device consumed a large amount of power, decreasing the residual power for battery charging. The FPS trace indicates that although the default system reduced the charging power considerably, the FPS loss is similar to that for charge100 up to 500 seconds. Therefore, we consider that the charging efficiency can be improved even in a thermal workload.

For a detailed analysis, we experimented with the schemes with further applications. Table 2 shows the summary of the results. The charging power of the default charging was kept constant among all loads. With charge100, there were situations where the charging rate was kept to the maximum without performance degradation. In the case of Aquarium2, which is a thermal workload, the performance was degraded, but in other cases, there was no performance difference even if the charging speed was doubled or more. The default scheme reduced the charging rate even when there was no performance degradation. In summary, the default scheme used a conservative method to prevent the performance degradation caused by high power charging. We thought these shortcomings could be overcome by adjusting the charging rate appropriately.

F. Summary

With the preliminary experiments, we observed the charging characteristics of a mobile device. First, we found that the temperature of the PMIC component in the device rises during charging, and the heat generated by the PMIC affects the AP. As a result, charging raises the temperature of the AP. Second, the heat generated by charging affects the performance of the device in some cases, when a heavy workload is running. Third, we found that the default charging policy implemented in commercial devices often operates inefficiently, especially when the device is in use. In many cases, there were opportunities for high-rate charging, but the default scheme did not work that way.

III. ALWAYS-ON QUICK CHARGING

The existing charging schemes severely reduce the charging power to a certain level, when the device is used while charging, regardless of the workload and temperature. The charging speed remains slow even when the heat generated by the charging does

TABLE II
PERFORMANCE OF DEFAULT CHARGING

Name	Power (W)	Charge100		Default	
		FPS	Charging Power (W)	FPS	Charging Power (W)
Aquarium1	2.95	59.5	9.87	59.6	4.02
Aquarium2	4.3	29.6	9.17	33.7	4
YouTube	1.9	55.2	10.2	56.1	4.01
Asphalt	2.4	30.6	10.4	30.5	4.02

not affect the performance. We now propose a scheme, called Always-on Quick Charging (AQC, hereafter), which controls the charging rate in order to provide a fast charging speed even with device use. The key idea is to maintain the maximum charging power adaptively without performance degradation caused by heat generated with charging.

One of the key challenges is to accurately predict the performance degradation due to the heat, and to figure out the effect of regulating the charging rate on a mobile device. We focused on thermal throttling which degrades the performance of the device resulting from the increase in temperature. In general, thermal throttling works based on two parameters: the thermal threshold, $T_{threshold}$ and the clearing set point, T_{clr}. $T_{threshold}$ is the temperature point at which the throttling should be activated, whereas T_{clr} is the target temperature at which the temperature is to be lowered. Thermal throttling occurs when the temperature reaches $T_{threshold}$ and is released when the temperature reaches T_{clr}. When the AP temperature reaches $T_{threshold}$ and performance degradation is expected, the proposed scheme lowers the charging power drastically to decrease the temperature immediately. When the AP temperature drops below T_{clr}, the charging power is raised slowly until the AP's temperature becomes higher than T_{clr}. The charging power is maintained while the AP temperature stays between $T_{threshold}$ and T_{clr}. That is, the charging power is increased up to the point where the thermal violation does not occur.

To control the charging power effectively, the charging power is divided by L steps. A single step of charging power α is $ChargingPower_{maximum}/L$ where $ChargingPower_{maximum}$ is the maximum charging power of the device. When the temperature is lower than T_{clr}, the proposed scheme increases the charging level step by step. In the case of a thermal violation, we should determine how many levels to drop to lower the temperature under T_{clr}. We obtained T_{α}, the temperature difference for a single charging step, and then calculated the number of charging levels to drop, D, as $(T_{threshold} - T_{clr})/T_{\alpha}$. As the temperature change takes some time, the proposed scheme waits $Time_{wait}$ after controlling the charging level and then adjusts the charging power again depending on the temperature. Note that $T_{threshold}$, T_{clr}, L, D, and $Time_{wait}$ should be empirically determined for the target device. The detailed implementation of the temperature monitoring and charging control is described in Section 5.1.

Fig. 8 illustrates the working mechanism of the AQC scheme. When the charging starts, the charging power is kept at the maximum. As the charge progresses, the temperature is checked

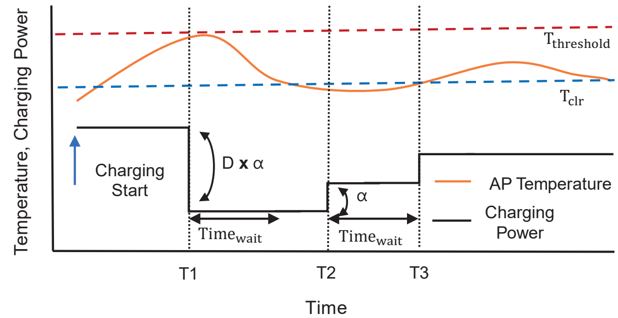


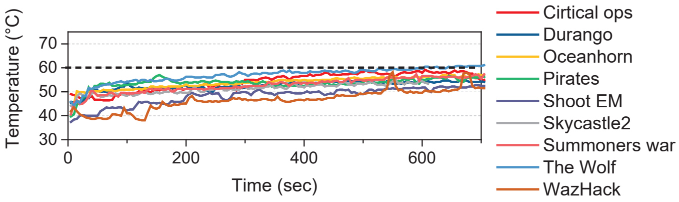
Fig. 8: Concept of Always-on Quick Charging.

to see if it has reached $T_{threshold}$. If it has, the charging rate is reduced by D (T1 in the figure). After waiting $Time_{wait}$, if the temperature is lower than T_{clr}, the charging power is increased by α (T2). Eventually, if the temperature stays between $T_{threshold}$ and T_{clr}, the charging rate is maintained (after T3). This way, AQC finds the charging power without a thermal violation.

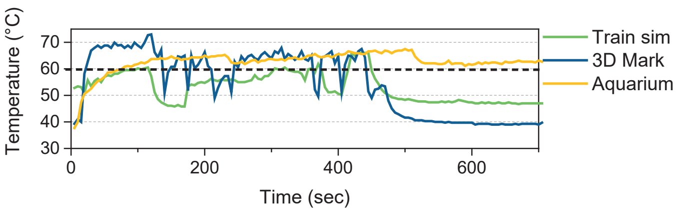
Smartphone employs many sophisticated software schemes optimized for target hardware. Any functional add-up or modification should not interfere with the baseline system, and the resulting system should be working effectively with maximum transparency. Developing a thermal model which predicts the internal temperature of device is practically very difficult. For a pragmatic solution, we exploited an intuitive and experimental approach, rather than a control theoretic thermal management, to cope with the generality issue for target devices. Although our AQC mechanism is simple and intuitive, the scheme indeed enhances the charging efficiency of device without performance degradation (see Section 5). This is a meaningful approach in terms of pragmatic aspect of device use. Compared to previous studies [7-10], our scheme is the first to handle the issue on maximizing charging efficiency in mobile device.

IV. IMPLEMENTATION

Always-on Quick Charging is implemented on Google's Pixel 2XL smartphone running Android 8.1.0 (Oreo). We modified the kernel to regulate the charging power and monitor the temperature. Because AQC requires modifications of only the kernel, the scheme can be applied to other mobile devices that support quick charging. In Pixel 2XL, the charging current can be controlled by modifying the PMIC driver. We adjusted the charging rate by changing the POWER_SUPPLY_PROP_CONSTANT_CHARG_CURREN T_MAX value in /driver/power/*smbcharger.c which determines the charging current. Note that controlling the charging current is enabled if the charger and the device support fast charging. To monitor the device temperature, we collected data from /sys/class/thermal/. We obtained various information, such as the list of temperature sensors and the temperature data of other modules, including the AP, battery, and cameras. Finally, the actual charging control is conducted by an application. We developed an application that monitors the AP temperature in the background and controls the charging using the kernel interface.

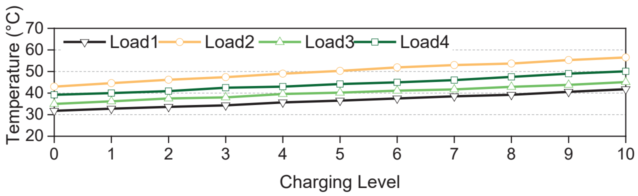


(a) Normal workloads

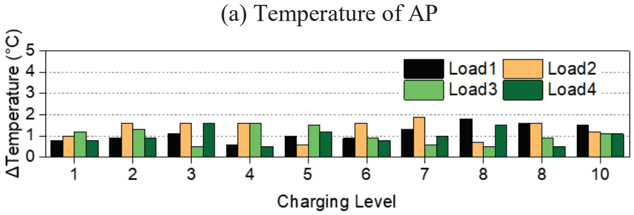


(b) Thermal workloads

Fig. 9: AP temperature in Pixel 2XL.



(a) Temperature of AP



(b) Temperature difference from previous level

Fig. 10: Temperature change with charging level in Pixel 2XL.

V. Evaluation

We evaluated AQC with the Pixel 2XL. First, the AQC parameters were determined experimentally. We then validated the improvement in the charging efficiency in a controlled environment and with real applications.

A. Parameters

The proposed charging control requires several parameters which are specifically determined for the target device. First, the thermal violation point $T_{\text{threshold}}$ which triggers thermal throttling of the default system should be known for the device. We experimentally observed the temperature by running a number of workloads. Fig. 9(a) and (b) show the temperature without and with a thermal violation, respectively. In Fig. 9(a), the temperature of the AP did not exceed 60 °C, whereas, in Fig. 9(b), the temperature dropped sharply due to thermal throttling around 60 °C. We set this point as $T_{\text{threshold}}$. We also observed that after thermal throttling, the temperature began to increase again around 55°C, which indicates the thermal throttling has ended. Thus, T_{clr} was determined as 55°C.

Next, the number of charging levels L should be fixed. If L is large, the charging speed will be slowed down because the charging level is increased step by step. If L is small, the charging current cannot be fine-tuned; thus, the charging efficiency is degraded. For our target device Pixel 2XL, we determined L as 10 heuristically. As L was set to 10, α became 1.2 W. The temperature change per single charging step, T_{α}, should also be determined to calculate the number of charging steps to decrease, i.e., D. We observed the temperature by increasing the charging level gradually. During the experiment, we found that the temperature saturated 1 min after changing the charging level. Therefore, we set the $Time_{wait}$ as 1 min. Fig. 10. (a) and (b) show the temperature of AP and its difference, respectively, according to the charging level while running four different static workloads. Note that we used various workloads to backup the generality of the approach. As shown in the figure, T_{α} is 1°C, on average, independent of the workload. D is calculated as $(T_{threshold} - T_{clr})/T_{\alpha}$, and set as 5. We used these parameters in the rest of the experiments.

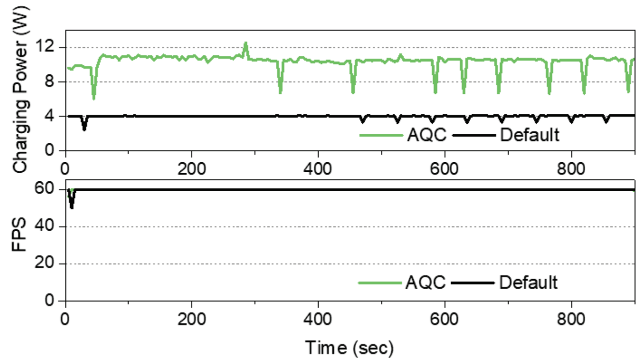


Fig. 11: Aquarium 1 (normal workload) in Pixel 2XL.

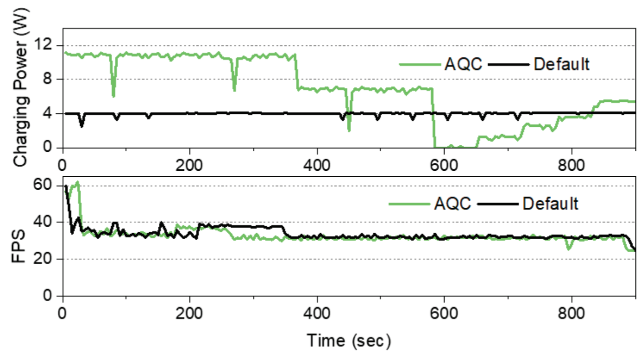


Fig. 12: Aquarium 2 (thermal workload) in Pixel 2XL.

B. Controlled Environments

First, we evaluated the system in a controlled environment. We measured the charging current and the temperature while running the workloads used in Section 2.4. The performance of the proposed scheme is then compared with the default scheme, i.e., the baseline.

Fig. 11 shows the results for the normal workload. As we observed in the motivation experiments, the baseline system charged the device with 4.3 W. The charging current of the proposed scheme was, however, 9.7 W, which is 2.4 times higher than the baseline. In addition, the FPS did not suffer any degradation with the proposed scheme. For the thermal workload, Fig. 12 shows that the charging current decreased

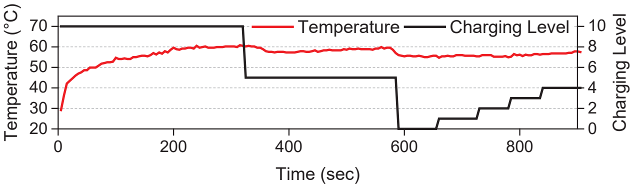


Fig. 13: AQC performance with thermal workload.

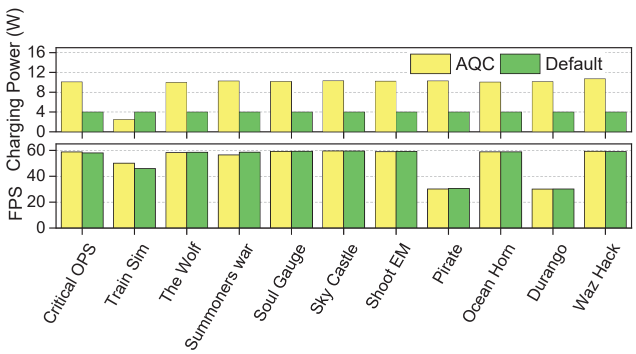


Fig. 14: Performance with real applications.

lower than 8 W around 300 s. At around 600 s, the temperature reached the thermal violation point, and the proposed scheme decreased the charging current again. As the temperature kept rising, the charging current remained at a minimum value. The baseline maintained the charging current at about 4 W, indicating that the charging efficiency was better than that of the proposed system. However, because the temperature of the proposed scheme was lower than the baseline, the proposed scheme will decrease the temperature faster when the workload is light.

To further analyze the detailed operation of the AQC, we observed the charging level and temperature specifically with thermal workload. Fig. 13 shows that the charging level was the maximum at the beginning of the experiment because the temperature was much lower than T_{clr}. Around 300 s, the temperature reached a thermal violation point, and the charging level dropped to 5, which results in decreasing temperature. As the temperature stayed between $T_{violation}$ and T_{clr}, the charging level remained at 5. When the temperature rose to $T_{violation}$ at around 600 s, the proposed scheme dropped the charging level again. After waiting 60 s, the charging level increased by one because the temperature was under T_{clr}. Finally, the proposed scheme determined the optimal charging level was 4 .

With this experiment, we confirmed that AQC is efficient for normal and thermal workloads, and the charging power and the FPS of the proposed scheme were higher than the baseline.

C. *Real Applications*

We then validated the effectiveness of the AQC with real applications whose workload changed dynamically. We measured the power consumption and the FPS while running the applications in Fig. 9, for 15 min. In Fig. 14, the charging power and the FPS of the proposed system are compared with those of the baseline system. For all applications except for Train Sim, AQC improved the charging efficiency by 2.4 times compared to the default charging system while preserving the FPS with only a 3% loss. In the case of Train Sim, AQC lowered the charging current to guarantee the FPS because a large amount of heat was generated due to the high workload. Consequently, although the FPS of both systems was under 60, the FPS of the proposed system was higher than that of the baseline. Note that

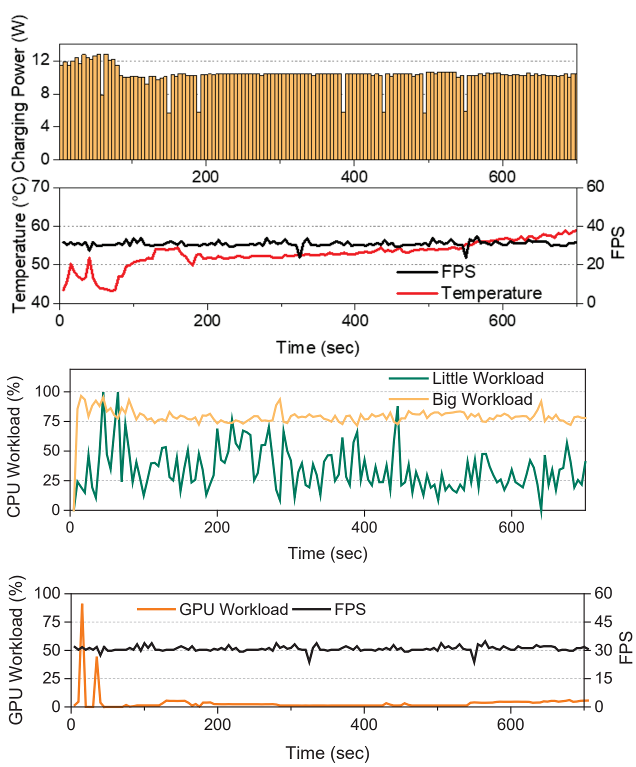


Fig. 15: Assassin's Creed Pirate (moderate workload).

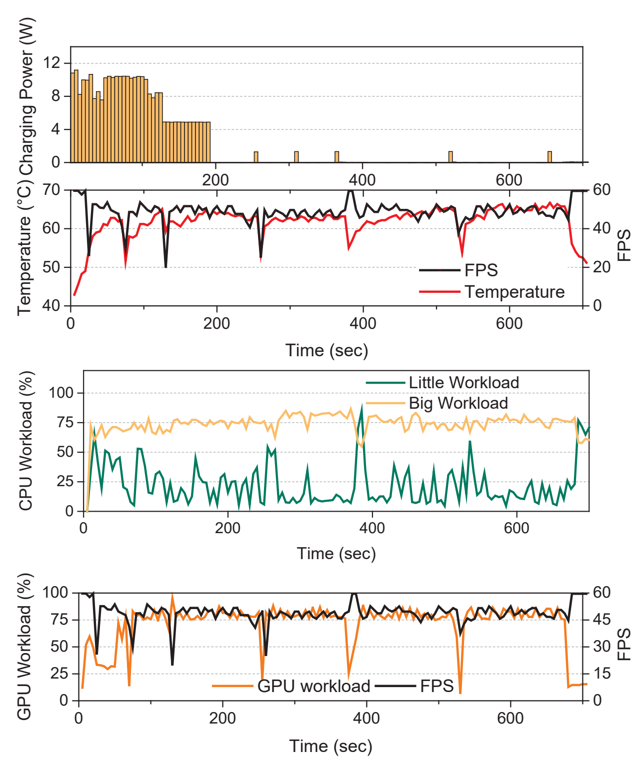


Fig. 16: Train Sim (high workload).

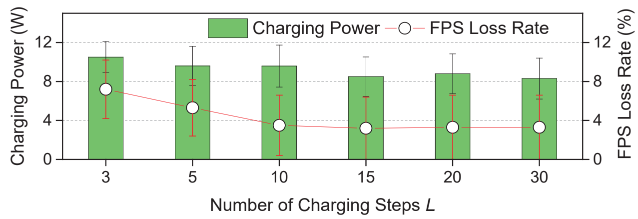


Fig. 17: Charging power and FPS loss vs. L.

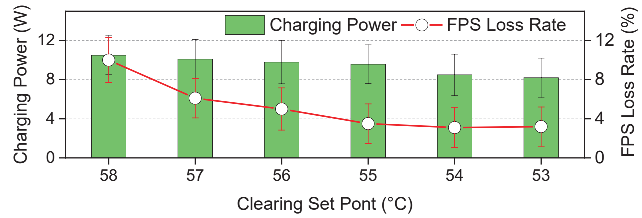


Fig. 18: Charging power and FPS loss vs. clearing set points.

the user's quality of experience for Pirate and Durango was not degraded because their maximum FPS was 30.

We further analyzed the device behavior to validate the accurate operation of the proposed scheme. The analysis was conducted with Pirate and Train Sim whose workloads were moderate and high, respectively. First, we investigated device behaviors while running the Pirate game for about 15 min. Fig. 15 shows the charging current, temperature, FPS, CPU workload, and GPU workload. The CPU workload was calculated by the same method of GPU. The charging current remained high, i.e., two times more than the baseline, because not much heat was generated due to the moderate workload. The temperature graph verifies that the device's temperature did not exceed the thermal violation point. Accordingly, the FPS was maintained at 30. The CPU workload graph shows that AQC actively used big and LITTLE cores, indicating that thermal throttling was not performed. Similarly, we analyzed the high workload case of Train Sim. Fig. 16 shows that the charging current was reduced step by step to lower the temperature. However, due to the high workload, the temperature was not reduced considerably. The CPU workload graph confirms that thermal throttling was not conducted, while the GPU graph indicates that the FPS is degraded due to GPU throttling. As Train Sim is a game application, GPU usage is dominant over CPU usage, causing only GPU throttling. Despite these efforts, the overall FPS was lower than 60. This is not avoidable because the workload is extremely high and generates a large amount of heat.

In summary, the experiment results verified that Always-on Quick Charging outperforms the original charging system in terms of charging efficiency and FPS. By balancing the charging current and temperature increment, AQC maximized the charging efficiency while preserving the FPS. Although the user experience of a high-workload application was degraded, the proposed system still provided a high FPS, compared to the original system, by aggressively lowering the charging current.

D. Charging Steps and Clearing Set Point

The proposed charging control has two critical parameters: the number of charging steps L and the clearing set point T_{clr}. In our evaluation, we determined the number of charging steps heuristically. As other parameters such as α, T_{α}, and D are calculated by L, it severely affects the performance of the proposed scheme. We observed the charging efficiency and FPS loss according to L. The experiment scenario was the same as in Fig. 13. Fig. 17 shows that the charging power was large when L was small. However, because the charging power for one step is large, the temperature increased sharply although the proposed system raised only a single step. Therefore, thermal

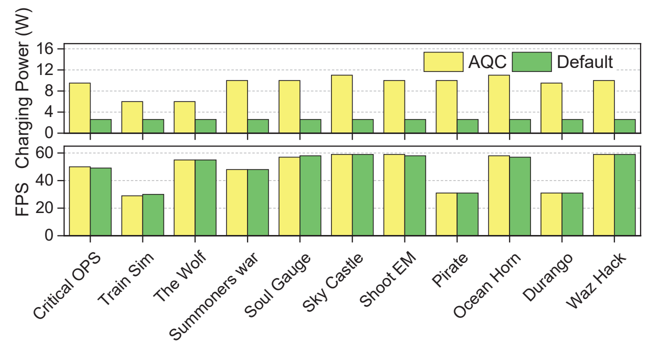


Fig. 19: LG V20 with real applications.

throttling occurred more often, degrading the FPS. However, when the charging level was more finely divided, the FPS was preserved, but the average charging power decreased because the charging level increased too slowly. Following this principle, ten steps of charging were chosen for the target device Pixel 2XL.

In the case of T_{clr}, we set the value at 55°C, the same as for the baseline system. Because the existing charging system aims to preserve the user experience rather than the charging efficiency, we consider 55°C a conservative threshold. We specifically measured the charging power and the FPS according to T_{clr}. Fig. 18 indicates that as T_{clr} increases, the charging speed becomes faster because the charging level is raised up until the temperature reaches T_{clr}. However, thermal throttling occurred more frequently, degrading the FPS. As the ultimate goal of the proposed system is maximizing the charging efficiency without FPS loss, 55°C is chosen as appropriate for Pixel 2XL. Note that these results would change depending on the target device.

VI. DISSCUSSION

Robustness The device manufacturers carefully design and implement sophisticated software schemes such as DVFS, touch boosting, and thermal management to preserve the user experience. Instead of replacing the default thermal management scheme, the proposed system works on it to guarantee the user experience because the robustness of the default system has been proved by the manufacturer. By augmenting the proposed system on the default thermal management, we can maximize the charging efficiency while minimizing the device temperature and degradation of the user experience. Considering the long-term effects on the device, our scheme operates within the maximum charging current of the baseline system. Therefore, over-charging is fundamentally prevented, assuring no adverse effect on the underlying battery system.

Generality To prove that the effectiveness of the proposed system is not limited to the Pixel 2XL, we also implemented AQC for the LG V20 smartphone and conducted the same experiment as in Section 5.3. The experimental result shown in Fig. 19 verifies that the proposed system can improve the charging efficiency for other devices.

Most of smartphones adopts the CC-CV (constant current, constant voltage) scheme for battery charging. The scheme charges device with constant current in the first phase (CC mode), followed by charging with constant voltage in the second phase (CV mode). In the CV mode, the charging current decreases gradually as the voltage of battery increases. This means that devices controls the charging current during the CV mode. Our scheme which controls the charging current in CC mode can therefore be applied to general devices. We found that controlling the charging current is indeed possible, through kernel modifications, in Galaxy S7, S8, Pixel2, Pixel XL, Pixel 2XL, Nexus 5X, Nexus 6P, and LG V20, which are the recent smartphones in the market. Although we were not able to check the functionality for all the devices in the market, we believe that the functionality is generally available in typical smartphones.

Portability Several issues should be considered to implement the proposed system: (1) understanding of the device characteristics, such as the maximum available charging current, (2) modification of the kernel for monitoring the temperature and controlling the charging power, and (3) determining the parameters for the charging mechanism which are specific to the target device. For example, certain devices prohibit charging power control due to security issues. Although the porting issue exists depending on the target devices, we believe that this issue will be overcome easily by manufacturers.

VII. Related Work

Since charging efficiency is an important factor for mobile users, there have been several attempts to improve charging efficiency in various aspects. He et al. [6] maximized the charged capacity within the available charging time determined by the user. Zaghib et al. [7] developed fast charging using a new material which charges 800 mAh of battery within 4 min. Chen et al. [8] adopted a grey prediction technique to improve the charging speed of the Constant Current (CV) mode in conventional charging. He et al. [9] focused on battery health and tried to minimize the negative effect of conventional Constant Current - Constant Voltage (CC-CV) charging on the battery life by adjusting the duration of the CC-CV charging mode based on usage patterns. None of the previous works considered the impact of charging on the performance while using the device. To the best of our knowledge, we are the first to propose a charging mechanism to maximize the charging efficiency while using a device.

Other works focused on managing the temperature of the mobile device. Bhat et al. [10] analyzed the power-temperature trajectory and predicted heat generation according to the workload. Singla et al. [11] predicted the leakage power due to high temperatures and managed the thermal issue based on the prediction result. Lee et al. [12] tried to predict the temperature of each hardware component based on internal temperature sensors and proposed a thermal management scheme. Paterna and Rosing [13] formulated a model to describe how the temperature of the battery in a mobile device affects its AP and utilize it for thermal management. Sahin et al. [14] defined a new metric for the Quality of Service (QoS) and throttled the device's performance before degrading the QoS. Kim et al. [15] managed the thermal issue depending on the user's preference between performance and power. Park and Chen [16] considered CPU variation during the manufacturing process and reduced heat generation by finding the minimum voltage for each CPU. Dai et al. [17] tried active cooling for mobile devices by using additional hardware, such as Peltier material. However, no previous works attempted to manage thermal issues by changing the charging current.

VIII. Conclusion and Future Work

We observed that using a mobile device while charging affects the temperature of the device's AP. Furthermore, we showed that such heat affects the thermal violation and thus, diminishes the performance of the device. Given the circumstances, our Always-on Quick Charging scheme successfully monitors the temperature and adjusts the charging current adequately to provide accelerated charging capability. The proposed scheme was shown to perform better than the existing schemes in all cases we tested.

The proposed scheme can be improved further in several ways. The charging current could be adjusted adaptively by predicting when the user demands high performance. This will require personalizing a charging policy. Another improvement is to simplify the collection of the required parameters. With an automatic or inferred method of collecting the parameters, a more effective solution can be devised.

Acknowledgement

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Science and ICT (No. NRF-2017M3C4A7083677).

References

[1] Qualcomm Quick charge, https://www.qualcomm.com/solutions/mobile-computing/features/quick-charge, [Online; accessed January 2019]

[2] Fast Adaptive Charging, http://www.samsung.com/global/galaxy/what-is/adaptive-fast-charging/, [Online; accessed January 2019]

[3] Power Delivery, http://www.usb.org/developers/powerdelivery/, [Online; accessed January 2019]

[4] Q. Xie, J. Kim, Y. Wang, D. Shin, N. Chang, and M. Pedram, : "Dynamic thermal management in mobile devices considering the thermal coupling between battery and application processor," in Computer-Aided Design (ICCAD), 2013 IEEE/ACM International Conference on, 2013, pp. 242-247: IEEE.

[5] R. Muralidhar et al., "Experiences with power management enabling on the Intel Medfield phone," in Proc. of Linux Symposium, 2012, pp. 35-46.

[6] L. He, Y.-C. Tung, and K. G. Shin, "icharge: User-interactive charging of mobile devices," in Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services, 2017, pp. 413-426: ACM.

[7] K. Zaghib et al., "Safe and fast-charging Li-ion battery with long shelf life for power applications," Journal of Power Sources, vol. 196, no. 8, pp. 3949-3954, 2011.

[8] L.-R. Chen, R. C. Hsu, and C.-S. Liu, "A design of a grey-predicted Li-ion battery charge system," IEEE Transactions on Industrial Electronics, vol. 55, no. 10, pp. 3692-3701, 2008.

[9] L. He, E. Kim, and K. G. Shin, "*-Aware charging of lithium-ion battery cells," in Proceedings of the 7th International Conference on Cyber-Physical Systems, 2016, p. 26: IEEE Press.

[10] G. Bhat, S. Gumussoy, and U. Y. Ogras, "Power-temperature stability and safety analysis for multiprocessor systems," ACM Transactions on Embedded Computing Systems (TECS), vol. 16, no. 5s, p. 145, 2017.

[11] G. Singla, G. Kaur, A. K. Unver, and U. Y. Ogras, "Predictive dynamic thermal and power management for heterogeneous mobile platforms," in Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition, 2015, pp. 960-965: EDA Consortium.

[12] J. S. Lee, K. Skadron, and S. W. Chung, "Predictive Temperature-Aware DVFS," IEEE Trans. Computers, vol. 59, no. 1, pp. 127-133, 2010.

[13] F. Paterna and T. Š. Rosing, "Modeling and mitigation of extra-SoC thermal coupling effects and heat transfer variations in mobile devices," in Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, 2015, pp. 831-838: IEEE Press.

[14] O. Sahin, P. T. Varghese, and A. K. Coskun, "Just enough is more: Achieving sustainable performance in mobile devices under thermal limitations," in Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, 2015, pp. 839-846: IEEE Press.

[15] J. M. Kim, Y. G. Kim, and S. W. Chung, "Stabilizing CPU frequency and voltage for temperature-aware DVFS in mobile devices," IEEE Transactions on Computers, vol. 64, no. 1, pp. 286-292, 2015.

[16] J. Park and H. Cha, "T-DVS: Temperature-aware DVS based on temperature inversion phenomenon," in Proceedings of the 2016 International Symposium on Low Power Electronics and Design, 2016, pp. 248-253: ACM.

[17] Y. Dai, T. Li, B. Liu, M. Song, and H. Chen, "Exploiting Dynamic Thermal Energy Harvesting for Reusing in Smartphone with Mobile Applications," in Proceedings of the Twenty-Third International Conference on Architectural Support for Programming Languages and Operating Systems, 2018, pp. 243-256: ACM.