Minimizing Capacity Degradation of Heterogeneous Batteries in a Mobile Embedded System

Jaeheon Kwak and Jinkyu Lee, *Member, IEEE*

***Abstract*—As different batteries exhibit their own advantages and disadvantages, a single-type battery system for a mobile embedded system (such as smartphones and tablet PC) cannot overcome the inherent limitations of its target battery. For example, although the lithium cobalt oxide (LCO) battery is the most popular battery for mobile embedded systems due to its high energy density, mobile embedded systems equipped with LCO batteries suffer from rapid capacity degradation. In this letter, we target heterogeneous batteries in a mobile embedded system and propose a method for minimizing the capacity degradation of the batteries. To this end, we first analyze factors that affect capacity degradation, and choose dominant factors that are controllable in a mobile embedded system. We then build a battery degradation model considering the factors, and finally develop a battery scheduling algorithm that minimizes capacity degradation using the degradation model. Our evaluation with real experiments and simulations demonstrates the effectiveness of the proposed algorithm in minimizing capacity degradation.**

***Index Terms*—Battery degradation model, battery scheduling algorithm, capacity degradation, heterogeneous batteries, mobile embedded systems.**

I. Introduction

In MOBILE embedded systems, such as smartphones and tablet PC, battery performance is one of the most critical issues due to their "mobile," nature, and battery technology has evolved to support increasing power/life-cycle demand for these systems [1]–[4]. Meanwhile, individual batteries have their own characteristics determined by their materials, as shown in Fig. 1 (originally from [5]). For example, the lithium iron phosphate (LFP) battery exhibits a long life cycle and low energy density, whereas the lithium cobalt oxide (LCO) battery has high energy density; also, lithium nickel manganese cobalt oxide (NMC) battery has balanced characteristics [5]. Although the LCO battery is the most popular battery for mobile embedded systems due to its high energy density, mobile embedded systems with the LCO battery suffer from rapid capacity degradation via aging. One may think that other batteries, such as LFP and NMC remedy the disadvantage of LCO, but replacing LCO with LFP or NMC leads to other problems such as low energy density because of their own disadvantages. This necessitates utilizing multiple types of batteries in a mobile embedded system.

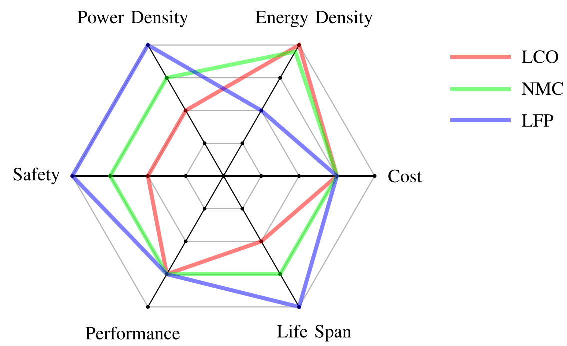


Fig. 1. Characteristics of different batteries [5].

In this letter, we target a heterogeneous battery system that consists of different types of batteries in a mobile embedded system and propose how to minimize capacity degradation of the battery system. Considering the high demand of energy density in mobile devices, we target LCO and NMC battery in this letter. Badam *et al.* [6] proposed a heterogeneous battery system for a mobile embedded system, including a design of hardware architecture and system-level application programming interface. Based on such hardware/software support for a heterogeneous battery system, we focus on a battery scheduling algorithm to achieve our goal of minimizing capacity degradation of the system. To this end, we first analyze factors that affect capacity degradation and choose dominant factors that are controllable in a mobile embedded system. We then build a battery degradation model, utilizing the analysis of the dominant factors. Finally, we develop a battery scheduling algorithm that minimizes capacity degradation using the degradation model. Note that since there has been no study for managing capacity degradation or aging of heterogeneous batteries in a mobile embedded system, this letter is differentiated from existing studies for heterogeneous battery systems, e.g., a study [7] that suggested balanced charge/discharge control strategies for interconnected heterogeneous battery systems in order to maintain balanced state-of-charge (SoC) of batteries.

Our evaluation with real experiments and simulations demonstrates that a heterogeneous battery system with our

Manuscript received February 22, 2019; revised May 7, 2019; accepted June 22, 2019. Date of publication July 15, 2019; date of current version February 26, 2020. This work was supported in part by the National Research Foundation of Korea (NRF) funded by the Korea Government under Grant 2019R1A2B5B02001794, and in part by the Program for Fostering Next-Generation Researchers in Engineering of NRF funded by the Ministry of Science, Information, and Communication Technology, and Future Planning under Grant 2017H1D8A2031628. This manuscript was recommended for publication by H. Tomiyama. *(Corresponding author: Jinkyu Lee.)*

The authors are with the Department of Computer Science and Engineering, Sungkyunkwan University, Suwon 16419, South Korea (e-mail: jaehunny@skku.edu; jinkyu.lee@skku.edu).

Digital Object Identifier 10.1109/LES.2019.2929069

algorithm decreases capacity degradation by up to 16.95% and 26.22% compared to the corresponding homogeneous battery system and the corresponding heterogeneous battery system with existing algorithms, respectively.

II. Problem Statement

Our goal is to minimize the capacity degradation of heterogeneous batteries in a mobile embedded system. Because the energy density of a battery system for a mobile embedded system depends on a given set of different types of batteries, we focus on the scheduling issue (i.e., how much we use given individual batteries). There are several strategies for the scheduling issue of multiple batteries, but most of these not only target different battery systems, such as energy grid systems [8] or electric vehicles [9] but also consider a homogeneous battery system [1]. Our target system, heterogeneous batteries in a mobile embedded system, is different from those systems for the following reasons; consequently, it is not proper to use existing strategies. First, in mobile embedded systems, it is impossible to use hardware used in existing studies (e.g., an energy buffer such as an ultracapacitor) and to control many important factors that affect battery status (e.g., active cooling/heating system to control temperature). Second, the portability of a mobile embedded system causes various usage patterns [10], which limits the opportunity to improve battery performance using predictable usage patterns. These two circumstances also result in a limitation of control knobs for a mobile embedded system's batteries. That is, in many cases, one may extend battery life by controlling the mean of SoC, depth-of-discharge (DoD), temperature, and/or C-rate, but only the manageable parameter for a mobile embedded system's batteries is the C-rate. Considering such a practical issue, this letter only controls C-rate of individual batteries in a mobile embedded system's heterogeneous batteries.

III. Minimizing Capacity Degradation

We now propose a method to minimize the capacity degradation of heterogeneous batteries in a mobile embedded system, which consists of the following three steps:

(S1) identifying important factors that determine battery degradation;
(S2) building a battery degradation model using the important factors;
(S3) developing an algorithm to minimize capacity degradation.

As for S1, it is known that the major factors that degrade battery capacity are the effect of solid electrolyte interface (SEI) layer growth (Q_{SEI}) and that of active material loss (Q_{AM}) [11]. The influence of Q_{SEI} and Q_{AM} changes as time progresses, as follows. In the early period, Q_{SEI} is the dominant factor, but the effect of Q_{AM} becomes dominant over time [12]; finally, the influence of Q_{AM} becomes much greater than that of Q_{SEI} as Q_{SEI} stops growing [12]. Therefore, with a long-term perspective, we approximate the total loss Q_{loss} as $k_Q + Q_{\mathrm{AM}}$, regarding Q_{SEI} as a constant k_Q. Jin *et al.* [11] modeled Q_{AM} via electrochemical background as follows:

$$Q_{\mathrm{AM}} = k_{\mathrm{AM}} * \exp\left(\frac{-E_{\mathrm{AM}}}{R * T}\right) * \mathrm{AH}. \tag{1}$$

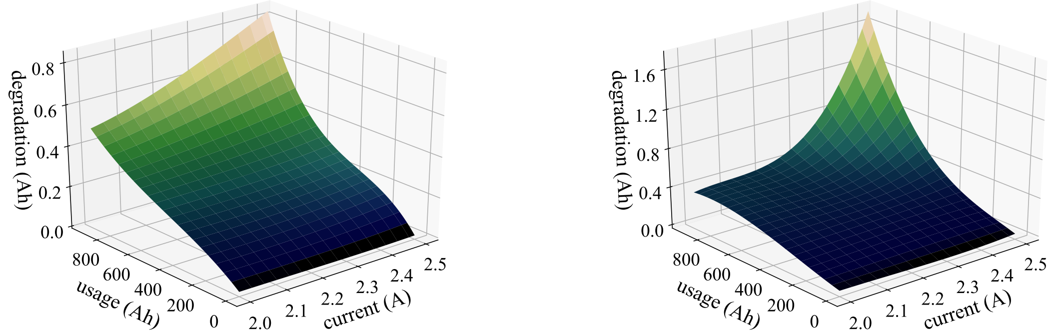
To simplify the above equation, we can get tuned values of: 1) the activation energy of active material loss E_{AM}; 2) the constant related to the electrochemical properties inside the battery k_{AM} [13], [14]; and 3) the gas constant R. Also, we can fix the change of temperature T, which actually changes but is difficult to control in a mobile embedded system. Then, the above model can be approximated as $Q_{\mathrm{AM}} \approx k * AH$ by regarding the tuned values as a constant k. Note that AH means SoC weighted usage and is defined as $AH = \int \mathrm{SoC} * I \, dt$, where SoC, I, and t are state of charge, C-rate, and time, respectively. Because C-rate varies with time, we can define C-rate as a function of time $I = f(t)$ and SoC as $\mathrm{SoC} = \int_{t_1}^{t_2} [f(t)/\mathrm{Cap}]dt + \mathrm{SoC}_0$, where SoC_0 is the initial state of SoC, and Cap is the capacity of the battery. Let $F(t)$ denote a function of the integration of $f(t)$. When time progresses from t_1 to t_2, we obtain the following equation:

$$Q_{\mathrm{AM}} \approx k * \frac{\int_{t_1}^{t_2} F(t)f(t)dt}{\mathrm{Cap}} + \int_{t_1}^{t_2} f(t)dt * \mathrm{SoC}_0. \tag{2}$$

Equation (2) implies three points. First, when temperature and the other tuned parameters shown in (1) are fixed, Q_{AM} is only determined by the amount of SoC change, which is the amount of battery usage. Second, a small change of battery current does not affect degradation of the battery very much in the long-term; however, on the contrary, it decreases the cycle life due to increased heat generation. Third, the C-rate is a controllable, important factor, when batteries have the same usage. Because temperature determines degradation speed (although it is difficult to control on a mobile embedded system), C-rate is a controllable factor and has a significant impact on heat-generation. Based on the derivation of (2) and its observation, we establish a strategy using fixed current for individual batteries and select usage and C-rate as important factors for controlling degradation speed.

For S2, we build a degradation model to understand the change in degradation related to some important factors. Although C-rate and usage have been selected as important factors that affect Q_{AM} via SoC and temperature changes, it is time-intractable to conduct real experiments of battery capacity degradation for every possible pair of C-rate and usage. Therefore, we develop our degradation model by experimenting with real batteries in several pairs and generating data for other pairs using interpolation. We apply curve fitting with a polynomial function, which is commonly used to model battery characteristics [15]. We modeled capacity degradation with the third-order polynomial function by computing least-square solution. Fig. 2 is the result of our polynomial battery model, which shows 1.51% and 0.99% average error for the 3.0-Ah LCO and 2.6-Ah NMC battery, respectively.

The remaining step (i.e., S3) is to develop an algorithm that minimizes capacity degradation. To minimize capacity degradation of the target battery system, we should find the optimal C-rate for individual battery types and for given battery usage. We solve this problem by searching the optimal



(a) (b)

Fig. 2. Different degradation characteristics of (a) LCO and (b) NMC.

Algorithm 1 Capacity Degradation Minimization

1: *numT*, *numN* = the number of battery types, batteries
2: *totalU*, *totalC* = demand of usage, C-rate
3: *listT* = a list of the number of batteries in each battery type
4: *rangeT* = a list of the range of C-rate of each battery type
5: Construct *setC*, a set of combinations each of which is a list of assigned C-rate of each battery type, with four constraints
6: *minDegradation* = ∞
7: **for** *i* in *setC* **do**
8:     *tempDegradation* = 0
9:     **for** *n* in size(*setC*) **do**
10:         *oneDegradation* = batteryModel(*n*, *i*[*n*])
11:         *tempDegradation* + = *oneDegradation* * *listT*[*n*]
12:     **if** *tempDegradation* < *minDegradation* **then**
13:         *minDegradation* = *tempDegradation*
14:         *solutionListC* = *i*
15: **return** *solutionListC*

C-rate with our polynomial battery model developed in S2, which is described in Algorithm 1. Let the number of battery types, number of batteries, usage demand, and C-rate demand be denoted as *numT*, *numN*, *totalU*, and *totalC*, respectively (lines 1 and 2). Also, *listT* and *rangeT* store the number of every battery type and the range of C-rate of every battery type (e.g., *listT* = [1, 1], *rangeT* = 2.0 A to 2.5 A), respectively (lines 3 and 4). The algorithm then constructs *setC*, a set of combinations. Each combination is a list of the assigned C-rate of every battery type, such that the following four conditions should be satisfied (line 5).

1) The C-rate of each battery type is within its range in *rangeT*.
2) The C-rate of the same battery type should be the same.
3) The sum of C-rate should be equal to *totalC*.
4) Finally, the step size is X, meaning that every C-rate is assigned by X multiplied by a natural number; this makes the algorithm computationally tractable.

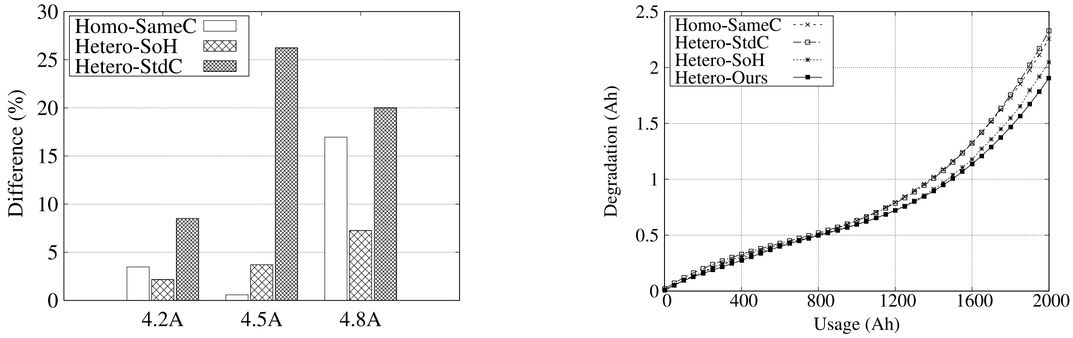
Then, the algorithm searches the optimal C-rate allocation to minimize capacity degradation by checking all combinations in *setC* (lines 6–15). For each combination, lines 9–11 calculate the total degradation of each combination using the battery model developed in S2. Then, lines 12–14 compare the total degradation of each combination with the minimum degradation obtained so far to keep the optimal C-rate list. Finally, the algorithm returns the solution C-rate list (line 15).

One may wonder whether the time complexity of the algorithm can be high, which is $O((rangeT_{\max}/X)^{numT})$ in the worst case where $rangeT_{\max}$ is the largest *rangeT* among all battery types, but this does not cause a problem for two reasons. First, *numT* and *numN* are usually small in practice. Second, we can control the time complexity by adjusting the step size of X according to *numT* and *numN*, i.e., use larger X for larger *numT* and *rangeT*.

IV. Evaluation

To evaluate our approach, we use the following experiments and simulations. We develop our battery model based on the experimental result of the 18 650 size battery using the NEWARE BTS-4000 battery tester. Samsung ICR18650-30A 3.0-Ah LCO battery and LG INR18650 B4 2.6-Ah NMC battery are used for experiments. We experiment with batteries with currents of 2.0 A, 2.25 A, and 2.5 A and interpolate untested current values by the third-order polynomial equation, as described in the previous section. Among the several factors that affect battery degradation mentioned in the previous section, we fix the values of temperature, DoD, and mean SoC to 25 °C, 100%, and 50%, respectively, and C-rate is determined by Algorithm 1. Targeting the two batteries of LCO and NMC, we develop a polynomial battery model and generate degradation curves for LCO and NMC as shown in Fig. 2.

We compare our approach `Hetero-Ours`, with three other methods. `Homo-SameC` considers a homogeneous battery system (every battery is the same type). Because it works by using either an LCO or NMC battery with the same C-rate, the performance of `Homo-SameC` is represented by the average performance of LCO and NMC. `Hetero-SoH` attempts to minimize the state-of-health (SoH, i.e., the degradation ratio) difference among all batteries. This method is commonly used in a multibattery system. `Hetero-StdC` attempts to follow the C-rate specified by the battery manufacturer. These four methods are tested for three scenarios with different C-rate demand: *totalC* = 4.2 A, 4.5 A, and 4.8 A. Also, other variables in Algorithm 1 are set as follows: *numT* = 2, *numN* = 2,



(a) (b)

Fig. 3. Capacity degradation comparison. (a) Degradation difference compared with `Hetero-Ours`. (b) Scenario of 4.8 A from 0 to 2000 Ah.

totalU = 2000 Ah, *listT* = [1, 1], *rangeT* = 2.0 A to 2.5 A, and X = 0.05 A.

Fig. 3(a) describes the difference of capacity degradation between the other three methods and that of `Hetero-Ours`. It indicates up to 16.95% and 26.22% of additional degradation are accumulated by the homogeneous method and the two heterogeneous methods, respectively, compared to `Hetero-Ours`. An interesting point is that the two heterogeneous methods show worse performance than the homogeneous method for some cases. This coincides with our description in the problem statement section—existing battery management methods do not necessarily work properly for the heterogeneous battery system. From Fig. 3(a) and (b), we can observe that a larger C-rate demand and usage increase capacity degradation. As capacity degradation increases, there is more room to optimize itself, which results in improvement of `Hetero-Ours` in general. However, there are some exceptions; for example, `Homo-SameC` works well in the 4.5 A case. In exceptional cases, these approaches happen to yield similar solutions to `Hetero-Ours` or at least better solutions than the same approaches with different scenarios. Despite such exceptional cases, our method *always* outperforms other methods and has a tendency to show more enhanced improvement in scenarios exhibiting substantial capacity degradation.

V. CONCLUSION

In this letter, we proposed a method to minimize the capacity degradation of heterogeneous batteries in a mobile embedded system. We also demonstrated the effectiveness of the proposed approach—up to 16.95% and 26.22% less battery capacity degradation compared to the corresponding homogeneous battery system and the corresponding heterogeneous battery system with existing algorithms, respectively.

In the future, we would like to extend this letter toward a more realistic system with heterogeneous batteries, with considering a thermal effect and nonfixed current demand.

REFERENCES

[1] S. Baek, M. Go, S. Lee, and H. Cha, “Exploiting multi-cell battery for mobile devices: Design, management, and performance,” in *Proc. 15th ACM Conf. Embedded Netw. Sensor Syst.*, 2017, p. 20.

[2] L. He, Y.-C. Tung, and K. G. Shin, “iCharge: User-interactive charging of mobile devices,” in *Proc. 15th Annu. Int. Conf. Mobile Syst. Appl. Services*, 2017, pp. 413–426.

[3] H. Li and L. Chen, “RSSI-aware energy saving for large file downloading on smartphones,” *IEEE Embedded Syst. Lett.*, vol. 7, no. 2, pp. 63–66, Jun. 2015.

[4] O. Sahin and A. K. Coskun, “On the impacts of greedy thermal management in mobile devices,” *IEEE Embedded Syst. Lett.*, vol. 7, no. 2, pp. 55–58, Jun. 2015.

[5] T. P. Hanusa, *The Lightest Metals: Science and Technology From Lithium to Calcium*. Chichester, U.K., Wiley, 2015.

[6] A. Badam *et al.*, “Software defined batteries,” in *Proc. 25th Symp. Oper. Syst. Principles*, 2015, pp. 215–229.

[7] L. Y. Wang *et al.*, “Balanced control strategies for interconnected heterogeneous battery systems,” *IEEE Trans. Sustain. Energy*, vol. 7, no. 1, pp. 189–199, Jan. 2016.

[8] H. Morais, P. Kádár, P. Faria, Z. A. Vale, and H. M. Khodr, “Optimal scheduling of a renewable micro-grid in an isolated load area using mixed-integer linear programming,” *Renew. Energy*, vol. 35, no. 1, pp. 151–156, 2010.

[9] Y. He, B. Venkatesh, and L. Guan, “Optimal scheduling for charging and discharging of electric vehicles,” *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1095–1105, Sep. 2012.

[10] D. Ferreira, A. K. Dey, and V. Kostakos, “Understanding human-smartphone concerns: A study of battery life,” in *Proc. Int. Conf. Pervasive Comput.*, 2011, pp. 19–33.

[11] X. Jin *et al.*, “Physically-based reduced-order capacity loss model for graphite anodes in li-ion battery cells,” *J. Power Sources*, vol. 342, pp. 750–761, Feb. 2017.

[12] X. Lin, J. Park, L. Liu, Y. Lee, A. M. Sastry, and W. Lu, “A comprehensive capacity fade model and analysis for li–ion batteries,” *J. Electrochem. Soc.*, vol. 160, no. 10, pp. A1701–A1710, 2013.

[13] C. Delacourt and M. Safari, “Life simulation of a graphite/lifepo$_4$ cell under cycling and storage,” *J. Electrochem. Soc.*, vol. 159, no. 8, pp. A1283–A1291, 2012.

[14] V. Ramadesigan, P. W. Northrop, S. De, S. Santhanagopalan, R. D. Braatz, and V. R. Subramanian, “Modeling and simulation of lithium–ion batteries from a systems engineering perspective,” *J. Electrochem. Soc.*, vol. 159, no. 3, pp. R31–R45, 2012.

[15] K. Kim, Y. Choi, and H. Kim, “Data-driven battery degradation model leveraging average degradation function fitting,” *Electron. Lett.*, vol. 53, no. 2, pp. 102–104, Jan. 2016.