energies

MDPI

Article

A Novel State of Charge Estimation Algorithm for Lithium-Ion Battery Packs of Electric Vehicles

Zheng Chen, Xiaoyu Li, Jiangwei Shen, Wensheng Yan and Renxin Xiao *

Faculty of Transportation Engineering, Kunming University of Science and Technology,
Kunming 650500, China; chen@kmust.edu.cn (Z.C.); xiaoyu_li187@163.com (X.L.);
shenjiangwei6@163.com (J.S.); yanwensheng65@sina.com (W.Y.)
* Correspondence: xrx1127@foxmail.com; Tel.: +86-137-6919-9607

Abstract: This paper focuses on state of charge (*SOC*) estimation for the battery packs of electric vehicles (EVs). By modeling a battery based on the equivalent circuit model (ECM), the adaptive extended Kalman filter (AEKF) method can be applied to estimate the battery cell *SOC*. By adaptively setting different weighed coefficients, a battery pack *SOC* estimation algorithm is established based on the single cell estimation. The proposed method can not only precisely estimate the battery pack *SOC*, but also effectively prevent the battery pack from overcharge and over-discharge, thus providing safe operation. Experiment results verify the feasibility of the proposed algorithm.

Keywords: adaptive extended Kalman filter (AEKF); electric vehicle (EV); state of charge (*SOC*); weighed coefficients

1. Introduction

Due to energy crisis and environmental issues, energy storage systems have been applied more and more widely in electric vehicles (EVs), power grids, consumer electronics equipment, etc. Lithium-ion batteries dominate a considerable percentage of these applications in view of their high energy density, low life cycle, and lack of memory effect [1]. In EVs or plug-in hybrid electric vehicles (PHEVs), a battery pack is usually comprised of tens to thousands of battery cells connected in series, parallel, or an even more complex topology. Hence, it is imperative to employ a so-called battery management system (BMS) to safeguard, manage, and oversee the pack to ensure its proper operation. An important task for the BMS is to estimate the battery's current statuses, which usually include state of charge (*SOC*) and state of health (SOH) [2]. This paper focuses on *SOC* estimation for the battery packs of EVs.

Battery *SOC*, according to its definition, represents the ratio of available battery capacity compared with the current battery rated capacity [3]. The available battery capacity can influence and even determine how long a battery can be fully charged, and consequently how far a vehicle can drive. Thus, it is imperative to estimate the battery *SOC* precisely. An accurate *SOC* can present the battery available capacity status, prevent a battery pack from overcharge or over-discharge, and can effectively protect the battery, thereby potentially prolonging the battery life. The most direct and easiest method of estimating *SOC* is the coulomb counting method, which is also called the ampere-hour (Ah) integration method. Currently, most vehicle original equipment manufacturers (OEMs) and BMS manufacturers adopt this method, along with a variety of look-up tables and threshold protections [4]; however, there exist three obvious defects with this method. First, it needs precise estimation of the initial *SOC* values, which usually relies on memorial values of the last operation or the help of the initially built relationship with battery open circuit voltage (OCV). Second, it may bring the drift of *SOC* measurement due to the offset of the current sensors. Third, it requires accurate battery current capacity estimation,

which may vary with the current, temperature, and operation cycles [5,6]. As a result, various research innovations have been spurred to overcome these estimation defects and to improve *SOC* precision [7].

Currently, there is an abundance of intelligent methods developed for battery cell *SOC* estimation, leading to a rich pool of publications. The purpose of all these methods is to compensate for error accumulation and initial *SOC* error induced by the coulomb counting method. Of these methods, the most widely employed include: neural networks [8–10], supported vector machines [11–13], fuzzy logic [14–16], and miscellaneous filters [8,10,13,17–23]. Among all the candidates, the most popular method is the Kalman filter and its variously extended formats, which mainly include the extended Kalman filter (EKF) [24,25], sigma-point Kalman filter [26], unscented Kalman filter (UKF) [17,27], adaptive extended Kalman filter (AEKF) [28], etc. The first three methods declare the advances of improving estimation precision and dispensing the initial *SOC* values; however, they need the estimation of covariance for state and measured variables, which may be unfeasible and vary with the operation conditions. Some methods even require a considerable quantity of computations. On the other hand, the AEKF can estimate the covariance of noises recursively and regressively, which varies with the external conditions, thus reducing the critical covariance estimation [29]. In view of these merits, the AEKF has attracted vast attention due to its enclosed solution for the whole *SOC* estimation.

The definition of *SOC* of a battery pack is much more complex than that of a single battery cell. This is because in a battery pack, there exists a variety of cells, of which the voltage and capacity cannot be guaranteed to be consistent due to differences in production engineering, materials, as well as the electric connection and internal temperature field distribution. Hence it is not accurate to regard a battery pack as some simple connection of battery cells when calculating *SOC*. In [30], the highest and lowest cell *SOC* values are quantified to calibrate the cell capacity. The final battery pack *SOC* can be determined based on the weighted average values with respect to all the *SOC* values and their rated capacities. This method relies on calculating each cell's *SOC* based on the AEKF method. This amount of computation cannot be accepted, however, due to the abundance of matrix transformation for each cell's AEKF-based estimation. In [31,32], a battery pack model is built to be regarded as an integrated cell; however, this method is not universal and needs substantial training based on a variety of measured data. The most popular method is to calculate the average value of all cells and then treat it as the pack value. In [15], the *SOC* of each cell is calculated based on EKF, and adaptive neuro-fuzzy inference system (ANFIS) is employed to calculate the pack value. This process is time-consuming and calculation-heavy. Another popular method for determining battery pack *SOC* is firstly calculating each cell's *SOC*, and then selecting the maximum cell *SOC* value when the battery is being charged and the minimum cell *SOC* value when the battery is being discharged as the pack *SOC*. However, this method has a few obvious disadvantages. First, it relies on each cell's *SOC* calculation, which requires considerable labor and storage memory, especially when hundreds of cells constitute a battery pack. Second, it can induce oscillations when the battery is transitioned between the charge process and discharge process, leading to an unsatisfactory user experience. When the battery is under discharge process, the lowest cell *SOC* should represent the battery pack to escape from over-discharge. When the EV operation transfers from driving mode to regeneration mode, the battery pack begins to charge, and now the highest cell *SOC* should become the pack *SOC* instead, thus preventing overcharge. During this process, the battery pack *SOC* transfers from the lowest *SOC* value to the highest cell *SOC* value. Supposing their difference is 5%, obviously, then the pack *SOC*'s oscillation is 5%. This is not acceptable during real optimization, even some OEMs usually adopt filters to decrease the oscillation to some extent. Third, it needs an arrangement algorithm to queue all the *SOC* values, which can also contribute to the calculation labor. Based on these considerations, this method is not applicable to estimate battery pack *SOC* in real applications of EVs.

From this point, it is imperative to build a battery pack *SOC* estimation algorithm which can firstly determine the overall battery pack *SOC* value, and secondly achieve a satisfactory user experience. Based on the authors' understanding, the user experience should include these four typical criteria. First, when one cell of the battery pack is fully charged, the pack *SOC* should be equal to 100%. Second,

when one cell is discharged to the lowest voltage threshold, the pack *SOC* should reach 0%. In addition, the pack *SOC* should not decrease when the battery is charged, and should not increase when the battery is discharged. Moreover, the battery pack *SOC* should not oscillate frequently. These demands motivate authors to conduct research in this area. In order to properly evaluate the *SOC* estimation, a powerful and effective battery model, which has a fast calculation speed, enough precision, and limited storage size, is necessary to quantify the battery electrochemical reaction. Currently, the electrochemical model can precisely estimate battery *SOC* based on calculating a variety of differential equations [33,34]. Usually, these kinds of models can be finished by some particular finite element analysis software, such as Ansys. The black box model, just like its name implies, employs some artificial intelligence tools, such as neural networks [22], supported vector machines [13], etc. to construct an invisible model to simulate battery performance. Obviously, this kind of model relies on a huge amount of experimental data to train the model output, i.e., battery *SOC* with inputs of the battery's measured voltage, current, as well as temperature. Therefore, it is not feasible to apply the black box model in a real-time operating BMS. The equivalent circuit model (ECM) usually consists of one or two resistance-capacitance (RC) networks, an OCV source, and a resistor, which are connected in series [5,35]. Due to its simple structure and relatively acceptable precision, the ECM has been widely used in BMSs for all kinds of lithium-ion batteries. In this paper, the ECM is employed to model a single cell, from which the battery pack model is built and validated. Based on the ECM, genetic algorithm (GA) is applied to optimally identify the model parameters. Then, AEKF is applied to estimate the single cell *SOC*. By setting a series of rules and threshold values, the pack *SOC* is determined based on the highest cell *SOC* and the lowest cell *SOC*. Finally, experiments examine and justify the effectiveness of the proposed method.

2. Battery Modeling

In a battery pack, there exist tens to thousands of cells connected in series or in parallel. The production engineering, the material performance and the thermal distribution lead to inconsistency of the battery cell performance. Balancing technologies are implemented to compensate for these difference to some extent. In order to determine the battery pack *SOC*, it is imperative to first model the cell with an acceptable precision. Then, the cell *SOC* can be estimated, thereby paving the road for the pack *SOC* estimation.

2.1. Battery Cell Modeling

An effective battery model should not only reflect a battery's dynamic and static characteristics, but also should be of simple structure and require easy calculations. In this paper, a lithium iron phosphate battery pack is considered as the research object, of which each cells' capacity is 5 Ah and the rated voltage is 3.3 V. By analyzing the electrochemical impedance map, a one-order ECM is adopted to characterize battery performance, which is suitable to be implemented in a BMS. As shown in Figure 1, the proposed model mainly encompasses four elements, i.e., an OCV source, one RC network, and two resistors.
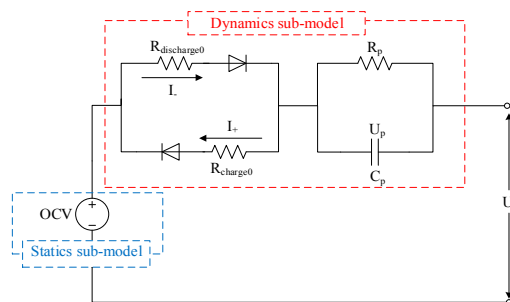
Figure 1. The equivalent circuit model (ECM) of the lithium-ion battery.

The main function of an OCV is to quantify the relationship between the open circuit terminal voltage and the battery *SOC*. The RC network can simulate the diffusion reaction when the battery is excited with different currents. The resistors can capture the immediate voltage drop when a discharge or charge operation occurs. It is necessary to mention that a series-connected resistor shows different values when transferring between charge and discharge operations, hence we employ two resistors connected in series with a diode, and parallel them to describe this phenomenon in the model.

According to Figure 1, the following equations can be formulated based on Kirchhoff's voltage law:

$$\begin{cases} \frac{dU_p}{dt} = \frac{I}{C_p} - \frac{U_p}{R_p C_p} \\ U_t = U_{ocv} + U_p + IR_0 \end{cases} \tag{1}$$

$$\begin{cases} R_0 = R_{discharge0} & (I_-) \\ R_0 = R_{charge0} & (I_+) \end{cases} \tag{2}$$

where I denotes the battery current, and U_{OCV}, U_t, and U_p are, respectively, the voltage drops of the open circuit, battery terminal, and R_p. In Equation (2), R_0 equals $R_{discharge0}$ and $R_{charge0}$ when the battery is discharged and charged, respectively. For ease of calculation, a discrete form of Equation (1) can be expressed as:

$$\begin{cases} U_{p,k+1} = U_{p,k} \exp\left(-\Delta T/\tau\right) + \left[1 - \exp\left(-\Delta T/\tau\right)\right] R_p I_k \\ U_{t,k+1} = U_{ocv,k+1} + U_{p,k+1} + I_{k+1} R_0 \end{cases} \tag{3}$$

$$\begin{cases} I_k = I_{-,k} & (\text{discharge}) \\ I_k = I_{+,k} & (\text{charge}) \end{cases} \tag{4}$$

where T represents the sampling interval and is set to be 1 s in this paper, and $\tau = R_p C_p$. Correspondingly, the cell *SOC* can be calculated according to the definition:

$$SOC = SOC_0 + \eta \int_0^t I(t)/C_b dt \tag{5}$$

where SOC_0 indicates the initial value of *SOC* and should be determined in advance with help of the preset relationship with the OCV. C_b is the battery rated capacity, and η is the integral coefficient varying with temperature, current rate, and battery degradation.

Based on Equations (3) and (5), and the definitions $x(t) = \begin{bmatrix} x_1(t) & x_2(t) \end{bmatrix}^T = \begin{bmatrix} U_p & SOC \end{bmatrix}^T$ and $y(t) = U_t(t)$, the following state space function can be obtained:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu \\ y(t) = Cx(t) + Du + U_{OCV} \end{cases} \tag{6}$$

where $A = \begin{bmatrix} -1/(R_p C_p) & 0 \\ 0 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 1/C_p & 1/C_b \end{bmatrix}^T$, $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $D = R_0$, and $u = I$.

Next, the model parameters need to be identified with accepted precision, and the model precision should be verified simultaneously.
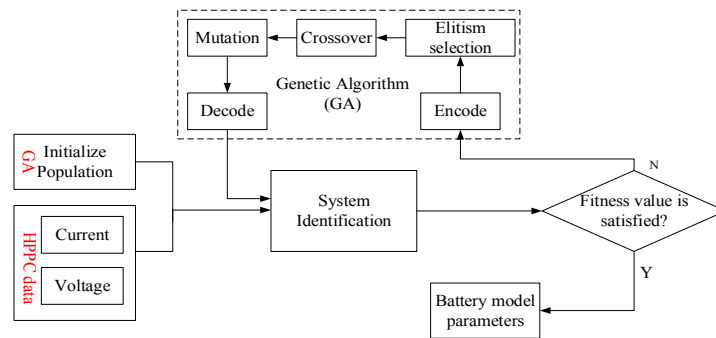
2.2. Parameters Identification Based on Genetic Algorithm

Currently, the least square method is widely employed to identify the model parameters [36–39]. This method is easy to implement and has acceptable local identification performance, though the overall identification precision will be compromised as the identified data quantity becomes large [40]. In this paper, GA is applied to estimate the battery parameters to gain the global optimal solution. GA is a stochastic method simulating the revolution of biology [41]. By a series of actions, including

elitism selection, crossover and mutation, GA can find the optimal solution with numerous iterations. The main objective of elitism selection is to pick the optimal offspring; the purpose of crossover is to generate new offspring by combing the performances of the current populations. The goal of mutation is to stochastically change some populations to prevent them from sinking into locally optimal solutions [25,41]. Currently, GA has been widely implemented in industry applications [42], such as automatic control, robotics, image processing, and machine learning. For these highly nonlinear systems with variable model parameters and multiple objects, GA can find optimal or suboptimal solutions conveniently with acceptable calculation speed.

The detailed identification schematic is shown in Figure 2. During the process of parameters identification, the voltage and current data are initially identified, and the identified model parameters are considered as the initial population. GA is then applied to identify the model parameters by a series of actions including crossover, mutation, and elitism selection. During the process of parameters identification, the measured current and battery terminal voltage are set as the input and output of the model, respectively. The absolute value of differences between voltage measurement and the model output is considered as the fitness value, as shown in Equation (7). By iteration, the model parameters can be eventually determined until the fitness value reaches a preset low threshold. In this paper, the low threshold is set as 0.01. Finally, the algorithm terminates and outputs the model parameters.

$$F = \left| V_{\text{model}} - V_{\text{sample}} \right| \tag{7}$$
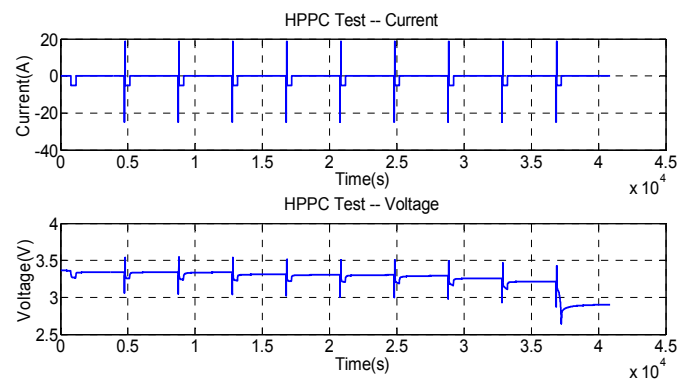

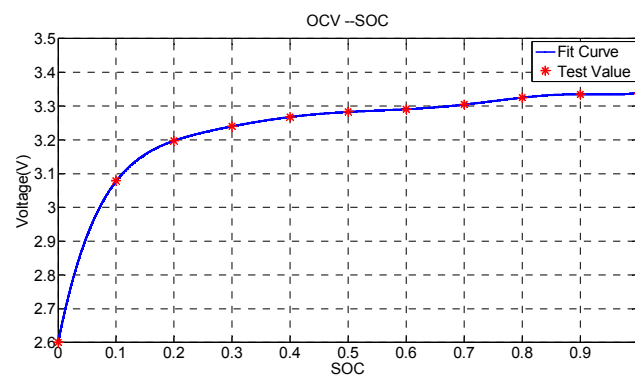**Figure 2.** Parameter identification based on genetic algorithm (GA).

In order to identify the model parameters, a hybrid pulse power characterization (HPPC) experiment is carried out to supply an extreme voltage response by exciting high current charge and discharge pulses, followed by a constant current discharge and a certain time rest. By this way, the battery dynamic performances can be determined, and subsequently the OCV response with respect to different *SOC* values can be quantified. By this experiment, the battery parameters can then be effectively identified by GA. As shown in Figure 3, the HPPC current profiles can be divided into 10 cycles, each of which contains a 5C current pulse charge, a 5C current pulse discharge, 1C current discharge, and 1 h rest, where C denotes the battery rated capacity with unit Ah. The battery rated capacity is 5 Ah, and thus C equals 5. The voltage thresholds for avoiding overcharge and over-discharge are 3.8 V and 2.6 V, respectively. The voltage response curve is also depicted in Figure 3. It can be observed that the OCV and the battery voltage reaction can be determined, thereby paving the road for fully identifying the battery parameters.

The OCV curve can be fitted using poly numerical equations, as displayed in Equation (8), of which the coefficients, i.e., P_1, P_2, P_3, P_4, P_5, P_6, P_7 and P_8 are equal to 70.9239, −280.5262, 453.0390, −386.2610, 188.2906, −53.2899, 8.5636 and 2.6, respectively. Figure 4 proves that the polynomial equations can effectively describe the nonlinear relationship between the OCV and *SOC*.

$$y = P_1 \times SOC^7 + P_2 \times SOC^6 + P_3 \times SOC^5 + P_4 \times SOC^4 + P_5 \times SOC^3 + P_6 \times SOC^2 + P_7 \times SOC + P_8 \tag{8}$$

Figure 3. The hybrid pulse power characterization (HPPC) profile.

Figure 4. Open circuit voltage (OCV) curve with respect to state of charge (*SOC*).

Based on the identification results, the battery model parameters, including $R_{charge0}$, $R_{discharge0}$, R_p, and C_p, are 0.0144 ohm, 0.0146 ohm, 0.0092 ohm and 3073.4 F, respectively. Figure 5 shows the model fitting error, which states that the maximum error between the measured voltage and the model output is 0.067 V, thereby proving that the constructed model and the corresponding parameters can accurately simulate the battery characterization.

With the built model and the identified model parameters, the AEKF algorithm is then applied to estimate the single cell *SOC*, and correspondingly the battery pack *SOC* is determined.
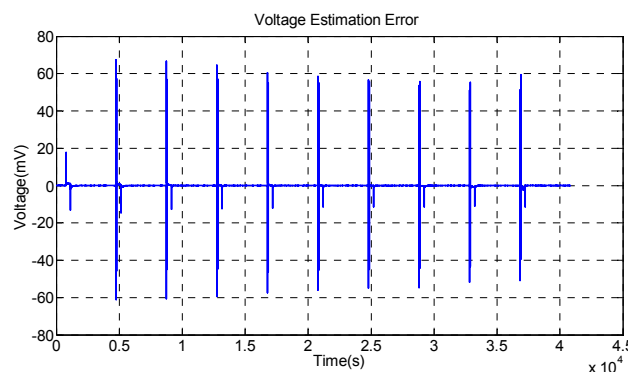
Figure 5. Voltage estimation error.

3. State of Charge Estimation Algorithm Design

In this paper, the AEKF algorithm is applied to estimate the cell *SOC* first, and then a novel pack *SOC* algorithm is proposed based on the cell estimation.

3.1. The Cell State of Charge Estimation Based on Adaptive Extended Kalman Filter

Compared with coulomb counting method, the main principle of applying the AEKF to estimate the cell *SOC* is that the AEKF can combine the output of coulomb counting method and the model output with Kalman gain as the weighed coefficient, thereby improving the estimation precision. In addition, it can adaptively regulate the Kalman gain automatically based on the difference between the measured voltage and the model output. In order to apply AEKF to estimate the cell *SOC*, the space state equation of the battery cell model is established based on Equation (6), and the discrete form can be formulated as:

$$\begin{cases} x_k = A_k x_{k-1} + B_k u_{k-1} + w_k \\ y_k = C_k x_k + D_k u_k + v_k \end{cases} \tag{9}$$

Here, A_k, B_k, C_k, and D_k are discrete forms of A, B, C, and D presented in Equation (6), as shown in Equation (10):

$$\begin{cases} A_k = \begin{bmatrix} e^{-\frac{T}{R_\mathrm{P}C_\mathrm{P}}} & 0 \\ 0 & 1 \end{bmatrix} & C_k = [1 \ \partial U_{\mathrm{ocv},k}/\partial SOC_k] \\[2em] B_k = \begin{bmatrix} R_\mathrm{P}\left(1 - e^{-\frac{T}{R_\mathrm{P}C_\mathrm{P}}}\right) \\ \frac{T}{3600 \times C_\mathrm{b}} \end{bmatrix} & D_k = [R_0] \end{cases} \tag{10}$$

where T is the sampling rate with 1 s as an interval and k denotes the steps. w_k and v_k are process noise and measurement noise, respectively, both of which are supposed to be Gauss white noise with zero average values. Their corresponding covariance values are Q_k and R_k, respectively. In this paper, by observation, the battery current varies from 0C to 1C ampere most of time. Only 3C current occurs when the vehicle is accelerated or decelerated sharply, while the duration is usually within several seconds. When the battery current is less than 1C ampere, the integral coefficient varies within 3%, and therefore can be regarded as unchanged. Based on these considerations, the integral coefficient, i.e., C_b, is supposed to be a constant. It is necessary to mention that when applying the *SOC* estimation method based on AEKF, it does not rely on a precise initial *SOC* value estimation, unlike the coulomb counting method. The *SOC* can gradually converge to the true value by iteration. It can be set to any arbitrary value between 0 and 1. In this paper, we set the initial value as 0.5.

Essentially, AEKF is based on the principle of minimizing the mean square error estimation to seek a recursive estimation algorithm, which is comprised of initialization, time update, and measurement update. The detailed process is listed as follows.

(a) Initialization of filtering equation:

$$\begin{cases} \hat{x}_0 = E(x_0) \\ P_0 = \mathrm{var}(x_0) \end{cases} \tag{11}$$

    where E and var are the mean value of the state variable and the covariance of the system, respectively.

(b) Time update, which includes state space update and covariance of error update:

$$\hat{x}_{k+1}^- = A_k \hat{x}_k^+ + B_k u_k + w_k \tag{12}$$

$$P_k^- = A_k P_k A_k^T + Q_k \tag{13}$$

    where $\hat{x}_{k+1}^-$ is a prior estimation of state x at step $k+1$, $\hat{x}_k^+$ denotes the poster estimation at step k, and u_k is the current input at step k.

(c) Measurement update, which involves Kalman gain calculation, state estimation measurement update, and error covariance update:

$$H_k = P_k^- C_k^T / \left(C_k P_k^- C_k^T + R_k \right) \tag{14}$$

$$\hat{x}_k^+ = \hat{x}_k^- + H_k \left(y_k - C_k \hat{x}_k^- - D_k u_k \right) \tag{15}$$

$$P_k = \left(I - H_k C_k \right) P_k^- \tag{16}$$

where $C_k = [1\ \partial U_{\mathrm{ocv},k} / \partial SOC_k]$, R_k can be determined by the voltage sensors during operation, y_k is the acquired measured voltage at step k, and I states a two-order unit matrix employed to update the error covariance.

According to Equations (12) and (13), the calculation of EKF requires the accurate estimation of Q_k and R_k, which cannot be estimated accurately during the EKF calculation process. The advantage of AEKF is that it can automatically and adaptively regulate Q_k and R_k during the actual calculation process [28]. In this manner, it can dispense the need of Q_k and R_k estimation, which may change stochastically with operating conditions. The adaptive estimation process is formulated as follows:

$$\begin{cases} e_j = y_k - C_k \hat{x}_k^- - D_k u_k \\ d_k = \frac{1}{N} \sum\limits_{j=k-N+1}^{k} e_j e_j^T \\ Q_k = H_k d_k H_k^T \\ R_k = d_k + C_k P_k C_k^T \end{cases} \tag{17}$$

where e_j is the remaining error. In order to recursively calculate Q_k and R_k, first the model remaining error e_j is calculated, and then d_k is the calculation which states covariance of e_j with a moving estimation window of size N. Based on the movingly updated d_k, Q_k and R_k can be recursively estimated based on the model errors and their covariance. By this manner, Q_k and R_k can adaptively vary with the error magnitude, thereby realizing the online estimation.

3.2. Battery Pack State of Charge Estimation

For all the cell voltages in a battery pack, there exists a maximum value $V_{\max}$ and a minimum value $V_{\min}$. Suppose their corresponding *SOC* values are $SOC_{\max}$ and $SOC_{\min}$, respectively, which are calculated based on the above AEKF method. In this paper, we only consider $SOC_{\max}$ and $SOC_{\min}$ to determine the final pack *SOC*. Thus, it can be said that only two iterations of AEKF are applied, independent of battery cells connected in series. Consequently, the calculation labor can be substantially reduced compared with what is required when calculating each cell's *SOC*. In this paper, two weighed coefficients, w_1 and w_2, are considered, which change within $SOC_{\max}$ and $SOC_{\min}$. The final battery pack *SOC* output can be formulated as,

$$SOC_{\mathrm{pack}}(k) = SOC_{\max}(k) \times w_1(k) + SOC_{\min}(k) \times w_2(k) \tag{18}$$

where $w_1(k)$ and $w_2(k)$ range from 0 to 1. In order to properly determine $w_1(k)$ and $w_2(k)$, the vehicle operation status and the customer experience should be analyzed in detail. During the battery operation, customers usually pay attention to two extreme statuses including the status when the battery is almost fully charged and the status when the battery is almost fully discharged. When the battery is almost fully charged, the battery pack *SOC* needs to converge to the highest *SOC* gradually, thus ensuring an acceptable user experience and simultaneously preventing overcharge. When the battery is near empty, similarly, the battery pack *SOC* should be close to the lowest cell *SOC* with fast speed. Thus, it can effectively avoid over-discharge. In addition, when the battery status is in between, two weighed coefficients are employed to smoothly transfer the *SOC* estimation between the highest cell *SOC* and the lowest cell *SOC*. Therefore, three classifications are made based on $SOC_{\max}(k)$, $SOC_{\min}(k)$, and their differences. By this manner, the pack *SOC* output will not oscillate greatly while ensuring the safe operation. The detailed calculation process is shown in Figure 6. It is necessary to

mention that only when the difference between $SOC_{max}(k)$ and $SOC_{min}(k)$ is more than 0.5 will a BMS consider a fault to have occurred, at which point it will then stop the algorithm calculation and send out alarms.
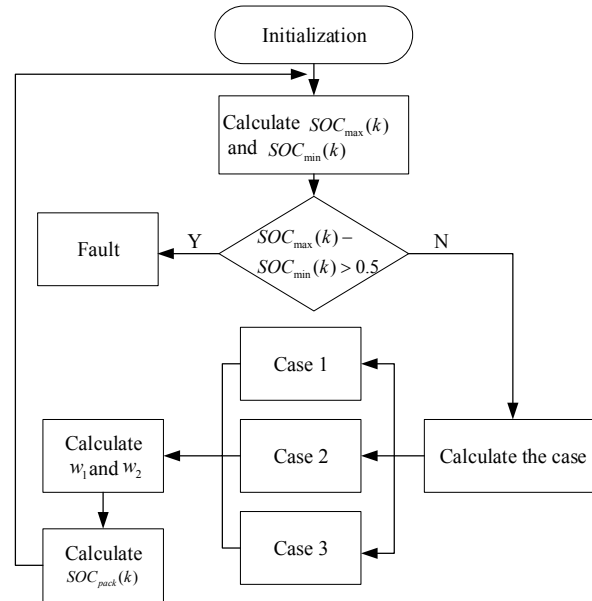
Figure 6. The calculation flowchart for the pack *SOC* estimation.

(a) Case 1:

Condition: $SOC_{max}(k) - SOC_{min}(k) < SOC_{pack}(k-1) < 1 - (SOC_{max}(k) - SOC_{min}(k))$

Determination:

$$\begin{cases} w_1(k) = (SOC_{max}(k) + SOC_{min}(k))/2 \\ w_2(k) = 1 - (SOC_{max}(k) + SOC_{min}(k))/2 \end{cases} \tag{19}$$

In this case, $w_1(k)$ equals with the mean value of $SOC_{max}(k)$ and $SOC_{min}(k)$, and $w_2(k) = 1 - w_1(k)$. Thus according to Equation (18), $SOC_{pack}(k)$ can smoothly transfer between $SOC_{max}(k)$ and $SOC_{min}(k)$.

(b) Case 2:

Condition: $SOC_{pack}(k-1) \leq SOC_{max}(k) - SOC_{min}(k)$

Determination:

$$\begin{cases} w_1(k) = 0 \\ w_2(k) = (SOC_{max}(k) - SOC_{min}(k))/SOC_{min}(k) \end{cases} \tag{20}$$

This case considers the situation where the battery is being discharged with relatively large cell *SOC* difference. Now the algorithm becomes a tracking system, where $SOC_{pack}(k)$ tries to catc $SOC_{min}(k)$. $w_1(k)$ equals 0, and $w_2(k)$ can reflect the tracking time and the tracking rate. Here $w_2(k)$ is supposed to be a constant and equals its initial values when the estimation transfers from Case 1 to Case 2.

(c) Case 3:

Condition: $SOC_{pack}(k-1) \leq 1 - (SOC_{max}(k) - SOC_{min}(k))$

Determination:

$$\begin{cases} w_1(k) = (SOC_{max}(k) - SOC_{min}(k))/(1 - SOC_{max}(k)) \\ w_2(k) = 0 \end{cases} \tag{21}$$

Similar to Case 2, when the battery is being charged, $SOC_{\text{pack}}(k)$ begins to catch $SOC_{\text{max}}(k)$. Hence, $w_1(k)$ becomes the transferring rate and $w_2(k)$ equals 0.

It can be observed that these three boundaries dynamically change with $SOC_{\text{max}}(k)$ and $SOC_{\text{min}}(k)$, and are correlated with their differences. The main reason why we classify the SOC estimation process into three cases is that during the end process of both charging and discharging, the weighed coefficients change more obviously than the average threshold values shown in Equation (19). Thus, the pack SOC can converge to $SOC_{\text{min}}(k)$ and $SOC_{\text{max}}(k)$ at a faster speed, compared to the calculation that considers only Case 1.

Next step, experiment validation, is performed to verify the effectiveness of the proposed algorithm.

4. Experiment Validation

In this paper, the experiment validation is divided into two steps. First, the experiment is carried out in the laboratory, which uses a two-parallel and twelve-series connected battery pack as the test platform. The voltage and current values of each cell and battery pack can be measured based on the BMS module. During the validation, each cell's SOC is estimated and the pack SOC is determined based on these estimations. Second, the program is verified using an electric logistics vehicle, shown in Figure 7, which consists of a battery pack with 150 modules connected in series and 28 cells connected in parallel as a module. The rated capacity of the battery pack is 140 Ah, and the rated voltage is 540 V. When the vehicle is operated, the related measurements of the battery pack can be obtained through a monitoring platform.

Figure 7. The tested electric logistic vehicle.

4.1. Laboratory Validation

As shown in Figure 8, a battery test system NEWARE BTS-400 (NEWARE, Shenzhen, China) is employed to charge and discharge the battery pack in the laboratory. A BMS along with a monitoring unit is responsible for measuring the cell and pack voltage and current, calculating the SOC and communicating with the host computer with controller area network (CAN) bus.
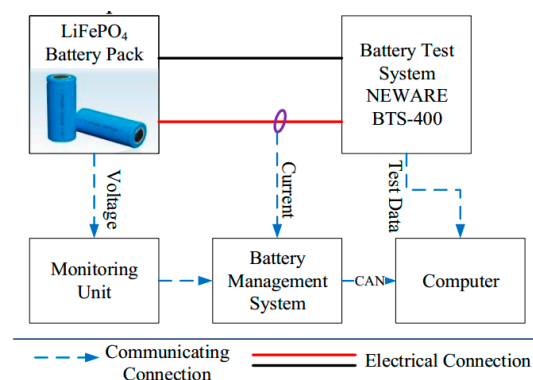

Figure 8. Test bench.

Figure 9 shows the cell *SOC* estimation based on the AEKF method. It can be observed that the proposed method can converge to the reference value effectively. The difference between the highest *SOC* and lowest *SOC* is around 0.1. The analysis of the error for the lowest *SOC* is also shown in Figure 9, which states that the maximum error is 0.65%. Thus, our experiment proves that the AEKF can estimate cell *SOC* precisely.
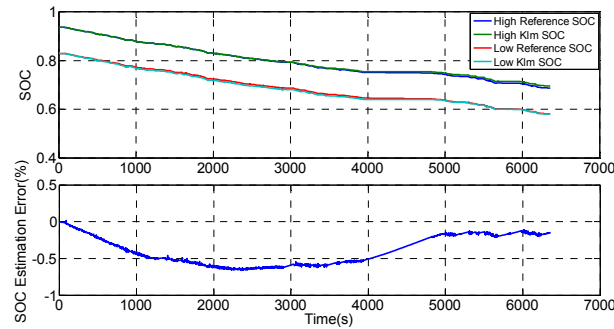

Figure 9. Battery cell *SOC* estimation and lowest *SOC* estimation error.

Figure 10 shows R_k variation when estimating the lowest cell *SOC*, and it can be seen that R_k varies from 0 to 0.00126 dynamically. After sorting all the voltage values, and calculating one *SOC* for every two voltage measurements, six cells' *SOC* values in total are obtained, as shown in Figure 11.
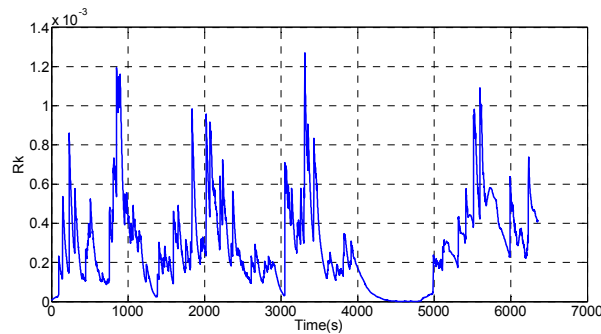

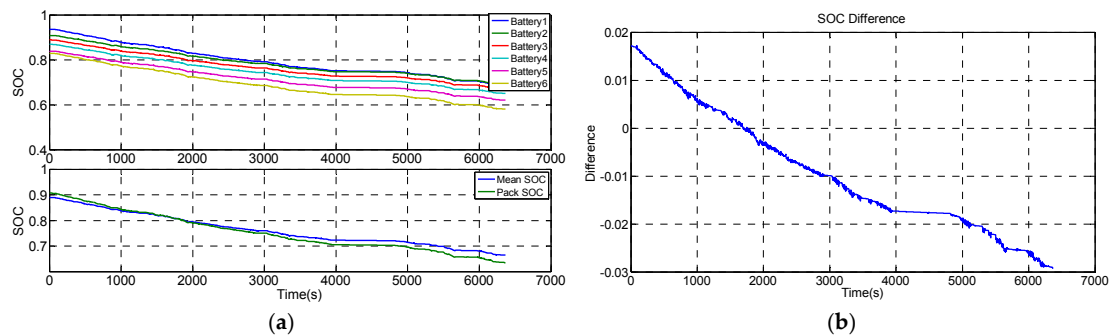**Figure 10.** The R_k variation when calculating the lowest cell *SOC*.

(**a**)　　　　　　　　　　　　　　　　　　(**b**)

Figure 11. The *SOC* output comparison based on different methods: (**a**) The cell *SOC* estimation and the pack *SOC* estimation based on different methods; and (**b**) the pack *SOC* difference.

Based on the proposed algorithm, the final battery pack *SOC* can be determined. In order to feature the performance of the proposed algorithm, the pack *SOC* is also calculated by averaging all the cell *SOC* values. Figure 11 also shows their differences. It can be found that based on the proposed method, the battery pack *SOC* smoothly transfers between $SOC_{\max}$ and $SOC_{\min}$. When SOC_{pack} is

relatively high, it is near SOC_{max}; when SOC_{pack} is relatively low, it decreases faster and becomes nearer to SOC_{min}. Compared with the average cell SOC method, the pack SOC based on the proposed method is higher when the battery begins to be discharged. The difference becomes smaller as the SOC decreases. It can be observed that the pack SOC based on the proposed method varies more obviously than that based on the average method. To sum up, the experiment proves that the pack SOC can effectively transfer between SOC_{max} and SOC_{min}. In this way, the proposed algorithm not only can safeguard the safety of the battery pack both when running and charging, but also can produce a satisfactory user experience.

4.2. Electric Vehicle Validation

During the vehicle validation, only the cell current, the highest and lowest voltages, the corresponding cell SOC, and the pack SOC are broadcast due to the limited calculation capability. The highest voltage and lowest voltage are obtained, as presented in Figure 12. It can be found that the difference between the maximum voltage and minimum voltage is more than 0.05 V. Meanwhile, the battery current is collected and the pack current is translated into the cell current based on the average distribution, also shown in Figure 12.
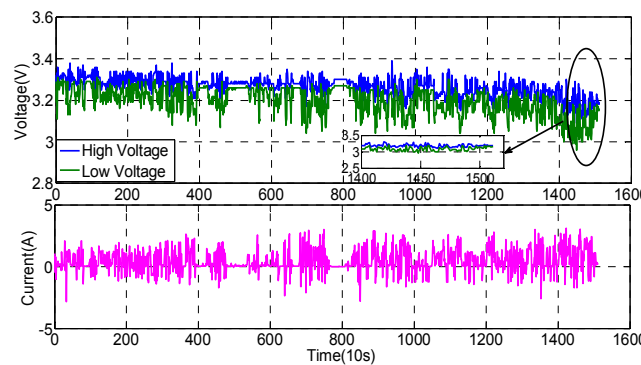

Figure 12. The highest and lowest cell voltages as well as the cell current.

According to the current profile, the working condition consists of acceleration, deceleration, and idle operating mode, which cover most of the operating conditions of vehicles. Based on the proposed algorithm and the average cell SOC method, SOC_{max}, SOC_{min}, and two SOC_{pack} values are calculated according to the collection voltage and current, as shown in Figure 13. Figure 13 also shows the pack SOC differences when two different methods are applied. It can be clearly observed that the pack SOC based on the proposed method varies more obviously compared with that based on the average cell SOC method.
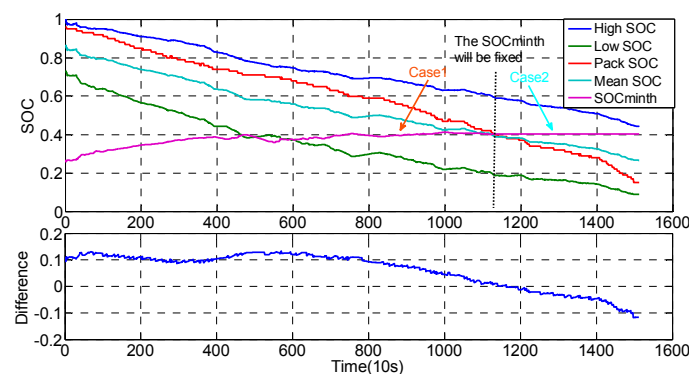

Figure 13. SOC comparison based on different methods.

Figure 14 shows the calculation case transitions. It can be observed than SOC_{pack} transfers from SOC_{max} to SOC_{min} when the battery is discharged, and the calculation process alters from Case 1 to Case 2. When the battery is being discharged, SOC_{max} is 97% and SOC_{pack} is 96%, while when the discharge process ends, SOC_{min} is 9% and the pack value is 15%. When the algorithm reaches Case 2, w_2 is taken into account and w_1 becomes zero. This transition can make SOC_{pack} converge to SOC_{min} more quickly to protect the battery cell from over-discharge using gain adjustment. Therefore, the experiment proves that the proposed algorithm can estimate the pack SOC more reasonably by only calculating two cells' SOC, compared with the average cell SOC method. In addition, it can transit from the highest SOC to the lowest SOC smoothly, and can thus provide a friendly user experience.
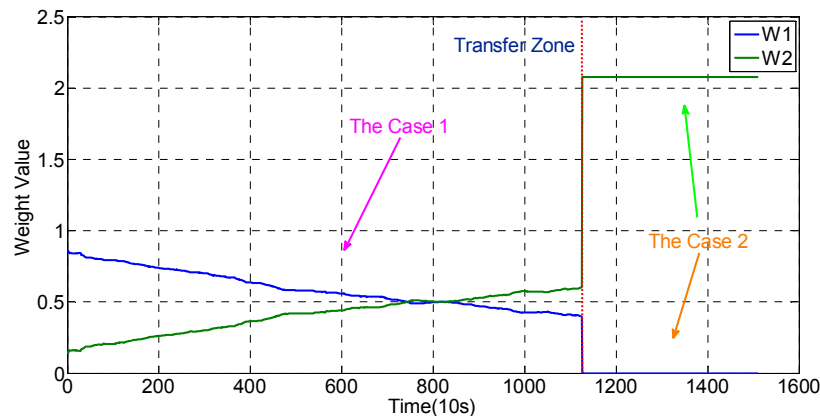

Figure 14. The variation of weighed coefficients.

5. Conclusions

In this paper, a novel battery SOC estimation algorithm is proposed based on cell SOC estimation and adaptive classification. GA is applied to optimally identify the cell model parameters, and the overall identification error is less than 1%. Two adaptively changed coefficients are introduced to recursively calculate the pack SOC value based on the maximum and minimum cell SOC values. Experiment results prove the feasibility of the proposed algorithm. To sum up, the proposed algorithm can not only decrease the operation labor, but is also easy to implement in a BMS with the premise of ensuring precision and a satisfactory user experience.

Future work will focus on considering SOH influence in battery packs to further improve the algorithm's robustness across the whole life cycle. In addition, the influences to the pack SOC induced by the internal temperature distribution of the battery pack can also be our next step work direction.

Author Contributions: Zheng Chen and Xiaoyu Li conceived this paper, carried out the battery pack test cycles experiment and completed the paper writing. Renxin Xiao discussed the AEKF algorithm and revised the paper. Jiangwei Shen conducted the HPPC experiment and parameter identification. Wensheng Yan provided some valuable suggestions.

References

1. Lawder, M.T.; Suthar, B.; Northrop, P.W.C.; De, S.; Hoff, C.M.; Leitermann, O. Battery Energy Storage System (BESS) and Battery Management System (BMS) for Grid-Scale Applications. *Proc. IEEE* **2014**, *102*, 1014–1030. [CrossRef]

2.   Lu, L.; Han, X.; Li, J.; Hua, J.; Ouyang, M. A review on the key issues for lithium-ion battery management in electric vehicles. *J. Power Sources* **2013**, *226*, 272–288. [CrossRef]

3.   Xiong, R.; Sun, F.; Chen, Z.; He, H. A data-driven multi-scale extended Kalman filtering based parameter and state estimation approach of lithium-ion polymer battery in electric vehicles. *Appl. Energy* **2014**, *113*, 463–476. [CrossRef]

4.   Zheng, Y.; Ouyang, M.; Lu, L.; Li, J.; Zhang, Z.; Li, X. Study on the correlation between state of charge and coulombic efficiency for commercial lithium ion batteries. *J. Power Sources* **2015**, *289*, 81–90. [CrossRef]

5.   Chen, Z.; Xia, B.; Mi, C.C. A novel state-of-charge estimation method for lithium-ion battery pack of electric vehicles. In Proceedings of the 2015 IEEE Transportation Electrification Conference and Expo (ITEC), Metro Detroit, MI, USA, 14–17 June 2015.

6.   Berecibar, M.; Gandiaga, I.; Villarreal, I.; Omar, N.; van Mierlo, J.; van den Bossche, P. Critical review of state of health estimation methods of Li-ion batteries for real applications. *Renew. Sustain. Energy Rev.* **2016**, *56*, 572–587. [CrossRef]

7.   Rahimi-Eichi, H.; Ojha, U.; Baronti, F.; Chow, M. Battery management system: An overview of its application in the smart grid and electric vehicles. *IEEE Ind. Electron. Mag.* **2013**, *7*, 4–16. [CrossRef]

8.   Dang, X.; Yan, L.; Xu, K.; Wu, X.; Jiang, H.; Sun, H. Open-circuit voltage-based state of charge estimation of lithium-ion battery using dual neural network fusion battery model. *Electrochim. Acta* **2016**, *188*, 356–366. [CrossRef]

9.   Dong, G.; Zhang, X.; Zhang, C.; Chen, Z. A method for state of energy estimation of lithium-ion batteries based on neural network model. *Energy* **2015**, *90*, 879–888. [CrossRef]

10.   Shen, Y. Adaptive online state-of-charge determination based on neuro-controller and neural network. *Energy Convers. Manag.* **2010**, *51*, 1093–1098. [CrossRef]

11.   Hu, J.N.; Hu, J.J.; Lin, H.B.; Li, X.P.; Jiang, C.L.; Qiu, X.H. State-of-charge estimation for battery management system using optimized support vector machine for regression. *J. Power Sources* **2014**, *269*, 682–693. [CrossRef]

12.   Patil, M.A.; Tagade, P.; Hariharan, K.S.; Kolake, S.M.; Song, T.; Yeo, T. A novel multistage support vector machine based approach for Li ion battery remaining useful life estimation. *Appl. Energy* **2015**, *159*, 285–297. [CrossRef]

13.   Sheng, H.; Xiao, J. Electric vehicle state of charge estimation: Nonlinear correlation and fuzzy support vector machine. *J. Power Sources* **2015**, *281*, 131–137. [CrossRef]

14.   Cai, C.H.; Du, D.; Liu, Z.Y. Battery state-of-charge (SOC) estimation using adaptive neuro-fuzzy inference system (ANFIS). In Proceedings of the 12th IEEE International Conference on Fuzzy Systems (FUZZ'03), St. Louis, MO, USA, 25–28 May 2003; pp. 1068–1073.

15.   Dai, H.; Guo, P.; Wei, X.; Sun, Z.; Wang, J. ANFIS (adaptive neuro-fuzzy inference system) based online *SOC* (State of Charge) correction considering cell divergence for the EV (electric vehicle) traction batteries. *Energy* **2015**, *80*, 350–360. [CrossRef]

16.   Zheng, Y.; Ouyang, M.; Lu, L.; Li, J.; Han, X.; Xu, L. On-line equalization for lithium-ion battery packs based on charging cell voltages: Part 2. Fuzzy logic equalization. *J. Power Sources* **2014**, *247*, 460–466. [CrossRef]

17.   He, W.; Williard, N.; Chen, C.; Pecht, M. State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation. *Int. J. Electr. Power Energy Syst.* **2014**, *62*, 783–791. [CrossRef]

18.   Kang, L.; Zhao, X.; Ma, J. A new neural network model for the state-of-charge estimation in the battery degradation process. *Appl. Energy* **2014**, *121*, 20–27. [CrossRef]

19.   Sakthivel, R.; Anbuvithya, R.; Mathiyalagan, K.; Prakash, P. Combined H∞ and passivity state estimation of memristive neural networks with random gain fluctuations. *Neurocomputing* **2015**, *168*, 1111–1120. [CrossRef]

20.   He, Y.; Liu, X.; Zhang, C.; Chen, Z. A new model for State-of-Charge (SOC) estimation for high-power Li-ion batteries. *Appl. Energy* **2013**, *101*, 808–814. [CrossRef]

21.   Weigert, T.; Tian, Q.; Lian, K. State-of-charge prediction of batteries and battery-supercapacitor hybrids using artificial neural networks. *J. Power Sources* **2011**, *196*, 4061–4066. [CrossRef]

22.   Xu, L.; Wang, J.; Chen, Q. Kalman filtering state of charge estimation for battery management system based on a stochastic fuzzy neural network battery model. *Energy Convers. Manag.* **2012**, *53*, 33–39. [CrossRef]

23.   Wang, Y.; Zhang, C.; Chen, Z. On-line battery state-of-charge estimation based on an integrated estimator. *Appl. Energy* **2015**. [CrossRef]

24. Plett, G.L. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation. *J. Power Sources* **2004**, *134*, 277–292. [CrossRef]

25. Xiao, R.; Shen, J.; Li, X.; Yan, W.; Pan, E.; Chen, Z. Comparisons of modeling and state of charge estimation for lithium-ion battery based on fractional order and integral order methods. *Energies* **2016**, *9*, 184. [CrossRef]

26. Plett, G.L. Sigma-point Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1: Introduction and state estimation. *J. Power Sources* **2006**, *161*, 1356–1368. [CrossRef]

27. He, W.; Williard, N.; Chen, C.; Pecht, M. State of charge estimation for electric vehicle batteries using unscented kalman filtering. *Microelectron. Reliab.* **2013**, *53*, 840–847. [CrossRef]

28. Han, J.; Kim, D.; Sunwoo, M. State-of-charge estimation of lead-acid batteries using an adaptive extended Kalman filter. *J. Power Sources* **2009**, *188*, 606–612. [CrossRef]

29. Xiong, R.; He, H.; Sun, F.; Zhao, K. Evaluation on state of charge estimation of batteries with adaptive extended Kalman filter by experiment approach. *IEEE Trans. Veh. Technol.* **2013**, *62*, 108–117. [CrossRef]

30. Zhong, L.; Zhang, C.; He, Y.; Chen, Z. A method for the estimation of the battery pack state of charge based on in-pack cells uniformity analysis. *Appl. Energy* **2014**, *113*, 558–564. [CrossRef]

31. Sun, F.; Xiong, R. A novel dual-scale cell state-of-charge estimation approach for series-connected battery pack used in electric vehicles. *J. Power Sources* **2015**, *274*, 582–594. [CrossRef]

32. Sun, F.; Xiong, R.; He, H. A systematic state-of-charge estimation framework for multi-cell battery pack in electric vehicles using bias correction technique. *Appl. Energy* **2016**, *162*, 1399–1409. [CrossRef]

33. Cuadras, A.; Kanoun, O. *SOC* Li-ion battery monitoring with impedance spectroscopy. In Proceedings of the 6th International Multi-Conference on Systems, Signals and Devices (SSD '09), Djerba, Tunisia, 23–26 March 2009.

34. Domenico, D.D.; Fiengo, G.; Stefanopoulou, A. Lithium-ion battery state of charge estimation with a Kalman Filter based on an electrochemical model. In Proceedings of the IEEE International Conference on Control Applications (CCA 2008), San Antonio, TX, USA, 3–5 September 2008; pp. 702–707.

35. Lee, S.; Kim, J.; Lee, J.; Cho, B.H. State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge. *J. Power Sources* **2008**, *185*, 1367–1373. [CrossRef]

36. Sun, D.; Chen, X. Adaptive parameter identification method and state of charge estimation of Lithium ion battery. In Proceedings of the 17th IEEE International Conference on Electrical Machines and Systems (ICEMS), Hangzhou, China, 22–25 October 2014; pp. 855–860.

37. Andre, D.; Appel, C.; Soczka-Guth, T.; Sauer, D.U. Advanced mathematical methods of SOC and SOH estimation for lithium-ion batteries. *J. Power Sources* **2013**, *224*, 20–27. [CrossRef]

38. Tong, S.; Klein, M.P.; Park, J.W. On-line optimization of battery open circuit voltage for improved state-of-charge and state-of-health estimation. *J. Power Sources* **2015**, *293*, 416–428. [CrossRef]

39. Chen, Z.; Mi, C.C.; Fu, Y.; Xu, J.; Gong, X. Online battery state of health estimation based on Genetic Algorithm for electric and hybrid vehicle applications. *J. Power Sources* **2013**, *240*, 184–192. [CrossRef]

40. Chen, Z.; Fu, Y.; Mi, C.C. State of charge estimation of lithium-ion batteries in electric drive vehicles using extended Kalman filtering. *IEEE Trans. Veh. Technol.* **2013**, *62*, 1020–1030. [CrossRef]

41. Chen, Z.; Mi, C.C.; Xiong, R.; Xu, J.; You, C. Energy management of a power-split plug-in hybrid electric vehicle based on genetic algorithm and quadratic programming. *J. Power Sources* **2014**, *248*, 416–426. [CrossRef]

42. Yang, X.S. Genetic Algorithms. In *Nature-Inspired Optimization Algorithms*; Elsevier: Oxford, UK, 2014; Chapter 5; pp. 77–87.