energies

MDPI

Article

A Cubature Particle Filter Algorithm to Estimate the State of the Charge of Lithium-Ion Batteries Based on a Second-Order Equivalent Circuit Model

Bizhong Xia, Zhen Sun, Ruifeng Zhang * and Zizhou Lao

Division of Advanced Manufacturing, Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China; xiabz@sz.tsinghua.edu.cn (B.X.); sz15@mails.tsinghua.edu.cn (Z.S.); lzz15@mails.tsinghua.edu.cn (Z.L.)
* Correspondence: zrf223@126.com; Tel.: +86-188-9836-6838

Academic Editor: Izumi Taniguchi
Received: 13 January 2017; Accepted: 23 March 2017; Published: 1 April 2017

Abstract: The state of charge (*SOC*) is the residual capacity of a battery. The *SOC* value indicates the mileage endurance, and an accurate *SOC* value is required to ensure the safe use of the battery to prevent over- and over-discharging. However, unlike size and weight, battery power is not easily determined. As a consequence, we can only estimate the *SOC* value based on the external characteristics of the battery. In this paper, a cubature particle filter (CPF) based on the cubature Kalman filter (CKF) and the particle filter (PF) is presented for accurate and reliable *SOC* estimation. The CPF algorithm combines the CKF and PF algorithms to generate a suggested density function for the PF algorithm based on the CKF. The second-order resistor-capacitor (RC) equivalent circuit model was used to approximate the dynamic performance of the battery, and the model parameters were identified by fitting. A dynamic stress test (DST) was used to separately estimate the accuracy and robustness of the CKF and the CPF algorithms. The experimental results show that the CPF algorithm exhibited better accuracy and robustness than the CKF algorithm.

Keywords: state of charge; cubature Kalman filter; cubature particle filter; electric vehicles; lithium-ion battery

1. Introduction

With the problems due to industrial development and environmental pollution, electric vehicles are becoming more popular as an environmentally-friendly mode of transportation. The battery management system (BMS) is an important component of electric vehicles. The main role of the BMS is to provide a reasonable algorithm to utilize the function of the battery at an appropriate level for the working environment and provide feedback to the driver [1,2]. The state of charge (*SOC*) of a battery is functionally equivalent to a fuel gauge indicating the life of the battery pack in a battery-containing electric vehicle. *SOC* estimation is an important function of the BMS, and is defined as the ratio of the remaining capacity of the battery to the rated capacity. Accurate *SOC* estimation can maximize the performance of the battery and protect the battery to prevent overcharge and over discharge. The *SOC* value should provide drivers information about the status of the electric vehicle so that drivers can develop a reasonable driving plan. However, it is difficult to measure *SOC* directly and it is typically estimated from direct measurement variables. Some approaches have been tested and found to provide precise estimation of battery *SOC*, but these methods are protracted, costly, and interrupt main battery performance. It is impossible to make intuitive *SOC* value measurements. Although *SOC* value exhibits a monotonous relationship with the battery open circuit voltage, the *SOC* value is very sensitive to the change of battery voltage, and even small voltage changes will translate to significant

changes in the *SOC* value. Overall, it is a significant challenge to obtain an accurate value of *SOC*. For this reason, estimation of the *SOC* value is a preferred approach.

Previous studies have proposed many methods for *SOC* estimation, such as the Coulomb counting method (ampere-hour (Ah) integration method) [3–5], the open circuit voltage method [5,6], the BP (back-prorogation) neural network algorithm [7], Kalman filtering algorithm [8,9], and the particle filter (PF) method [10]. Additionally, some improved algorithms based on these methods have been proposed, such as the adaptive Kalman filter [11], the unscented Kalman filter (UKF) [12–15], the strong tracking cubature Kalman filter (STCKF) [16], and the unscented PF [17]. The Coulomb counting method is widely used for *SOC* estimation of electric vehicles because it is a simple algorithm. This method uses the *SOC* formula to compute the current integration and obtain the charged or discharged capacity. However, the current sensor itself has an intrinsic measurement error. During estimation, this current measurement error is integrated, and the error increases. The open-circuit voltage method is based on the monotonous relationship of the open-circuit voltage between the battery and the *SOC*. In this method, the open-circuit voltage must be measured for a sufficient period of time. An additional challenge of accurate measurement is that the *SOC* value for a lithium-ion battery is highly sensitive to changes in open circuit voltage. Rumelhart and McCelland proposed a neural network algorithm using an error BP training algorithm [18]. The BP network is the simplest of the human brain abstraction and simulation models using input and output samples and a corresponding training network to achieve an input-output mapping function relationship. However, the neural network method requires a significant amount of sample data for training, and the estimation error is greatly affected by the quality of the training data and the training method. Additionally, the hardware requirements are high.

The Kalman filter (KF) is an autoregressive optimal data processing algorithm proposed by Kalman in 1960 [19]. Its core idea is to make the best estimate of the minimum variance in the system state. The KF algorithm overcomes the error accumulation effect of the coulomb counting method that occurs with increased time. The KF algorithm does not depend on an accurate initial *SOC* value, but can improve the *SOC* value accuracy. However, the accuracy of this method depends on the establishment of a battery equivalent model, and some physical properties of the battery model are nonlinear. The EKF algorithm [20,21] and the UKF algorithm are improved KF algorithms. The EKF algorithm implements recursive filtering by linearizing nonlinear functions [22], and the UKF algorithm applies nonlinear system equations to the standard Kalman filter system by means of unscented transformation (UT). UT is a mathematical function used to estimate the result of applying a given nonlinear transformation to a probability distribution that is characterized by a finite set of statistics. Compared with the EKF algorithm, the UKF algorithm exhibits higher accuracy and has a wider application range, making it well-suited for solving nonlinear problems [23].

The PF method is a random sampling-based filtering method used to solve non-linear non-Gaussian problems [24,25]. The rationale of this method is to use a series of weighted random sample sets (particles) in the state space to approximate the posterior probability density function of the system states. Although the PF algorithm is relatively simple, there may be significant loss of accuracy during sampling. The main way to decrease the degradation of the PF method is to increase the number of particles and resample. However, re-sampling will reduce the diversity of particles, and a large increase in the number of particles will greatly increase the required time of the calculation. Therefore, it is necessary to select the importance density function of the PF algorithm.

The cubature Kalman filter (CKF) algorithm is a kind of filtering algorithm that is free of differentiation and high in filtering precision [16,26–28]. In this paper, we present the cubature particle filter (CPF) method for *SOC* estimation. The CKF algorithm directly calculates the mean and variance of the state using the numerical integration method based on the cubature principle, and generates the importance density function of the PF algorithm using the mean and the variance. This approach combines aspects of the CKF algorithm and the PF algorithm. Through the dynamic stress test (DST) and New European Driving Cycle (NEDC) tests, we verified that the accuracy and robustness of the CPF algorithm is superior to that of the CKF algorithm.

The remainder of this paper is arranged as follows. The second-order resistor-capacitor (RC) equivalent circuit model and the model parameters are presented in Section 2; Section 3 introduces the CKF algorithm and the CPF algorithm in detail; Section 4 describes the schematic of the battery test; And in Section 5, the verification results and comparisons of the CKF algorithm and CPF algorithm for accuracy and robustness are presented, and as well as the summary and conclusions.

2. Battery Equivalent Circuit Model and Identification of Model Parameters

2.1. Battery Equivalent Circuit Model

Modeling of battery performance *SOC* is often required during *SOC* estimation. Many models have been proposed, including the electrochemical model [29], electrochemical-thermal coupling model [30,31], and performance models that are often used to describe the external characteristics of the battery during operation. The equivalent circuit model consists of resistors, capacitors, voltage sources, and other circuit components to simulate the dynamic characteristics of the battery. For the use of circuit and mathematical methods for analysis, the equivalent circuit model is the most widely used battery model.

The accuracy and computational complexity of estimating *SOC* vary for different battery models. The selection of circuit model should be based on whether the equivalent circuit model can reasonably reflect the characteristics of the battery and the CPU processing power of the BMS. A more complex circuit model may be able to accurately represent the performance characteristics of the battery but use of a more complex model can increase the computational complexity of the *SOC* calculation and increase the workload of the CPU. However, a simplified equivalent circuit model may not well represent the performance characteristics of the battery, lowering the *SOC* estimation accuracy.

The selection of a reasonable equivalent circuit model is important for *SOC* estimation. Many different equivalent circuit models have been proposed, such as the resistance model [32,33], the first-order RC equivalent circuit model [6,33], the second-order RC equivalent circuit model [34], and other models [35–37]. The resistor model is simple and computationally small, thus it requires a low CPU load but provides poor accuracy. The complex equivalent circuit models exhibit high precision for determining the dynamic voltage characteristic of a battery, but the model is computational complex and the CPU load is heavy. The first-order RC equivalent circuit model and the second-order RC equivalent circuit model achieve a good balance between computational complexity and the ability to provide approximate accuracy of battery characteristics, with the second-order equivalent circuit exhibited better accuracy for the estimation of battery characteristics. Therefore, for this study, we selected a second-order RC equivalent circuit model. The second-order RC equivalent circuit model includes a voltage source $U_{oc}(SOC)$ with monotonicity to the *SOC* and a resistor R_0 and two sets of resistors and capacitors $((R_1, C_1) \ ; \ (R_2, C_2))$ in parallel from the circuit. The structure of the circuit model is shown in Figure 1.

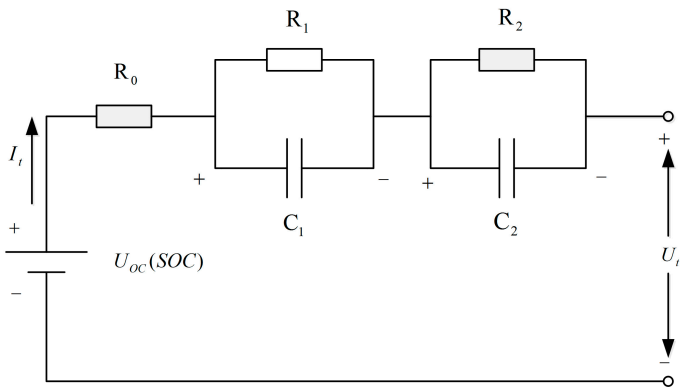


Figure 1. Model structure of a lithium-ion battery.

SOC is usually defined as shown in Equation (1):

$$SOC(t) = SOC(t_0) - \frac{\int_{t_0}^{t} idt}{Q_N} \Rightarrow \dot{SOC}(t) = -\frac{i(t)}{Q_N} \tag{1}$$

where $SOC(t)$ and $SOC(t_0)$ represent *SOC* values at times t and t_0, respectively; $i(t)$ represents the value of the current at time k; and Q_N indicates the rated capacity of the battery. $\dot{SOC}(t)$ is the derivative of $SOC(t)$.

According to Kirchhoff's law, the following equations (Equations (2)–(4)) are obtained from the second-order RC equivalent circuit model:

$$\frac{U_1}{R_1} + C_1\frac{dU_1}{dt} = i(t) \Rightarrow \dot{U_1} = \frac{i(t)}{C_1} - \frac{U_1}{C_1R_1} \tag{2}$$

$$\frac{U_2}{R_2} + C_2\frac{dU_2}{dt} = i(t) \Rightarrow \dot{U_2} = \frac{i(t)}{C_2} - \frac{U_2}{C_2R_2} \tag{3}$$

$$U_t = U_{oc}(SOC) - U_1 - U_2 - R_0 i(t) \tag{4}$$

U_1 and U_2 denote the terminal voltage of C_1 and C_2, respectively; $\dot{U_1}$ and $\dot{U_2}$ are the derivatives of U_1 and U_2, respectively; U_t and $i(t)$ represent the value of the terminal voltage and current, respectively. $U_{oc}(SOC)$ indicates the open circuit voltage of the battery (under the same environmental conditions, the open-circuit voltage value and the *SOC* value are monotonous.)

The equation of state for the discretization of the system is:

$$\begin{pmatrix} SOC(k) \\ U_1(k) \\ U_2(k) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1-\frac{T}{C_1R_1} & 0 \\ 0 & 0 & 1-\frac{T}{C_2R_2} \end{pmatrix} \begin{pmatrix} SOC(k-1) \\ U_1(k-1) \\ U_2(k-1) \end{pmatrix} + \begin{pmatrix} -\frac{T}{Q_N} \\ \frac{T}{C_1} \\ \frac{T}{C_2} \end{pmatrix} i(k) \tag{5}$$

Taking the terminal voltage as the observed value, we obtain the observed equation:

$$U_t(k) = U_{oc}(SOC) - R_0 i(k) - U_1(k) - U_2(k) \tag{6}$$

2.2. Parameter Identification

The second-order system model of the battery was used as follows. For *SOC* estimation, some parameters in the model must be identified in advance, including $U_{oc}(SOC)$ and the values of $\{R_0, R_1, R_2, C_1, C_2\}$. Under the same temperature conditions, an ICR18650-22F lithium-ion battery was tested as follows: (1) the battery was charged to the upper limit cut-off voltage at 4.2 V, and left standing for 2 h; (2) the battery was discharged to the lower limit cutoff voltage at 2.75 V, and the discharged capacity was recorded as the battery capacity Q_N; (3) repeat step (1), and the battery was continuously discharged at a rate of 0.1C, and was allowed to stand for 2 h after discharging the battery at a rated capacity of 10%; and (4) repeat step (3) until the battery is completely discharged.

As shown in Figure 2, the blue-solid triangles reflect the relationship between $U_{oc}(SOC)$ and *SOC* after 2 h of standing in the test step. The experimental results were fitted by a sixth-order polynomial fitting curve, and the results were approximate to the relationship between $U_{oc}(SOC)$ and *SOC*. The values of $\{R_0, R_1, R_2, C_1, C_2\}$ were obtained by fitting the data for a certain discharge and the standing process of the battery in the experimental step by exponential function fitting. The identified parameters are listed as follows:

$R_0 = 0.0361\Omega$, $R_1 = 0.0218\Omega$, $R_2 = 0.0051\Omega$, $C_1 = 1658.8$F, $C_2 = 87751$F.

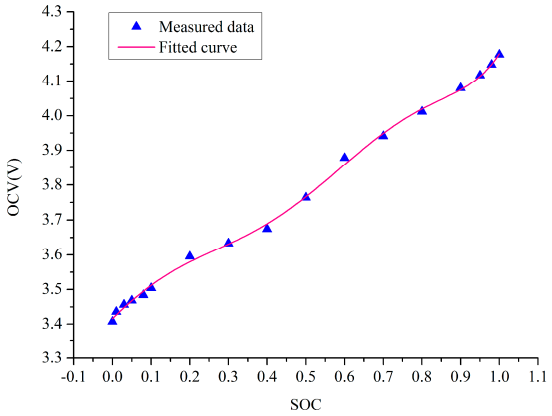


Figure 2. $U_{oc}(SOC)$ identification. *SOC*: state of charge.

3. Flow of State of Charge Estimation Algorithm

3.1. Cubature Kalman Filtering Algorithm (CKF)

Consider the following nonlinear non-Gaussian discrete-time dynamic systems:

$$\begin{cases} x_{k+1} = f(x_k) + w_k \\ y_{k+1} = h(x_{k+1}) + v_{k+1} \end{cases} \tag{7}$$

where x_{k+1} and y_{k+1} are the system state vector and the measured value at time $k+1$; $f(\cdot)$ and $h(\cdot)$ are the system state transition model function and the measurement model function, respectively; w_k and v_{k+1} are process noise and observation noise, respectively.

(1) Initialization:

$$\hat{x}_0 = E[x_0], P_0 = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^{\mathrm{T}}] \tag{8}$$

(2) Time update:

Calculate the cubature points:

$$P_k = S_k S_k^{\mathrm{T}} \tag{9}$$

$$X_{j,k} = S_k \xi_i + \hat{x}_k \qquad i = 1, 2, \cdots, 2n \tag{10}$$

where n represents the state dimension, the total number of volume points is 2 times the state dimension, and the volume point is calculated as shown in Equation (11):

$$\xi_j = \sqrt{n}\,[1]_j \qquad j = 1, \cdots, 2n \tag{11}$$

where $[1]_j$ indicates that the point is centered at the j-th point of $[1]$, and the symbol $[1]$ is a complete set of all symmetric points, denoting the set of points generated by the full permutation of the elements of the n-dimensional unit vector $e = [1, 0, \cdots, 0]^{\mathrm{T}}$ and the change of the element symbol.

Calculate the cubature points that are propagated through the state equation:

$$X_{j,k+1|k} = f(x_{j,k}) \tag{12}$$

Calculate the square root of state prediction and variance prediction:

$$\hat{x}_{k+1|k} = \frac{1}{2n} \sum_{i=1}^{2n} x_{k+1|k}^{i} \tag{13}$$

$$P_{k+1|k} = \frac{1}{2n}\sum_{i=1}^{2n} X_{k+1|k}^{i}(X_{k+1|k}^{i})^{\mathrm{T}} - \hat{x}_{k+1|k}(\hat{x}_{k+1|k})^{\mathrm{T}} + Q_k \tag{14}$$

where Q_k denotes the process noise covariance at time k.

(3) Measurement update:

Calculate the cubature points:

$$P_{k+1|k} = S_{k+1|k}S_{k+1|k}^{\mathrm{T}} \tag{15}$$

$$x_{j,k+1|k} = S_{k+1|k}\zeta_j + \hat{x}_{k+1|k} \tag{16}$$

Calculate the volume points propagated through the measurement equation and prediction:

$$y_{j,k+1} = h(x_{j,k+1|k}) \tag{17}$$

$$\hat{y}_{k+1} = \frac{1}{2n}\sum_{i=1}^{2n} y_{j,k+1} \tag{18}$$

Calculate the predicted covariance:

$$P_{y,k+1} = \frac{1}{2n}\sum_{i=1}^{2n} y_{j,k+1}(y_{j,k+1})^{\mathrm{T}} - \hat{y}_{k+1}(\hat{y}_{k+1})^{\mathrm{T}} + R_{k+1} \tag{19}$$

$$P_{xy,k+1} = \frac{1}{2n}\sum_{i=1}^{2n} x_{j,k+1|k}(x_{j,k+1|k})^{\mathrm{T}} - \hat{x}_{j,k+1|k}(\hat{x}_{j,k+1|k})^{\mathrm{T}} \tag{20}$$

where R_{k+1} denotes the measurement noise covariance at time $k+1$.

(4) Calculate the gain matrix, the state of time $k+1$ and the square root of the variance estimates:

$$K_{k+1} = P_{xy,k+1}(P_{y,k+1})^{-1} \tag{21}$$

$$\hat{x}_{k+1} = \hat{x}_{k+1|k} + K_{k+1}(y_{k+1} - \hat{y}_{k+1}) \tag{22}$$

$$P_{k+1} = P_{k+1|k} - K_{k+1}P_{y,k+1}K_{k+1}^{\mathrm{T}} \tag{23}$$

3.2. Cubature Particle Filtering Algorithm

CPF applies the CKF to the PF framework. At time k, the mean value and variance of the importance density function are calculated by the CKF algorithm based on the observation data [16,26]. A new posterior probability density distribution is generated, and a new posterior probability distribution can re-generate the new particles, calculate the weights of the particles, perform normalization, and then perform resampling to complete the state estimation. The implementation of the CPF algorithm is as follows:

(1) Initialize to obtain initial particles $\hat{x}_0^{(i)}$ and corresponding variance $\hat{P}_0^{(i)}$, i.e., to extract particles $x_0^{(i)}(1, \cdots, N)$ from the initial state probability distribution density $p(x_0)$, and:

$$\hat{x}_0^{(i)} = E[x_0^{(i)}] \tag{24}$$

$$\hat{P}_0^{(i)} = E[(x_0^{(i)} - \overline{x}_0^{(i)})(x_0^{(i)} - \overline{x}_0^{(i)})^{\mathrm{T}}] \tag{25}$$

(2) When $k = 1, 2, \cdots$, proceed as follows:

If the particle set with equal weights $(1/N)$ at time $k-1$ is $\{x_{k-1}^{(i)}, \hat{P}_{k-1}^{(i)}, i = 1, \cdots, N\}$, the particles are subjected to cubature Kalman filtering, and the particle set and covariance $(\hat{x}_k^{(i)}, \hat{P}_k^{(i)})$ are obtained.

The density function of choice is the Gaussian distribution function with $\hat{x}_k^{(i)}$ as mean and $\hat{P}_k^{(i)}$ as variance as follows:

$$q(x_k^{(i)} \Big| x_{k-1}^{(i)}, z_{1:k}) = N(\hat{x}_k^{(i)}, \hat{P}_k^{(i)}) \tag{26}$$

The importance density function defined in the formula is used to generate the predicted particle set and the corresponding variance $\{\hat{x}_k^{(i)}, \hat{P}_k^{(i)}, i = 1, \cdots, N\}$. Because the sampling points of each particle have different Gaussian distributions and any non-Gaussian distribution can include a series of different Gaussian distributions, it is reasonable to assume the importance density function as a Gaussian distribution.

The calculation of the weight of each particle:

$$w_k^{(i)} = w_k^{(i-1)} \frac{p(z_k \Big| x_k^{(i)}) p(x_k^{(i)} \Big| x_k^{(i-1)})}{q(x_k^{(i)} \Big| x_k^{(i-1)}, z_k)} \tag{27}$$

Normalized weights:

$$\overline{w}_k^{(i)} = \frac{w_k^{(i)}}{(\sum_{j=1}^{N} w_k^{(j)})} \tag{28}$$

Resampling is performed as described in [38].

Resampling judgment condition:

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N} (w_k^i)^2} < N_{threshold} \tag{29}$$

The particle set $\{\hat{x}_k^{(i)}, \hat{P}_k^{(i)}, i = 1, \cdots, N\}$ is resampled corresponding to the normalized weight $\overline{w}_k^{(i)}$ to obtain the set of particles $\{x_k^{(i)}, \hat{P}_k^{(i)}, i = 1, \cdots, N\}$ with equal weight $1/N$.

Output the state and variance estimates at time k:

$$\hat{x}_k = \frac{1}{N} \sum_{i=1}^{N} x_k^{(i)} \tag{30}$$

$$\hat{P}_k = \frac{1}{N} \sum_{i=1}^{N} x_k^{(i)} x_k^{(i)\mathrm{T}} - \hat{x}_k \hat{x}_k^{\mathrm{T}} \tag{31}$$

The cubature particle filtering process outlined in Figure 3.

4. Experimental Configurations and Analysis

4.1. Experimental Configurations

The schematic of the battery test bench is shown in Figure 4. It consists of: (1) lithium-ion batteries; (2) a control board (Sunwoda, Shenzhen, China) for the control of battery charge/discharge; (3) a personal computer with monitoring software for data sampling and MATLAB R2010a (MathWorks, Natick, MA, USA)) for data analysis; (4) a DC contactor (Schneider, Beijing, China) for charge/discharge switching; (5) a power supply (ITECH IT6952A, ITECH Electronics, Nanjing, China) for cell charging and (6) a programmable electric load (ITECH IT8510, ITECH Electronics, Nanjing, China) for cell discharging.

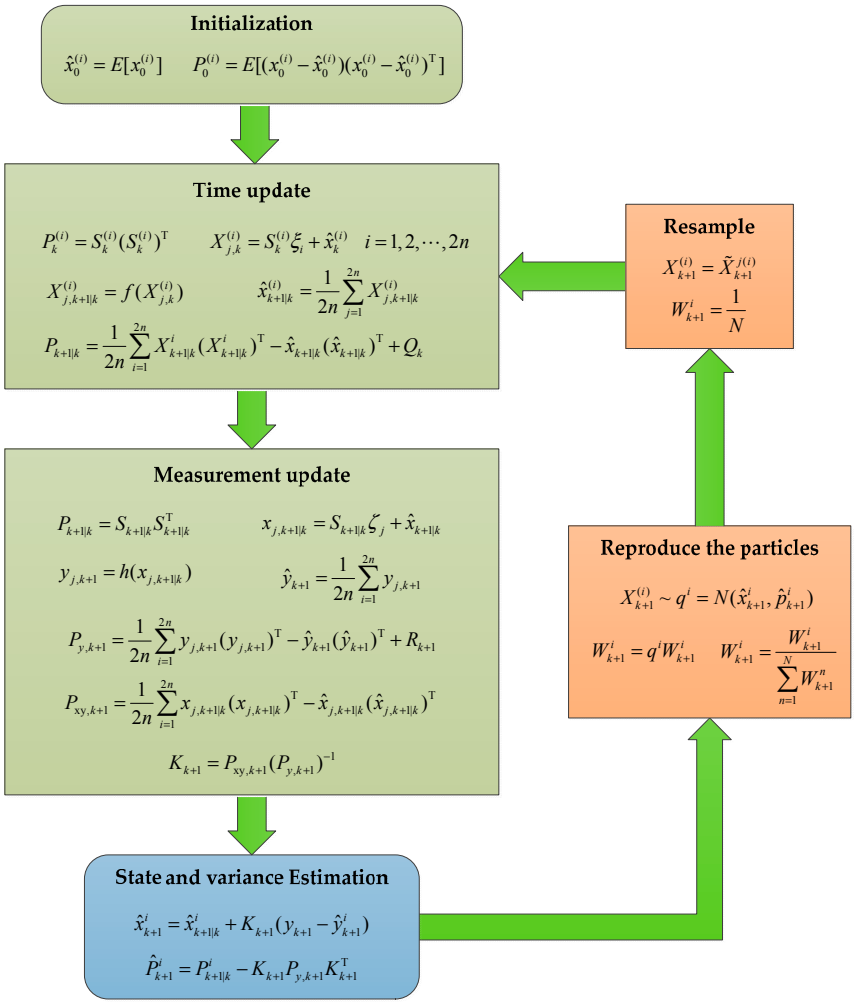


Figure 3. Cubature particle filtering (CPF) algorithm process.

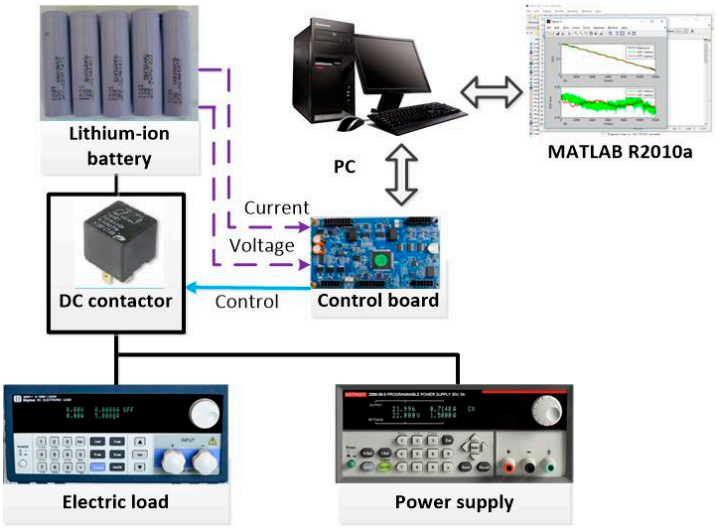


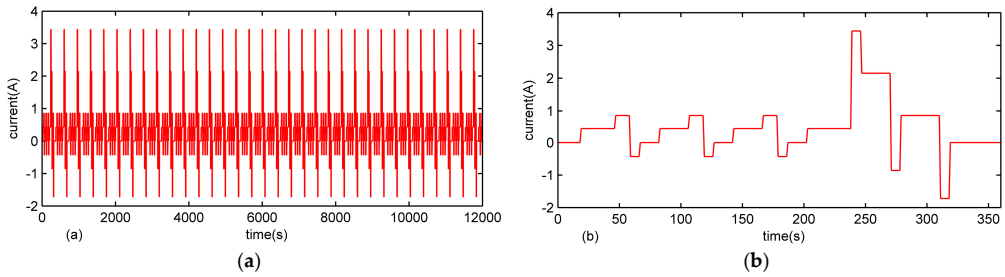
Figure 4. Schematic of the battery test bench.

Samsung ICR18650-22F lithium-ion batteries (Samsung SDI, Seoul, Korea) were used, with a nominal voltage and nominal capacity of 3.62 V and 2.2 Ah, respectively.

4.2. Experimental Analysis

4.2.1. Dynamic Stress Test

The DST tests the function of a battery under dynamic test conditions that are generated from the simplification of federal urban driving stress (FUDS) test [39]. To test whether the CPF algorithm is more accurate for *SOC* estimation than the CKF algorithm, we used the DST to conduct battery charge/discharge tests. The results are shown in Figure 5. In the experiment, 12,000 s of data were collected for analysis as shown in Figure 5a. A representative experimental cyclic unit is shown in Figure 5b.



(**a**) (**b**)

Figure 5. Dynamic stress testing (DST): (**a**) current profile and (**b**) magnified plot of the current profile.

Figure 6 shows the performance of the algorithm under the DST. The red solid-line and green solid-line are the estimated results from the CPF and EKF method, respectively. The capacity of the battery was obtained experimentally as described in Section 2.2. According to Equation (1), the reference *SOC* value shown in Figure 6a was obtained by integrating the charge and discharge current per second in the DST. To more easily compare the methods, an initial *SOC* value of 1 was used for the CKF and CPF algorithms, equal to the initial measured value. The experimental results are shown in Figure 6b and Table 1. The average absolute error and maximum error of the CPF algorithm were 0.5% and 2.6%, and the average absolute error and maximum error of the CKF algorithm were 1% and 3.8%. Thus, the CPF algorithm is more accurate than the CKF algorithm. As shown in Figure 7a, the blue solid-line represents the measured voltage and the red solid-line represents the voltage estimated by the CPF method. The blue solid-line and the red solid-line exhibit the same trend of change, with less than 0.04 voltage error, as shown in Figure 7b.

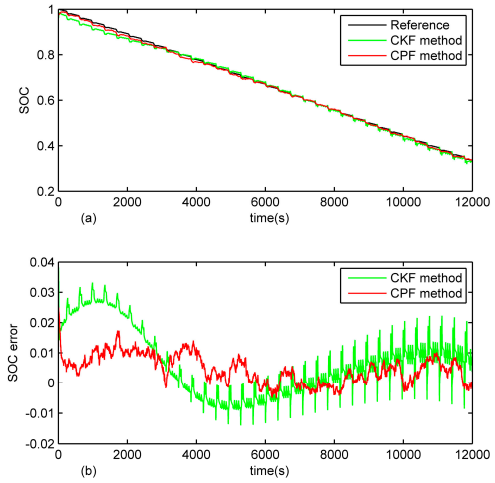


Figure 6. *SOC* estimation as determined by DST: (**a**) *SOC* estimation; and (**b**) *SOC* estimation error.

Table 1. Comparison of the performance of the cubature Kalman filtering (CKF) and CPF algorithms in the DST test.

Estimation Method	CKF	CPF
Mean absolute error	0.010	0.005
Maximum error	0.038	0.026

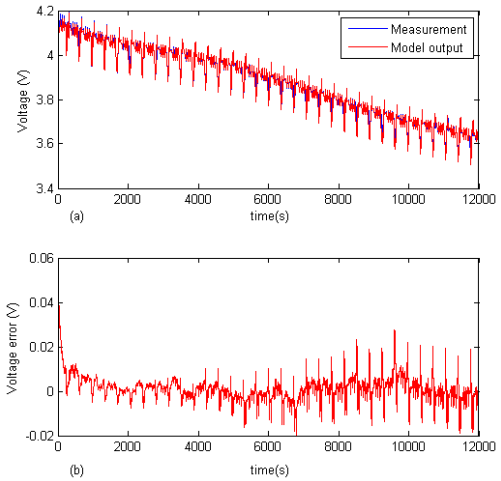


Figure 7. Voltage estimation under DST: (**a**) voltage; and (**b**) voltage error.

To compare the convergence rate of the CPF and CKF algorithms, the initial value of the *SOC* was reset to 0.5, significantly different from the experimental *SOC* values. The experimental results are shown in Figure 8. The red solid-line and green solid-line are the estimated results of the CPF and EKF methods. The convergence rate of the CKF algorithm was faster than the convergence rate of the CPF algorithm in the early stage of *SOC* estimation but then the convergence rate of CKF algorithm became slower than that of the CPF algorithm. Intersection occurs when the estimated error approaches zero, therefore, we can conclude that these convergence rates are approximate.

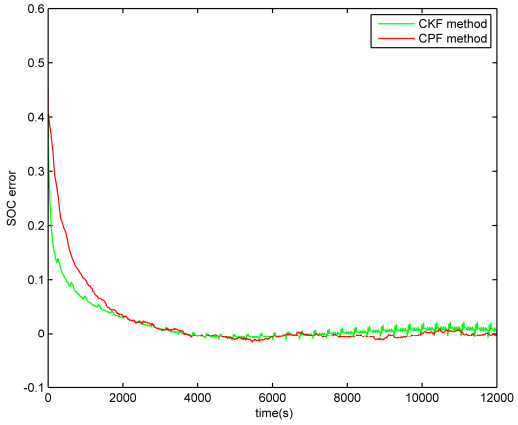


Figure 8. Comparison of convergence rates between the CKF and CPF algorithms.

4.2.2. New European Driving Cycle Operating Conditions

During actual *SOC* estimation, there will be noise in the data due to measurement errorss of the current sensor and voltage sensor and interference from the measurement environment. This increases the requirements for the robustness of the algorithm. Thus, we next tested NEDC conditions to assess

the robustness of the algorithm. The current profile and magnified profile for NEDC are shown in Figure 9.

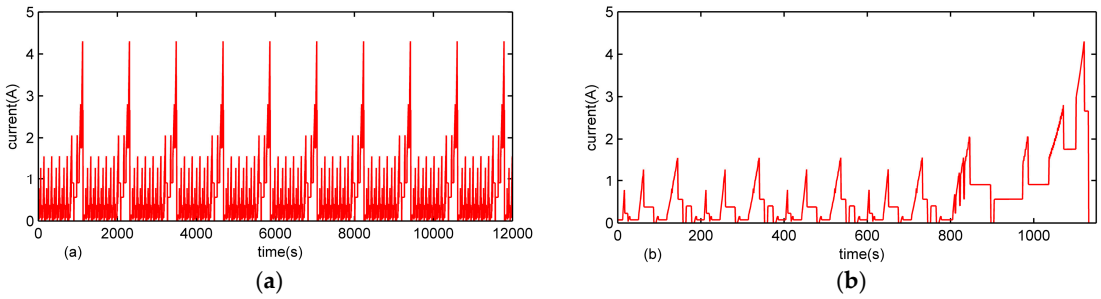


Figure 9. New European driving cycle (NEDC): (**a**) current profile and (**b**) magnified plot of the current profile.

The 0.03 V voltage random noise and the 0.1 A current random noise can be controlled by the circuit board in the actual process are added to verify the robustness of the algorithm. The red solid-line and green solid-line are the estimated results of the CPF and EKF methods in Figures 10–12.

When 0.03 V voltage random noise was added, the average absolute error and the maximum error of *SOC* estimation by the CKF method were 1.5% and 6.8%, respectively, as presented in Figure 10 and Table 2. The estimation accuracy of the CPF method was lower, 0.5% and 2.9%, respectively. Figure 11 and Table 3 showed that when the absolute value of the current random noise was less than 0.1A, the average absolute error and the maximum error of *SOC* estimation using the CKF method were 0.7% and 3.9%, respectively, and 0.6% and 2.6%, respectively, for the CPF method. As shown in Figure 12 and Table 4, the average absolute error and the maximum error of *SOC* estimation using the CKF method were 1.1% and 4.8%, respectively, and the accuracy of the CPF method was better than that of the CKF method, 0.5% and 2.9%, respectively. The results show that the average absolute error and maximum error of *SOC* estimation using the CPF method were lower than that of the CKF method. Overall, we conclude that the robustness of the CPF method is superior to that of the CKF method.

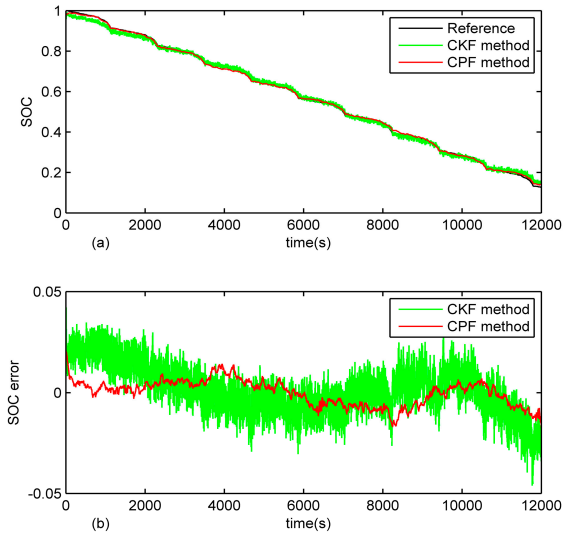


Figure 10. 0.03 V voltage random noise experiment results: (**a**) *SOC*; and (**b**) *SOC* error.

Table 2. Performance of the CKF and CPF algorithms under NEDC conditions with 0.03 V voltage random noise.

Estimation Method	CKF	CPF
Mean absolute error	0.015	0.005
Maximum error	0.068	0.029

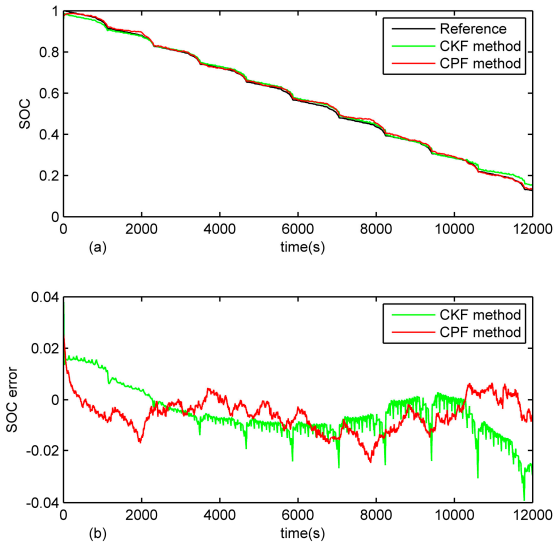


Figure 11. 0.1 A current random noise experiment results.

Table 3. Performance of the CKF and CPF algorithms under the NEDC conditions with 0.1 A current random noise.

Estimation Method	CKF	CPF
Mean absolute error	0.007	0.006
Maximum error	0.039	0.026

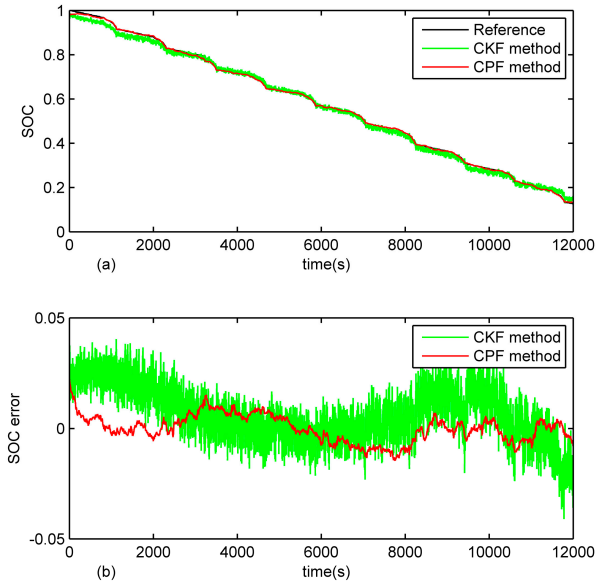


Figure 12. 0.03 V voltage random noise and 0.1 A current random noise experiment results.

Table 4. Performance of the CKF and CPF algorithms under NEDC conditions with 0.03 V voltage random noise and 0.1 A current random noise.

Estimation Method	CKF	CPF
Mean absolute error	0.011	0.005
Maximum error	0.048	0.029

5. Conclusions

The CPF algorithm was proposed for the *SOC* estimation of lithium-ion batteries in electric vehicles. In this method, preprocessing of the particles was performed using the CKF method and the suggested density function was generated using the PF algorithm. This CPF approach improved the estimation accuracy and robustness of the CKF algorithm. The second-order equivalent circuit was selected as a model to approximate the experimental characteristics of the battery, achieving the trade-off between computational complexity and precision. The experimental results were fitted by polynomials to test the parameters $\{U_{oc}(SOC)$ and R_0, R_1, R_2, C_1, $C_2\}$ of the second-order RC equivalent circuit model. We utilized the DST condition to verify the accuracy of the algorithm, added noise to the NEDC condition to verify the robustness of the algorithm, and compared the CPF method with the CKF method. We found that the CPF algorithm exhibited better accuracy and robustness than the original CKF method. Therefore, the CPF algorithm is more suitable for *SOC* estimation.

Acknowledgments: This work was supported by the Shenzhen Science and Technology Project (No. JCY20150331151358137).

Author Contributions: Bizhong Xia and Zhen Sun conceived and designed the experiments; Ruifeng Zhang performed the experiments; Ruifeng Zhang and Zizhou Lao analyzed the data; Bizhong Xia contributed reagents/materials/analysis tools; and Zhen Sun wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zheng, Y.; Ouyang, M.; Li, X.; Lu, L.; Li, J.; Zhou, L.; Zhang, Z. Recording frequency optimization for massive battery data storage in battery management systems. *Appl. Energy* **2016**, *183*, 380–389. [CrossRef]
2. Basu, S.; Hariharan, K.S.; Kolake, S.M.; Song, T.; Sohn, D.K.; Yeo, T. Coupled electrochemical thermal modelling of a novel Li-ion battery pack thermal management system. *Appl. Energy* **2016**, *181*, 1–13. [CrossRef]
3. Ng, K.S.; Moo, C.-S.; Chen, Y.-P.; Hsieh, Y.-C. Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries. *Appl. Energy* **2009**, *86*, 1506–1511. [CrossRef]
4. Zhang, H.; Zhao, L.; Chen, Y. A lossy counting-based state of charge estimation method and its application to electric vehicles. *Energies* **2015**, *8*, 13811–13828. [CrossRef]
5. Feng, F.; Lu, R.G.; Zhu, C.B. A combined state of charge estimation method for lithium-ion batteries used in a wide ambient temperature range. *Energies* **2014**, *7*, 3004–3032. [CrossRef]
6. 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]
7. 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]
8. Barbarisi, O.; Vasca, F.; Glielmo, L. State of charge Kalman filter estimator for automotive batteries. *Control Eng. Pract.* **2006**, *14*, 267–275. [CrossRef]
9. Sepasi, S.; Roose, L.R.; Matsuura, M.M. Extended Kalman filter with a fuzzy method for accurate battery pack state of charge estimation. *Energies* **2015**, *8*, 5217–5233. [CrossRef]
10. Tulsyan, A.; Tsai, Y.; Gopaluni, R.B.; Braatz, R.D. State-of-charge estimation in lithium-ion batteries: A particle filter approach. *J. Power Sources* **2016**, *331*, 208–223. [CrossRef]

11. Dai, H.; Zhu, L.; Zhu, J.; Wei, X.; Sun, Z. Adaptive Kalman filtering based internal temperature estimation with an equivalent electrical network thermal model for hard-cased batteries. *J. Power Sources* **2015**, *293*, 351–365. [CrossRef]
12. Tian, Y.; Xia, B.; Sun, W.; Xu, Z.; Zheng, W. A modified model based state of charge estimation of power lithium-ion batteries using unscented Kalman filter. *J. Power Sources* **2014**, *270*, 619–626. [CrossRef]
13. Shi, P.; Zhao, Y.W.; Shi, P. Application of unscented Kalman filter in the *SOC* estimation of Li-ion battery for Autonomous Mobile Robot. In Proceedings of the 2006 IEEE International Conference on Information Acquisition, Weihai, China, 20–23 August 2006; pp. 1279–1283.
14. Fei, Z.; Guangjun, L.; Lijin, F. Battery state estimation using unscented Kalman filter. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation (ICRA), Kobe, Japan, 12–17 May 2009; pp. 1863–1868.
15. Zhiwei, H.; Mingyu, G.; Caisheng, W.; Leyi, W.; Yuanyuan, L. Adaptive state of charge estimation for Li-ion batteries based on an unscented kalman filter with an enhanced battery model. *Energies* **2013**, *6*, 4134–4151.
16. Xia, B.Z.; Wang, H.Q.; Wang, M.W.; Sun, W.; Xu, Z.H.; Lai, Y.Z. A new method for state of charge estimation of lithium-ion battery based on strong tracking cubature Kalman filter. *Energies* **2015**, *8*, 13458–13472. [CrossRef]
17. Shen, Y.Q. Hybrid unscented particle filter based state-of-charge determination for lead-acid batteries. *Energy* **2014**, *74*, 795–803. [CrossRef]
18. Rumelhart, D.E.; Mcclelland, J.L.; Group, T.P. Parallel distributed processing: Explorations in the microstructure of cognition foundations. *Language* **1986**, *63*, 12–21.
19. Kalman, R.E. , A new approach to linear filtering and prediction problems. *J. Fluids Eng.* **1960**, *82*, 35–45. [CrossRef]
20. Xiong, B.; Zhao, J.; Wei, Z.; Skyllas-Kazacos, M. Extended Kalman filter method for state of charge estimation of vanadium redox flow battery using thermal-dependent electrical model. *J. Power Sources* **2014**, *262*, 50–61. [CrossRef]
21. Wang, S.; Fernandez, C.; Shang, L.; Li, Z.; Li, J. Online state of charge estimation for the aerial lithium-ion battery packs based on the improved extended Kalman filter method. *J. Energy Storage* **2017**, *9*, 69–83. [CrossRef]
22. Wan, E.A.; van der Merwe, R. The unscented Kalman filter for nonlinear estimation. In Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373), Lake Louise, AB, Canada, 4 October 2000; pp. 153–158.
23. Mazaheri, A.; Radan, A. Performance evaluation of nonlinear Kalman filtering techniques in low speed brushless DC motors driven sensor-less positioning systems. *Control Eng. Pract.* **2017**, *60*, 148–156. [CrossRef]
24. Ye, M.; Guo, H.; Cao, B. A model-based adaptive state of charge estimator for a lithium-ion battery using an improved adaptive particle filter. *Appl. Energy* **2017**, *190*, 740–748. [CrossRef]
25. Wang, Y.; Zhang, C.; Chen, Z. A method for state-of-charge estimation of LiFePO4 batteries at dynamic currents and temperatures using particle filter. *J. Power Sources* **2015**, *279*, 306–311. [CrossRef]
26. Xia, B.Z.; Wang, H.Q.; Tian, Y.; Wang, M.W.; Sun, W.; Xu, Z.H. State of charge estimation of lithium-ion batteries using an adaptive cubature Kalman filter. *Energies* **2015**, *8*, 5916–5936. [CrossRef]
27. Sun, T.; Xin, M. Hypersonic entry vehicle state estimation using nonlinearity-based adaptive cubature Kalman filters. *Acta Astronaut.* **2017**, *134*, 221–230. [CrossRef]
28. Zhang, Y.; Huang, Y.; Li, N.; Zhao, L. Embedded cubature Kalman filter with adaptive setting of free parameter. *Signal Process.* **2015**, *114*, 112–116. [CrossRef]
29. Zheng, L.; Zhang, L.; Zhu, J.; Wang, G.; Jiang, J. Co-estimation of state-of-charge, capacity and resistance for lithium-ion batteries based on a high-fidelity electrochemical model. *Appl. Energy* **2016**, *180*, 424–434. [CrossRef]
30. Li, J.; Wang, L.; Lyu, C.; Wang, H.; Liu, X. New method for parameter estimation of an electrochemical-thermal coupling model for LiCoO2 battery. *J. Power Sources* **2016**, *307*, 220–230. [CrossRef]
31. Jiang, J.; Ruan, H.; Sun, B.; Zhang, W.; Gao, W.; Wang, L.Y.; Zhang, L. A reduced low-temperature electro-thermal coupled model for lithium-ion batteries. *Appl. Energy* **2016**, *177*, 804–816. [CrossRef]
32. 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]
33. Johnson, V.H. Battery performance models in ADVISOR. *J. Power Sources* **2002**, *110*, 321–329. [CrossRef]

34. Chen, X.; Shen, W.; Cao, Z.; Kapoor, A. A novel approach for state of charge estimation based on adaptive switching gain sliding mode observer in electric vehicles. *J. Power Sources* **2014**, *246*, 667–678. [CrossRef]
35. Sepasi, S.; Ghorbani, R.; Liaw, B.Y. A novel on-board state-of-charge estimation method for aged Li-ion batteries based on model adaptive extended Kalman filter. *J. Power Sources* **2014**, *245*, 337–344. [CrossRef]
36. Lee, J.; Nam, O.; Cho, B.H. Li-ion battery *SOC* estimation method based on the reduced order extended Kalman filtering. *J. Power Sources* **2007**, *174*, 9–15. [CrossRef]
37. Jossen, A.; Späth, V.; Döring, H.; Garche, J. Reliable battery operation—A challenge for the battery management system. *J. Power Sources* **1999**, *84*, 283–286. [CrossRef]
38. Doucet, A.; Godsill, S.; Andrieu, C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput.* **2000**, *10*, 197–208. [CrossRef]
39. Chen, Q.Y.; Jiang, J.C.; Ruan, H.J.; Zhang, C.P. A New double sliding mode observer for EV lithium battery *SOC* estimation. *Math. Probl. Eng.* **2016**, *2016*, 8048905. [CrossRef]