HAO, X., WANG, S., FAN, Y., XIE, Y. and FERNANDEZ, C. 2023. An improved forgetting factor recursive least square and unscented particle filtering algorithm for accurate lithium-ion battery state of charge estimation. *Journal of energy storage* [online], 59, article 106478. Available from: https://doi.org/10.1016/j.est.2022.106478

An improved forgetting factor recursive least square and unscented particle filtering algorithm for accurate lithium-ion battery state of charge estimation.

HAO, X., WANG, S., FAN, Y., XIE, Y. and FERNANDEZ, C.

2023

An improved forgetting factor recursive least square and unscented particle filtering algorithm for accurate lithium-ion battery state of charge estimation

Xueyi Hao[a], Shunli Wang[a*b], Yongcun Fan[a], Yanxin Xie[a], Carlos Fernandez[c]

[a]*School of Information Engineering, Southwest University of Science and Technology, Mianyang 621010, China;*

[b]*Colledge of Electrical Engineering, Sichuan University, Chengdu, 610065, China;*

[c]*School of Pharmacy and Life Sciences, Robert Gordon University, Aberdeen AB10-7GJ, UK.*

Abstract: As an indispensable part of the battery management system, accurately predicting the estimation of the state of charge (SOC) has attracted more attention, which can improve the efficiency of battery use and ensure its safety performance. Taking the ternary lithium battery as the research object, we present an improved forgetting factor recursive least square (IFFRLS) method for parameter identification and a joint unscented particle filter algorithm for SOC estimation. First, take advantage of the particle swarm optimization (PSO) algorithm to select the optimal parameter initial value and forgetting factor value to improve the precision of the FFRLS method. At the same time, make use of the unscented Kalman algorithm (UKF) as the density function of the particle filter algorithm (PF) to form the unscented particle filtering (UPF) algorithm. Then, the IFFRLS method and UPF algorithm are proposed in this paper. The different working conditions results show that the proposed algorithm estimates the SOC with good convergence and high system robustness. The final estimation error of the algorithm is stable at 1.6%, which is lower than the errors of the currently used EKF algorithm, UKF algorithm and PF algorithm, which provides a reference for future research on lithium-ion batteries.

Keywords: lithium-ion batteries; state of charge estimation; Particle Swarm Optimization; forgetting factor least squares; unscented particle filter.

***Corresponding author**: Shun-Li Wang. E-mail address: 497420789@qq.com.

Nomenclature

Nomenclature		R_{p1}/R_{p2}	polarization internal resistance
C_{p1}/C_{p2}	polarization capacitance	Q_n	rated power of the battery
R_0	ohmic internal resistance	U_{OC}	open-circuit voltage

K	the gain	λ	genetic factor
$\Phi(k)$	observed vector	$\theta(k)$	parameter vector
$e(k)$	observation noise vector	$J(\theta)$	objective function
$\hat{\theta}(k)$	the final simulation effect	$\Phi^T(k+1)\hat{\theta}(k)$	the calculated value of the system observation at time $k+1$
$y(k+1)$	the actual observation value at time $k+1$	$p(X_0)$	the prior distribution
$X_{k-1}^{i,a}$	a set of Sigma points	W_k^i	the weight
SOC	state of charge	IFFRLS	improved forgetting factor recursive least-squares
FFRLS	forgetting factor recursive least-squares	UKF	unscented Kalman filter
PF	particle filter	EKF	extended Kalman filter
BMS	battery management system	RC	resistance-capacitance
PSO	particle swarm optimization	GA	genetic algorithm
MAE	mean absolute error	RMSE	root mean square error
PDF	probability density function	HPPC	hybrid pulse power characteristic
BBDST	Beijing Bus Dynamic Stress Test	UPF	unscented particle filter
AEKF	adaptive extended Kalman filter	DEKF	double expansion Kalman filter

1. Introduction

Nowadays, the lithium-ion battery has been widely used in all aspects of production and life with their excellent performance [1]. With its high energy density, high electric potential, and long life compared with other batteries, lithium-ion batteries are widely used in consumer electronics such as cell phones, notebook computers, electric vehicles, and aerospace electronics [2, 3]. The detection of the state of charge for lithium-ion batteries has received more and more attention [4]. Battery management has been intensively studied by a broad range of researchers, such as Zhang et al [5]. Among them, accurate estimation of the state of charge of lithium-ion batteries plays a very significant role in allowing full play to battery property and implementing efficient utilization of lithium-ion batteries.

The State of Charge of the battery is one of the core parameters of the battery management system (BMS) [6]. The accuracy of the SOC will directly affect the cycle life of the battery and the operating performance of the BMS [7, 8]. Under certain discharge conditions, the remaining capacity to the rated capacity is defined as the SOC value of the battery [9, 10]. The SOC value is a relative quantity, expressed as a percentage, and the value range of SOC is 0~100% [11].

At present, for lithium-ion batteries, there are many methods to estimate the state of charge, such as the ampere-hour integration method, open-circuit voltage method, discharge experiment method, neural network method, Kalman filter algorithm, and particle filter algorithm [12-14], among them, the most commonly used is the ampere-hour integration method [15-17]. The ampere-hour integration method will also cause the gradual accumulation of errors [18, 19]. The open-circuit voltage method requires that the lithium-ion battery must be left standing for a long time when estimating the SOC. The discharge experiment method is the easiest and most accurate among the traditional SOC prediction and estimation methods, but its efficiency is not high. The neural network algorithm is difficult to establish a relatively accurate mathematical model for the whole process. The neural network method does not require an accurate mathematical model and can learn the internal laws of the nonlinear system by learning the sample data, and a good neural network model can approximate the nonlinear mapping with arbitrary precision. However, in the application of the standard BP neural network algorithm, it is easy to forget old samples in the process of training, and it is easy to fall into local minima. The network convergence speed is slow, and the number of hidden layer nodes is mostly based on empirical formulas, lacking professional theory inadequate guidance, etc. Real-time requirements cannot be met with this method, so it is usually not used alone [20]. Filtering is a problem in system state estimation. Since Mr. Kalmal proposed the classical Kalman filter in 1960, it has provided an optimal solution to the linear problem [21, 22]. Up to now, it is still widely used. However, in the real world, most of the practical puzzles in the field of science have nonlinear characteristics, and the Kalman filter is powerless to solve these nonlinear problems [23]. With time, the extended Kalman filter becomes a powerful tool to solve nonlinear filtering [24, 25]. Then, there is the appearance of an unscented Kalman filter [26-28]. For any nonlinear system, the unscented Kalman filter can obtain the posterior mean and covariance estimates exactly to the third order [29]. However, the UKF algorithm assumes that the statistical properties of the system noise obey a Gaussian distribution when estimating the battery SOC, which leads to a reduction in accuracy and loss. Then came the particle filter [30-32], the particle filtering algorithm is not limited by the noise distribution. A particle filter can deal with any nonlinear model and any noise distribution [33, 34]. However, the particle filter itself also has many problems. For example, particle filtering algorithms suffer from particle degradation, and although resampling can reduce this phenomenon to some extent, it greatly increases the computational effort. Based on it, this paper combines the unscented Kalman filter with the particle filter to obtain an unscented particle filter algorithm. It can effectively ensure that the number of particles does not decrease significantly and improve the accuracy of the estimation results. The second-order RC equivalent model is employed for online parameter identification, while the optimal parameters are selected in real time using the PSO algorithm, so the accuracy for the state of charge estimation of lithium-ion batteries is improved and the particle filter algorithm is improved [35].

Currently, the Thevenin model is commonly used in lithium-ion battery SOC estimation [36]. The second-order resistance-capacitance (RC) model is an improvement of the Thevenin model. In comparison, the second-order RC model can describe the operating characteristics of the battery more accurately [37-39]. At the same time, the accuracy of online parameter identification is often higher than offline parameter identification [40, 41]. Therefore, this paper uses the PSO algorithm to optimize the FFRLS method to perform the improved Thevenin model [42-45]. To improve the estimation accuracy of the Kalman filter algorithm, unscented Kalman Mann filter algorithm, and particle filter algorithm [46-49]. To further enhance the estimation accuracy, this paper combines the unscented Kalman filter and the particle filter to form an unscented particle filter algorithm.

This paper aims to take the ternary lithium battery as the research object, study the state of charge estimation of the lithium-ion battery, optimize the FFRLS algorithm through the PSO algorithm, form the IFFRLS algorithm, and use the combined algorithm of the algorithm and the UPF algorithm to estimate the charge of the lithium-ion battery. This paper confirms the performances of the algorithm under different working conditions, which lays the foundation for the research on the SOC of lithium-ion batteries in the future.

Next, the main content of this paper will be elaborated on one by one. It mainly introduces the selection of the equivalent model of lithium battery, the forgetting factor recursive least square algorithm, particle swarm optimization algorithm, the improved forgetting factor recursive least square algorithm, and the unscented particle filter algorithm in chapter 2. Then, it illustrates the detailed results of parameter identification and SOC estimation under different complex working conditions step by step in chapter 3. The results display that the unscented example filtering algorithm under the online parameter identification method with the forgetting factor has better real-time performance and accuracy, and realizes the closed-loop online estimation. At last, it describes the conclusions of the paper. In addition, the proposed algorithm is compared with existing algorithms, such as the EKF algorithm, UKF algorithm, PF algorithm, AEKF algorithm, and DEKF algorithm. By comparing the estimation curves of several algorithms and the error curves with the century results, we can conclude that the proposed algorithm has better estimation than several other algorithms. The specific data results will be presented in detail below. Finally, the robustness of the algorithm is confirmed by simulation.

2. Mathematical analysis

2.1 The second-order equivalent model

The battery equivalent models commonly used today are the Thevenin model, second-order RC equivalent model, Rint model, PNGV model, etc. The second-order RC model is composed of a static ohmic resistance R_0 and two RC loops that characterize the dynamic response in series. The simple model can't describe the operating characteristics of the battery, although its calculation is simple. On the contrary, for complex models, it reduces the adaptation of the model, to the complex calculations. However, the complex model can better characterize the charge and discharge characteristics of the battery. To sum up, the second-order RC model is selected to estimate the SOC. The architecture of the second-order equivalent circuit is displayed in Figure 1.

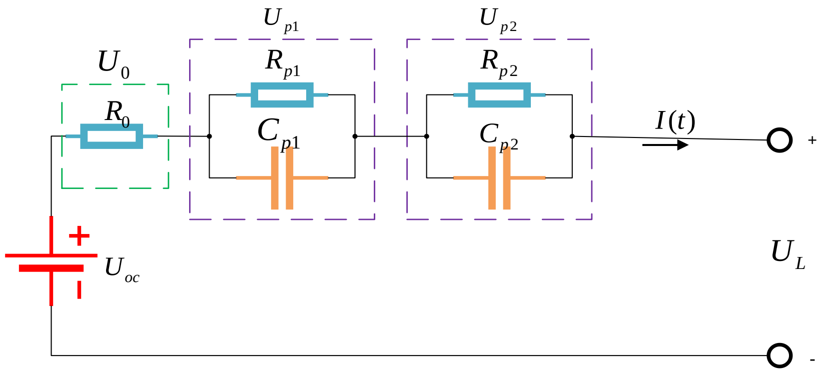


Figure 1 The second-order equivalent circuit model

As shown in Figure 1, the RC circuit is composed of R_{p1} and C_{p1}, this circuit can accurately express the stage of rapid voltage change during the internal chemical reaction of the battery. The RC loop is composed of R_{P2} and C_{P2}, and this loop represents the phase where the voltage changes slowly during the chemical reaction within the battery. Compared with the effects of equivalent circuit models of different orders on SOC estimation. It does not significantly increase the accuracy of models above the second order, but highly improves the computation. Therefore, the Thevenin model, PNGV model, or the second-order equivalent circuit model are used to estimate SOC according to the actual situation. According to Kirchhoff's circuit law, Equation (1) is listed in combination with Figure 1.

$$\begin{cases} U_L = U_{oc}(SOC) - i(t)R_0 - U_{p1} - U_{p2} \\ \dfrac{dU_{p1}}{dt} = -\dfrac{U_{p1}}{R_{p1}C_{p1}} + \dfrac{i}{C_{p1}} \\ \dfrac{dU_{p2}}{dt} = -\dfrac{U_{p2}}{R_{p2}C_{p2}} + \dfrac{i}{C_{p2}} \end{cases} \tag{1}$$

For the second-order equivalent model selected, $[SOC\ U_{P1}\ U_{P2}]$ is chosen as the state variable. Integrated with Equation (1) and the circumscription of SOC, its state space equation can be listed as demonstrated in Equation (2).

$$\begin{cases} \begin{bmatrix} SOC_{k+1} \\ U_{p1,k+1} \\ U_{p2,k+2} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1-\dfrac{T}{\tau_1} & 0 \\ 0 & 0 & 1-\dfrac{T}{\tau_2} \end{bmatrix} \\ U_{L,k+1} = U_{oc}(SOC,k+1) - U_{p1} - U_{p2} - IR_0 \end{cases} \tag{2}$$

In the above formula, Q_n is the rated power of the battery, and $R_{P1}C_{P1}$ and $R_{p2}C_{P2}$ are the cutoff angular frequencies. Parameters identified by the model include ohmic internal resistance R_0, open-circuit voltage U_{OC}, polarization internal resistance R_{P1} and R_{P2}, and polarization capacitance C_{P1} and C_{P2}.

2.2 Improved forgetting factor least squares algorithm

2.2.1 Forgetting Factor Recursive Least Squares

In this paper, we choose the online identification method, and firstly, we use the hybrid pulse power characteristic (HPPC) test experiment to test the lithium battery performance at the ambient temperature of 25 □. The rated capacity of the battery is 70 Ah, and the actual capacity of the battery is 68.74 Ah after three complete charge and discharge tests [50]. Obtain the voltage, current, and other data required for model parameter identification, analyze the working process of the battery under specific temperature conditions, and obtain the required parameters. Figure 2 and Figure 3 are the voltage and current change curves of the charge-discharge cycle of the HPPC test.

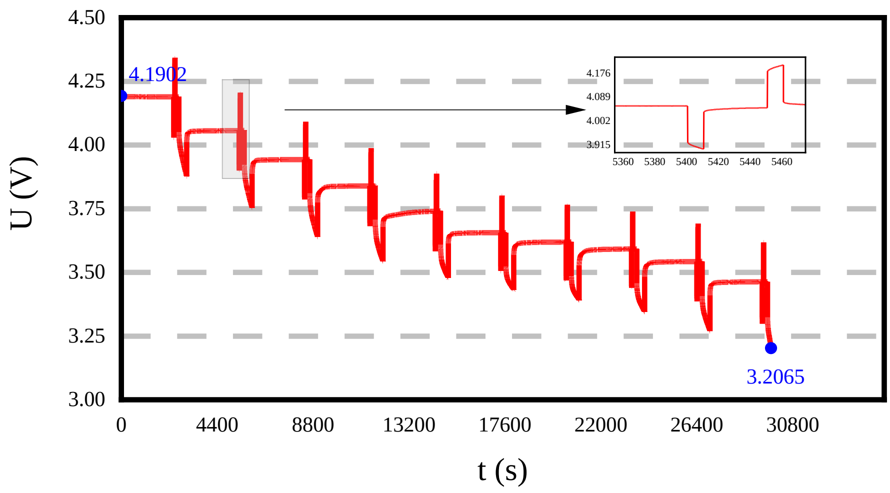


Figure 2 The voltage curve of the HPPC experiment

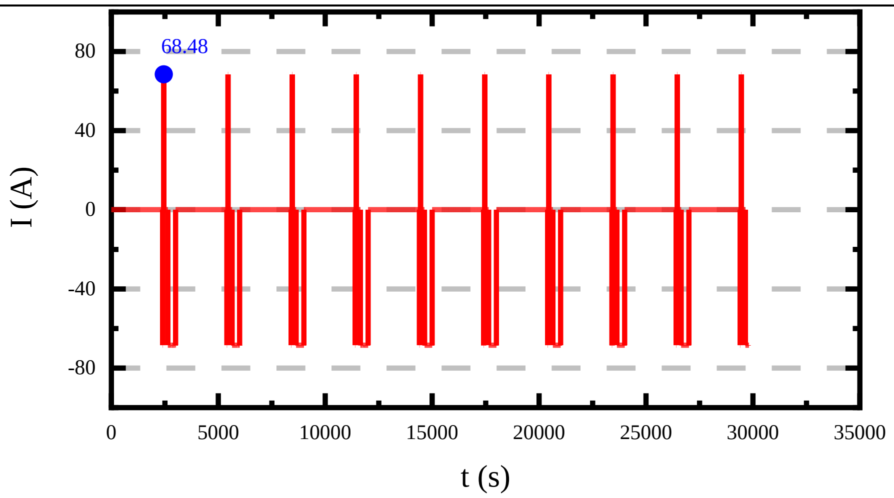


Figure 3 The current curve of the HPPC experiment

From Figure 2 and Figure 3, we can see that the battery end voltage declines or goes up abruptly after the battery is connected to the load and discharges or ends discharge, with the internal resistance effect of the lithium battery. The polarization effect of lithium batteries will make the terminal voltage drop rapidly after the first time or rise after the discharge. The polarization effect disappears when the battery is fully rested, then the interior of the battery reaches equilibrium.

The experimental environment can easily affect the reaction process of complex chemical reactions in the battery, which are generated during the use of lithium-ion batteries. In battery SOC estimation, taking advantage of the method of fitting a function to offline experimental data determines the value of each parameter in the battery equivalent model. However, the estimation result, with absolute errors, will occur when the method is used. Identifying the parameters of the model online and correcting the values of the parameters in real-time is the most important thing to improve the accuracy of SOC estimation. Based on the commonly used second-order RC equivalent model, this paper identifies the parameters of the model with the improving forgetting factor recursive least square method. The principle process diagram is illustrated in Figure 4.

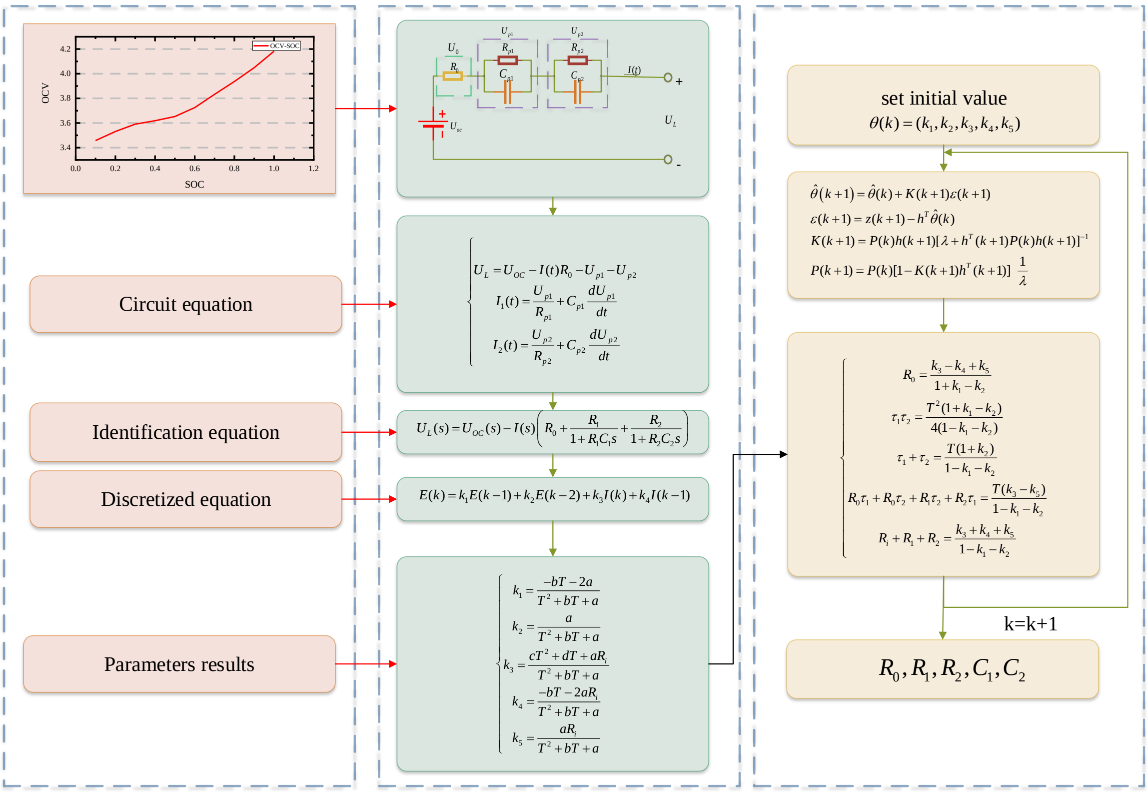


Figure 4 Flowchart of the FFRLS method

From Figure 4, we can understand the general process of the FFRLS method and its overview. Next, we will introduce the specific procedure of the FFRLS method. From equation (3), we can obtain the output equation of the circuit with the principle of the second-order RC equivalent model.

$$U_{oc} = \left(\frac{R_{p1}}{R_{p1}C_{p1}s + 1} + \frac{R_{p2}}{R_{p2}C_{p2}s + 1} + R_0\right) I + U \tag{3}$$

Due to the "filter saturation" phenomenon of the least squares method, the values of the gains K and P become smaller and smaller as the number of iterations of the algorithm increases. This makes the algorithm's ability to correct the data weaker and weaker, and the degree of data saturation becomes larger and larger, which eventually leads to more and more errors in parameter identification. In the final analysis, it is because the correction ability of the data will become weaker and weaker with this algorithm, and the saturation of the data will also become larger and larger. Therefore, to improve the accuracy of the parameter identification results, we choose the least squares method with the addition of forgetting factors for parameter identification. In the identified process, the function of the forgetting factor is to give a smaller weight to the data with a longer running time and the latest observation data more weight. After the introduction of genetic factor λ $(0 < \lambda < 1)$, the impact of previous old data will be weakened, while the feedback effect of new data will be enhanced. Equation (4) shows the mathematical expression of the least square method.

$$y(k) = \Phi(k)\theta^T + e(k) \tag{4}$$

Among them, the observed vector is denoted by $\Phi(k)$; the final parameter vector to be estimated is represented by $\theta(k)$; the observation noise vector is represented by $e(k)$.

Take the objective function $J(\theta)$. Finding $\hat{\theta}$ is the objective of the least square method. The premise that $J(\theta)$ takes the minimum value is that $\hat{\theta}$ exists. The objective function and estimated parameter values of the system are written in equation (5).

$$\begin{cases} J(\hat{\theta}) = \left[y(k) - \Phi(k)\hat{\theta}(k)\right]^T \left[y(k) - \Phi(k)\hat{\theta}(k)\right] \\ \hat{\theta} = [\Phi(k)\Phi(k)^T]^{-1}\Phi(k)y(k) \end{cases} \tag{5}$$

In the actual simulation calculation, before reaching the approved accuracy, to gradually improve the accuracy of parameter estimation, as the indispensable part, the latest experimental data must be continuously imported and exported, which is achieved in a continuous iterative process. After introducing the forgetting factor λ $(0 < \lambda < 1)$, the specific calculation process is expressed in equation (6).

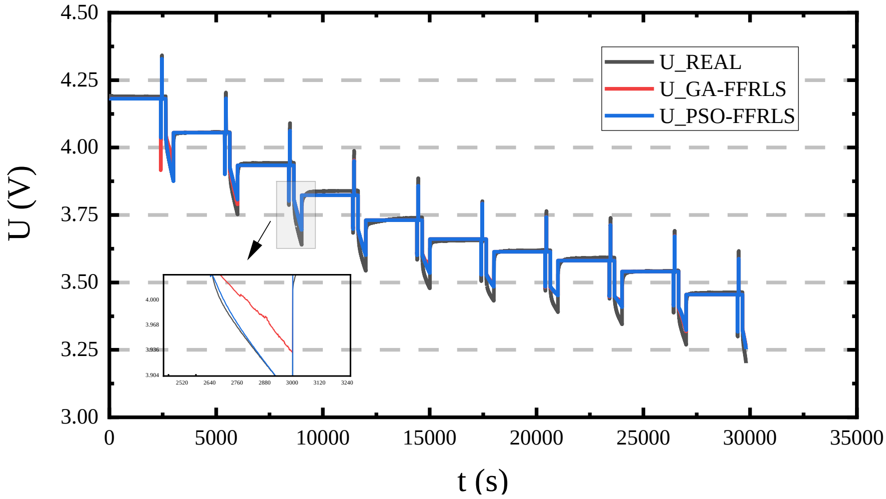
$$\begin{cases} \hat{\theta}(k+1) = \hat{\theta}(k) + K(k+1)\left[y(k+1) - \Phi^T(k+1)\hat{\theta}(k)\right] \\ K(k+1) = P(k+1)\Phi(k+1)[\Phi^T(k+1)P(k)\Phi(k+1) + \lambda]^{-1} \\ P(k+1) = \lambda^{-1}[I - K(k+1)\Phi^T(k+1)]P(k) \end{cases} \tag{6}$$

In the above equation, the closer the value λ is to 1, the better the final simulation effect is $\hat{\theta}(k)$. is the estimated value of the parameter at time k, $\Phi^T(k+1)\hat{\theta}(k)$ is the calculated value of the system observation at time $k+1$, and $y(k+1)$ is the actual observation value at time $k+1$. At every iteration, the algorithm uses the deviation between the calculated and actual observations of the system and the gain K to amend the ultimate estimate.

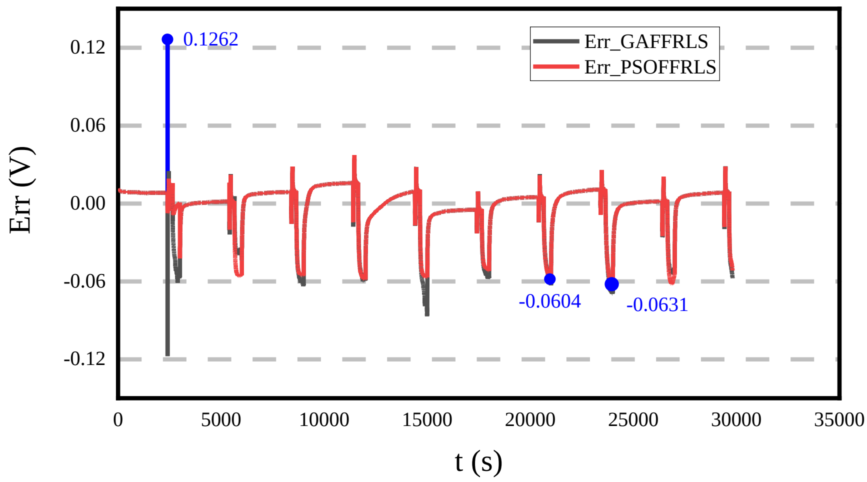
2.2.2 Particle Swarm Optimization

The genetic algorithm (GA) and PSO algorithm are both intelligent algorithms. Different from the GA algorithm, the GA algorithm mainly draws on the law of "survival of the fittest" in biological evolution. The PSO algorithm is proposed based on simulating social behaviors such as birds foraging and human cognition.

There are various types of intelligent algorithms. The usual intelligent algorithms in current research are genetic algorithm, particle swarm optimization algorithm, ant colony algorithm, simulated annealing algorithm, fish swarm algorithm, etc. Genetic algorithm has strong global search ability and weak local search ability, and often can only get the suboptimal solution but not optimal solution. The parameter setting of the ant colony algorithm is complicated. If the parameter setting is improper, it is easy to deviate from the high-quality solution. The simulated annealing algorithm is a global optimization, which is mainly suitable for use with algorithms such as particle swarms and whale optimization algorithms that are prone to fall into local optimal solutions. The fish swarm algorithm is similar to the ant colony algorithm. If the parameters are not set advisable, it is easy to deviate from the high-quality solution. If the particle swarm optimization algorithm is not weighted, it is easy to fall into the local optimal solution, so the weight value will be weighted when the particle swarm optimization algorithm is generally selected.



(a) Voltage contrast curves of FFRLS algorithm optimized by GA algorithm and FFRLS algorithm optimized by PSO algorithm



(b) Error comparison curve of FFRLS algorithm optimized by GA algorithm and FFRLS algorithm optimized by PSO algorithm

Figure 5 Comparison of FFRLS optimized by GA algorithm and FFRLS optimized by PSO algorithm

From Figure 5, we can see the difference between using the GA algorithm as the optimization algorithm and using the PSO algorithm as the optimization algorithm. Due to the weak local search ability of the GA algorithm, the result is prone to be not the optimal solution, and from the change of the graph, we can see that the PSO algorithm that affects the overall optimization is better than the GA algorithm, and we can calculate the performance indexes of the two algorithms again to make a better choice. The performance comparison of the two algorithms is shown in **Error! Reference source not found.**.

Table 1 Performance comparison of FFRLS optimized by GA algorithm and FFRLS optimized by PSO algorithm

Algorithm	Max	MAE	RMSE

GAFFRLS	0.1262	0.01181	0.01922
PSOFFRLS	0.0631	0.01126	0.01822

Error! Reference source not found. lists the performance indexes of the two algorithms. It can be seen that the overall performance metrics of the PSO algorithm are lower than those of the GA algorithm, and due to the nature of MAE and RMSE, we can see that the PSO algorithm is better than the GA algorithm for the optimization of the FFRLS algorithm.

The particle swarm optimization algorithm originated from research on the predation behavior of birds. Its core idea is to use the information sharing of individuals in the group to move the whole group to produce an evolution process from disorder to order in the problem-solving space, to obtain the optimal solution to the problem. The flowchart of the algorithm is reflected in Figure 6.

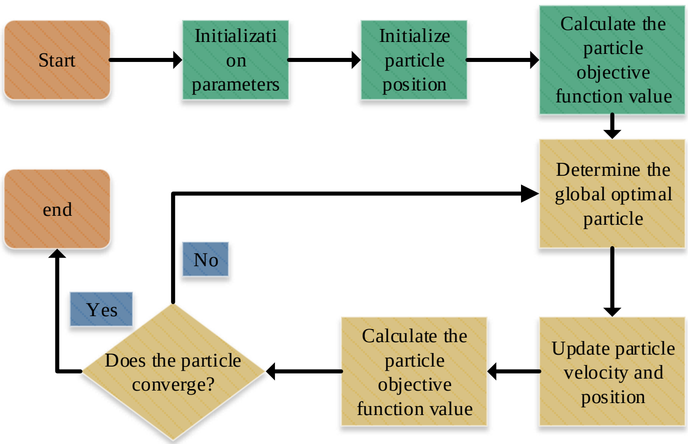


Figure 6 The flowchart of the PSO algorithm

From Figure 6, We can clearly understand the general process of the PSO algorithm, so that we can better understand the PSO algorithm. When using the PSO algorithm alone to identify the lithium-ion offline parameters, we select the root mean square error between the actual voltage and the model voltage as the objective function and obtain the parameter values of the model. In the process of lithium-ion parameter identification, we should set the number of particles, the number of parameters to be identified, and the number of iterations. It should be noted that when using the PSO algorithm for offline parameter identification of the second-order model, as the parameters to be identified increase, the upper and lower boundaries of the battery parameters need to be valued. At this time, we should reasonably set the upper and lower borders of the battery, and the identified parameter results are not applicable.

2.2.3 Improved Forgetting Factor Recursive Least Square

The least-squares method with the forgetting factor can effectively improve the "data saturation" problem of the time-varying system of the equivalent model. However, as the essence of the algorithm, how to select the optimal initial parameter value and forgetting factor is a problem that plagues us. In this paper, to solve this problem, the particle swarm optimization algorithm is employed. Compared with the improved algorithm of PSO, the PSO algorithm is less computationally intensive and has a good optimization effect at the same time. Therefore, in this paper, we choose to use the PSO algorithm as the optimization algorithm to optimize the FFRLS algorithm. The particle swarm optimization algorithm is adopted, and the objective function is set, with the terminal voltage error. On this basis, to improve the estimation accuracy of the lithium battery state of charge, the optimal initial parameter value and forgetting factor value are screened in real-time. The following Table 2 introduces the specific steps of using the particle swarm optimization algorithm to optimize the forgetting factor least squares algorithm.

Table 2 Process of improving forgetting factor least squares

Step 1, to start the loop, set k=3.
Step 2, input current and voltage data, that is, data vector, $\boldsymbol{\psi}(\boldsymbol{k}) = \left(\boldsymbol{U}(\boldsymbol{k}-\mathbf{1}), \boldsymbol{U}(\boldsymbol{k}-\mathbf{2}), -\boldsymbol{I}(\boldsymbol{k}), -\boldsymbol{I}(\boldsymbol{k}-\mathbf{1}), -\boldsymbol{I}(\boldsymbol{k}-\mathbf{2})\right)$.
Step 3, initialization parameter population: parameter initial vector $\boldsymbol{\theta}_\mathbf{1}(\mathbf{k}-\mathbf{1}), \boldsymbol{\theta}_\mathbf{2}(\mathbf{k}-\mathbf{1}), \cdots, \boldsymbol{\theta}_\mathbf{t}(\mathbf{k}-\mathbf{1})$ and forgetting factor $\boldsymbol{\lambda}_\mathbf{1}, \boldsymbol{\lambda}_\mathbf{2}, \cdots, \boldsymbol{\lambda}_{\boldsymbol{t}}$.
Step 4, select the absolute value of the terminal voltage error as the fitness function, $\boldsymbol{J} = \lvert\boldsymbol{U}(\boldsymbol{k}) - \boldsymbol{OCV}(\boldsymbol{k}) - \boldsymbol{\theta}(\boldsymbol{k}-\mathbf{1})^{\boldsymbol{T}}\boldsymbol{\psi}(\boldsymbol{k})\rvert$.
Step 5, the particle swarm iteratively selects the optimal initial parameter value and forgetting factor value at time $\mathbf{k}$.
Step 6, calculate the least squares gain matrix K, calculate the least squares covariance matrix $\boldsymbol{P}$.
Step 7, update the parameter vector.
Step 8, find the model parameters: $\boldsymbol{R}_\mathbf{0}(\boldsymbol{k}), \boldsymbol{R}_{\boldsymbol{P}\mathbf{1}}(\boldsymbol{k}), \boldsymbol{R}_{\boldsymbol{P}\mathbf{2}}(\boldsymbol{k}), \boldsymbol{C}_{\boldsymbol{P}\mathbf{1}}(\boldsymbol{k}), \boldsymbol{C}_{\boldsymbol{P}\mathbf{2}}(\boldsymbol{k})$.

The process of optimizing the FFRLS algorithm by the PSO algorithm in detail is shown in Table 2, from which we can clearly understand the calculation process of the IFFRLS algorithm after the combination of the two algorithms.

In addition, in terms of the complexity of the algorithm, the time complexity and space complexity of the proposed algorithm is higher than our most commonly used EKF algorithm, but compared with some other extension algorithms and optimization algorithms, the proposed algorithm The complexity of the algorithm is at a medium level. Considering the accuracy and stability of the estimation results, the proposed algorithm still has certain advantages.

2.3 Unscented Particle Filter Algorithm

The particle filter method implements recurrent Bayesian filtering through non-parametric Monte Carlo simulation methods. It is suitable for any nonlinear system, which can be depicted by a state-space model. Due to its non-parametric characteristics, it gets rid of the restriction that the random quantity must meet the Gaussian distribution when solving the nonlinear filtering problem. Compared with the Gaussian model, the particle filter is more widely used and has a better modeling ability. The central idea is to use some dispersed stochastic sampling points, with the posterior probability density function (PDF) of the state being approached. Then utilizing the sample mean replaces the integral calculation. At last, we can get the minimum variance estimate of the final condition. The unscented particle filtering (UPF) algorithm gains a PDF, with the latest observations, with the unscented Kalman filtering (UKF) algorithm to generate the recommended distribution.

Taking advantage of the unscented transformation algorithm optimizes the particle filtering (PF) algorithm in the UPF algorithm. The UKF algorithm can theoretically calculate the accuracy of the third-order square difference, which can be obtained from the comparison of the EKF algorithm based on the expansion of the first-order Taylor. The algorithm has higher precision and is also a valid calculation. Figure 7 presents the overall framework of the IFFRLS-UPF algorithm.

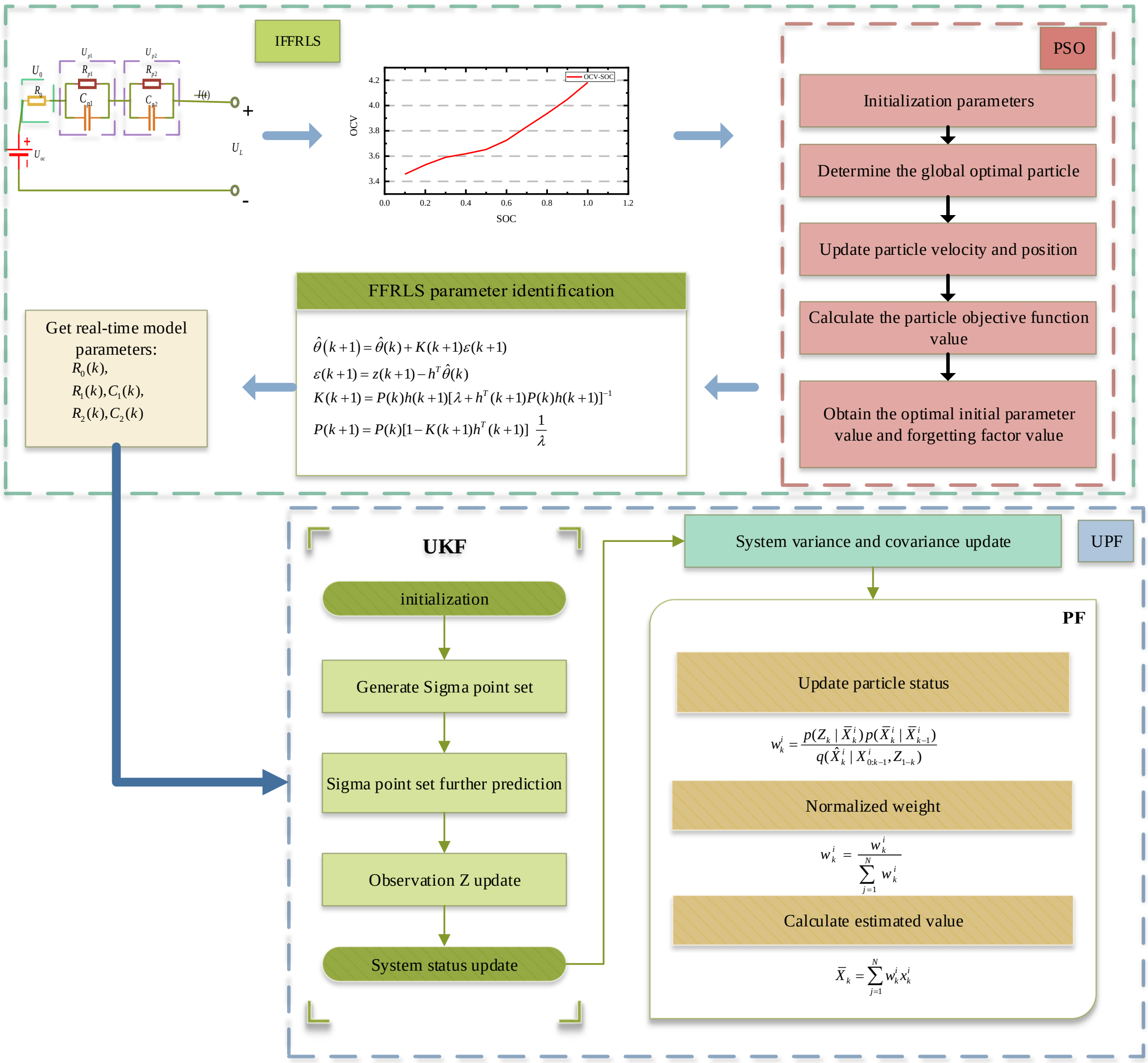


Figure 7 IFFRLS-UPF algorithm flow chart

Figure 7 introduces the main flow of the IFFRLS-UPF algorithm. The second-order RC model is used as an equivalent model to study the state of charge of lithium batteries, and the UKF algorithm is used as the proposed distribution function of the PF algorithm to form a new algorithm UPF algorithm. This algorithm fully embodies the advantages of the two algorithms. The specific introduction of the UPF algorithm will be expanded in the following. The processes of the UPF algorithm are as follows.

(1) The particles abstracted from the prior distribution $p(X_0)$ are used as the initial state of the new particle set.

$$X_0^i = E\left(X_0^i\right) \tag{7}$$

$$P_0^i = E[(X_0^i - \bar{X}_0^i)(X_0^i - \bar{X}_0^i)^{\mathrm{T}}] \tag{8}$$

$$X_0^{i,a} = E\left(\bar{X}_0^{i,a}\right) = [\bar{X}_0^{i\,\mathrm{T}}, 0, 0]^{\mathrm{T}} \tag{9}$$

$$P_0^{i,a} = E\left[\left(X_0^{i,a} - \bar{X}_0^{i,a}\right)\left(X_0^{i,a} - \bar{X}_0^{i,a}\right)^{\intercal}\right] = \begin{bmatrix} P_0^i & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{bmatrix} \tag{10}$$

(2) Generate a set of Sigma points.

$$X_{k-1}^{i,a} = [X_{k-1}^{i,a}\ X_{k-1}^{i,a} \pm \sqrt{(n_a+\lambda)P_{k-1}^{i,a}}] \tag{11}$$

(3) A further prediction of Sigma point set.

$$\bar{\mathrm{X}}_{\mathrm{k|k-1}}^{\mathrm{i,a}} = \mathrm{f}(\mathrm{X}_{\mathrm{k-1}}^{\mathrm{i,x}}, \mathrm{X}_{\mathrm{k-1}}^{\mathrm{i,v}}) \tag{12}$$

$$\bar{\mathrm{X}}_{\mathrm{k|k-1}}^{\mathrm{i}} = \sum_{\mathrm{j=0}}^{\mathrm{2n_a}} \mathrm{W}_{\mathrm{j}}^{\mathrm{m}} \mathrm{X}_{\mathrm{j,k|k-1}}^{\mathrm{i,x}} \tag{13}$$

$$\mathrm{P}_{\mathrm{k|k-1}}^{\mathrm{i}} = \sum_{\mathrm{j=0}}^{\mathrm{2n_a}} \mathrm{W}_{\mathrm{j}}^{\mathrm{c}} \left[\mathrm{X}_{\mathrm{j,k|k-1}}^{\mathrm{i,x}} - \bar{\mathrm{X}}_{\mathrm{k|k-1}}^{\mathrm{i}}\right][\mathrm{X}_{\mathrm{j,k|k-1}}^{\mathrm{i,x}} - \bar{\mathrm{X}}_{\mathrm{k|k-1,}}^{\mathrm{i}}]^{\intercal} \tag{14}$$

$$\mathrm{Z}_{\mathrm{k|k-1}}^{\mathrm{i}} = \mathrm{h}(\mathrm{X}_{\mathrm{k|k-1}}^{\mathrm{i}}, \mathrm{X}_{\mathrm{k-1}}^{\mathrm{i,n}}) \tag{15}$$

$$\bar{\mathrm{Z}}_{\mathrm{k|k-1}}^{\mathrm{i}} = \sum_{\mathrm{j=0}}^{\mathrm{2n_a}} \mathrm{W}_{\mathrm{j}}^{\mathrm{c}} \mathrm{Z}_{\mathrm{k|k-1}}^{\mathrm{i}} \tag{16}$$

(4) With the observations obtained, the system state is renewed.

$$P_{\bar{Z}_k} = \sum_{j=0}^{2n_a} W_j^c \left[Z_{j,k|k-1}^i - Z_{k|k-1}^i\right][Z_{j,k|k-1}^i - Z_{k|k-1}^i]^{\intercal} \tag{17}$$

$$P_{X_k, Z_k} = \sum_{j=0}^{2n_a} W_j^c [X_{j,k|k-1}^i - X_{k|k-1}^i][X_{j,k|k-1}^i - X_{k|k-1}^i]^{\intercal} \tag{18}$$

$$K = P_{\bar{Z}_k} P_{X_k, Z_k} \tag{19}$$

$$\bar{X}_k^i = \bar{X}_{k|k-1}^i + K(Z_k - \bar{Z}_{k|k-1}^i) \tag{20}$$

$$\hat{P}_k^i = P_{k|k-1}^i - K P_{\bar{Z}_k} K^{\intercal} \tag{21}$$

(5) Using the particle set updated by the algorithm, the state of the ith particle is updated as follows.

$$\hat{X}_k^i \sim q\left(\bar{X}_k^i \middle| X_{0:k-1}^i, Z_{1:k}\right) = N(\bar{X}_k^i, \hat{P}_k^i) \tag{22}$$

$$\hat{X}_{0:k}^i \triangleq (X_{0:k-1}^i, \bar{X}_k^i) \tag{23}$$

$$\hat{P}_{0:k}^i \triangleq (P_{0:k-1}^i, \hat{P}_k^i) \tag{24}$$

(6) Calculate the weight W_k^i for each particle.

$$q[X_k | X_{0:k}(i), y] = p[X_k | X_{k-1}(i)] \tag{25}$$

$$W_k^i = \frac{p\left(Z_k \middle| \hat{X}_k^i\right) p(\hat{X}_k^i | X_{k-1}^i)}{q(X_k^i | X_{0:k}^i, Z_{1-k})} \tag{26}$$

(7) Normalized weights.

$$w_k^i = w_k^i / \sum_{i=1}^{N} w_k^i \tag{27}$$

(8) State estimation.

$$\bar{X}_k = \sum_{j=1}^{N} w_k^i x_k^i \tag{28}$$

(9) Whether resampling is necessary can be judged by computing the efficacious particle count.

$$w_k^i = w_{k-1}^i \frac{p\left(Z_k \middle| x_k^i\right) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, Z_{1:k})} \tag{29}$$

It needs to be performed for the resampling step, with the value of the particles being inferior to the pre-set threshold. Otherwise, it should skip this step. Repeat step (5) ~ step (9) until implementing all state estimates for the entire period. In the traditional particle filter, against the severe problem, with particle degradation, this algorithm can effectively enhance the multiformity of particles by guiding the sampling according to the UKF to compute the mean and variance of each particle.

In estimating SOC using the proposed IFFRLS-UPF algorithm, we randomly select particles as training samples during sampling and then perform continuous training to obtain our desired model and parameters. The bootstrap method is used to divide the training samples and test samples.

3. Experimental analysis

3.1 Test platform construction

The experimental test in this paper chooses a ternary lithium battery with a rated capacity of 70Ah. The battery test system used for charging and discharging, the constant temperature box, and the lithium battery with a rated capacity of 70Ah is the experimental equipment used in this experiment, and the temperature stated in this experiment is 25°C.

The actual discharging capacity of a battery is the first task to estimate the SOC. However, in the actual test, there are often large deviations between the actual discharge capacity and the rated capacity due to battery aging and other reasons. In this paper, we choose the IFFRLS-UPF algorithm to estimate the SOC with higher precision.

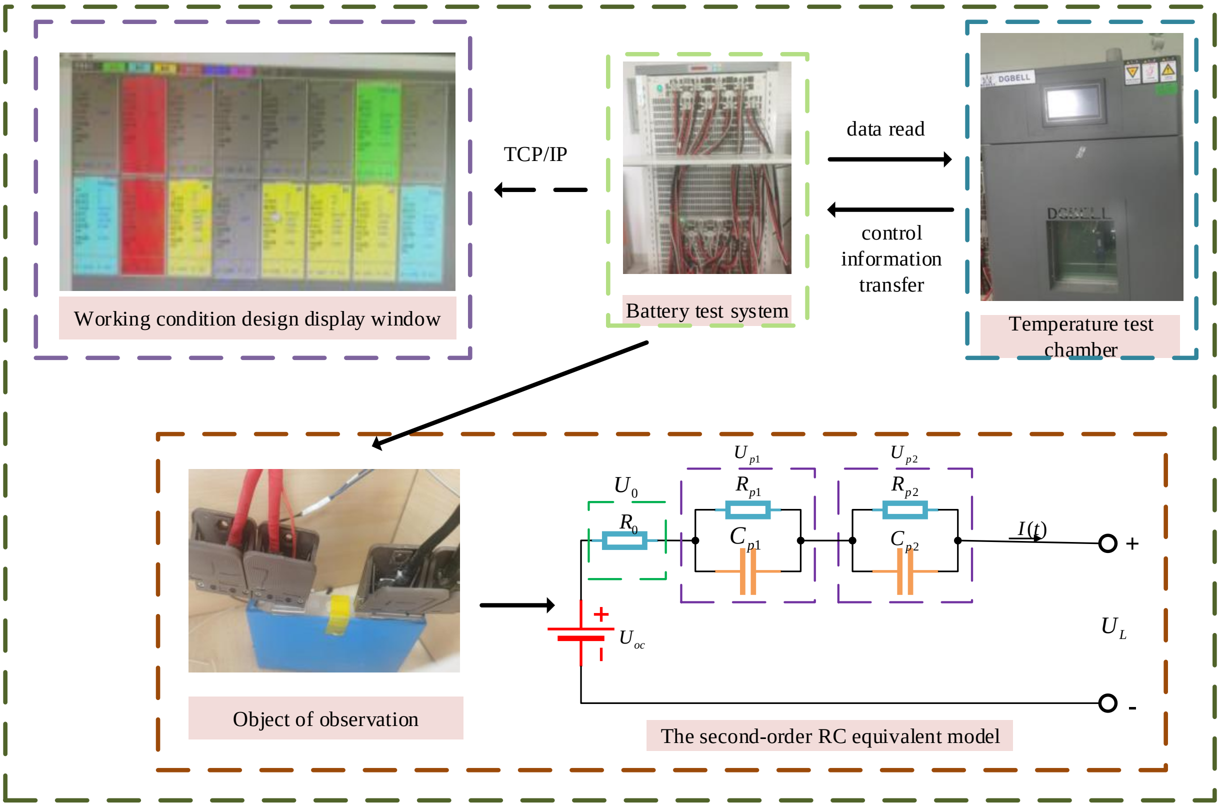
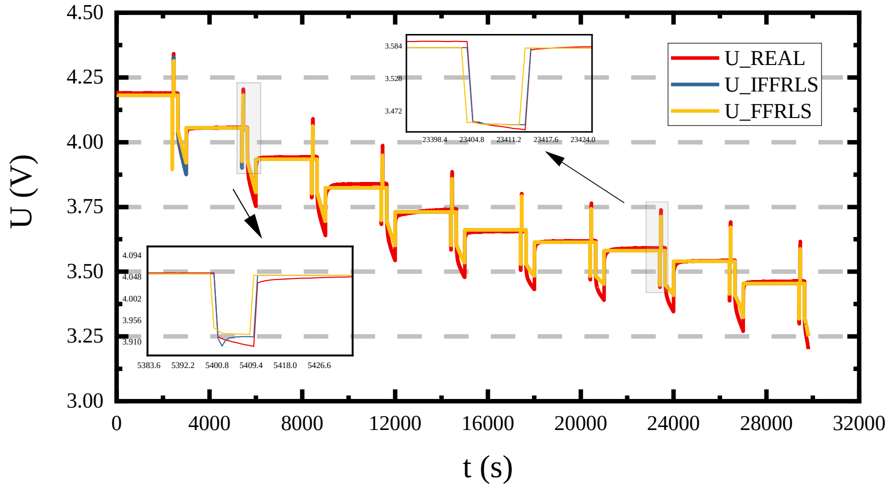


Figure 8 Lithium battery experimental test platform construction

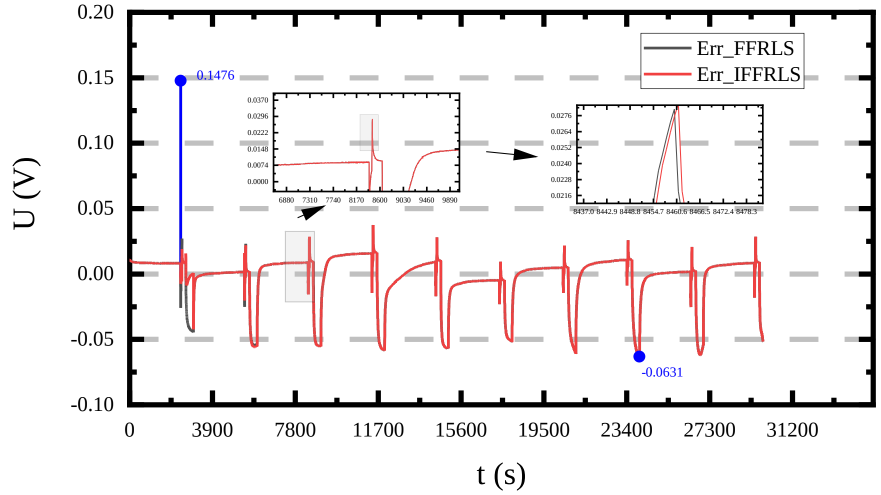
Figure 8 shows the experimental equipment used in our experiments. The experimental object is a ternary lithium battery with a rated capacity of 70Ah. The experimental equipment includes a battery test system to detect the voltage, current and temperature of the battery and to provide a constant temperature environment for the battery. Based on this equipment platform, we can complete all lithium battery testing experiments.

3.2 IFFRLS parameter identification experimental results

The forgetting factor method is also called the decaying memory method or the exponential window method. Its basic idea is to add a forgetting factor to the old data so that decreasing the impact of the old data and strengthen the effect of the new data. Running the PSO algorithm, we can screen out the optimal parameter value and forgetting factor value. Bring the obtained optimal solution into the FFRLS algorithm, then we can gain the target value after repeated iterations. Figure 9 illustrates the result of parameter identification.



(a) Model voltage and actual voltage comparison results



(b) Voltage Error of FFRLS Model

Figure 9 Contrast of emulation voltage and practical voltage based on the FFRLS method and its error

Based on the second-order RC model, the comparison curves of simulated and actual voltages obtained by different algorithms are plotted in Figure 9(a), while the error curves are reflected in Figure 9(b). From Figure 9(a), we can see that the simulated voltage, which the IFFRLS algorithm optimized by the PSO algorithm produced, is closer to the practical voltage obtained in the experiment than the simulated voltage acquired by the unoptimized FFRLS algorithm. The reason for such a result is that the optimized IFFRLS algorithm obtains the optimal result after filtering out the optimal parameter value and forgetting the factor value based on the error between the terminal voltage and the model voltage as the objective function. From Figure 9(b), it can be obtained that the maximum error between the simulated voltage of the IFFRLS algorithm and the actual voltage is 0.0631V. The result is small, compared with the maximum error value of 0.1476V, between the simulated voltage and practical voltage, with the FFRLS algorithm. It confirms that the IFFRLS algorithm can effectively drop the error between the model voltage and actual voltage.

In the premier phase of parameter identification, it can be seen from the above analysis that the parameter value changes violently, and its variance undulates enormously, with the bestial deviation from the difference of the model parameter initial values. With the lengthening of the identification time, in the sustained iterative procedure, the change of each parameter is relatively gentle, the variety of the variance tends to be stable, and the parameter identification value at this time is comparatively precise. The performance indicators of the two algorithms are listed in Table 3.

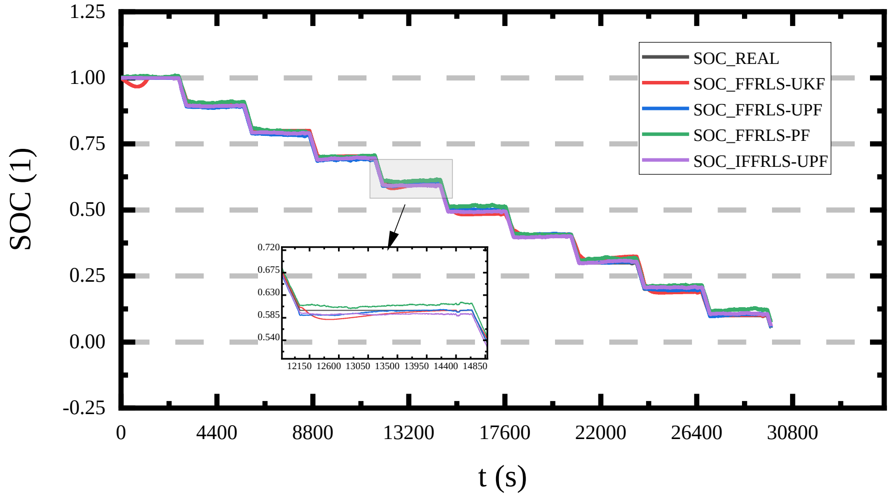
Table 3 Performance Indicators of FFRLS Algorithm and IFFRLS Algorithm

Algorithm	Max	MAE	RMSE
FFRLS	0.1476	0.011686	0.018693
IFFRLS	0.0631	0.011265	0.018215

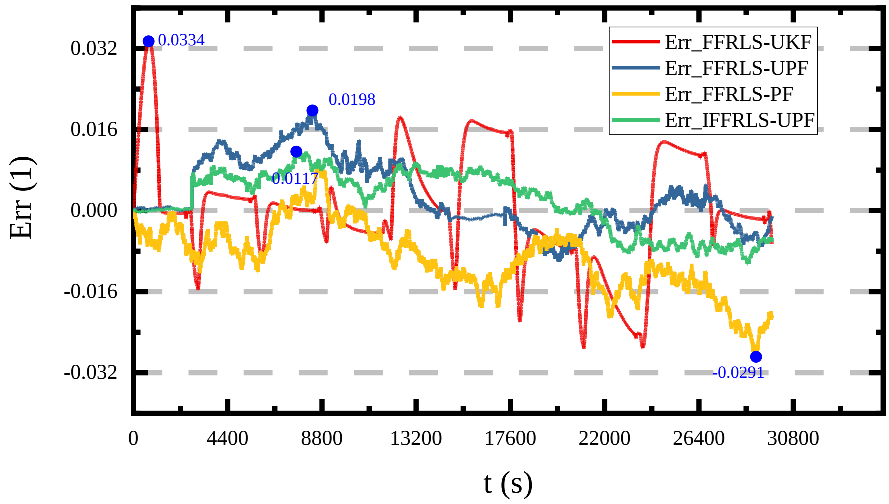
Table 3 respectively calculates two performance indicators of the FFRLS algorithm and IFFRLS algorithm: mean absolute error (MAE) and root mean square error (RMSE). MAE is the mean of absolute errors, which is essentially a more general form of the mean of deviations. RMSE measures the average size of the error and is the square root of the average of the squared differences between the predicted value and the actual observation. RMSE shows the overall average estimation effect. Under the same conditions, the smaller the value of RMSE, the better the estimation effect of the algorithm. Therefore, in the process of model parameter identification, through the comparison between the different performance indicators of the IFFRLS algorithm and the FFRLS algorithm, it can be known that the estimation effect of the IFFRLS algorithm is superior to that of the FFRLS algorithm.

3.3 Experimental results of HPPC working conditions

Through the IFFRLS online parameter identification experiment, building a second-order RC equivalent circuit model. Comparing the actual data with the estimated data obtained in other working conditions verifies the validity of the model. According to the established second-order RC equivalent circuit model, combined with particle filter algorithm and unscented particle filter algorithm, SOC is estimated for HPPC operating conditions. Figure 10 provides a comparison between the estimated value and the practical value.



(a) SOC estimate under HPPC working conditions



(b) SOC estimate error under HPPC working conditions

Figure 10 SOC estimation curves and error curves under HPPC condition

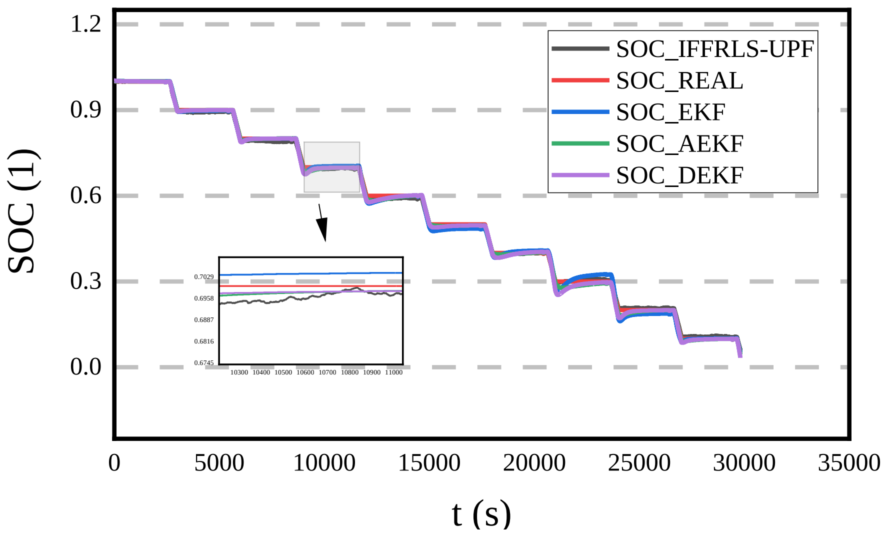
Figure 10 describes the SOC estimates obtained under the four different algorithms and the SOC estimates obtained by the ampere-hour integration method, and the SOC estimates obtained by the ampere-hour integration method are used as the reference values for the SOC estimation results of other algorithms. From the figure, we can see that under the same conditions, among the four algorithms, the IFFRLS-UPF algorithm proposed in this paper can converge to the practical value of SOC faster. In Figure 10(b), the four different algorithms, with the error curves between the SOC values and the actual values, reflects that the error curve of the FFRLS-UKF algorithm fluctuates mightily, and it describes that the algorithm has poor stability when estimating SOC. However, the fluctuation of the error curve by the FFRLS-PF algorithm is lower than that of the FFRLS-UKF algorithm. Its convergence, the error curve from the FFRLS-PF algorithm, is much worse than the proposed algorithm. Although the stability and convergence of the FFRLS-UPF algorithm are better than the FFRLS-UKF and FFRLS-PF algorithms. But compared with the IFFRLS-UPF algorithm proposed in this paper, its fluctuation is larger. In general, the IFFRLS-UPF algorithm proposed in this paper is superior to the other three algorithms in terms of stability and convergence. The calculated values of several performance indicators of the four algorithms are provided in Table 4.

Table 4 Comparison of SOC estimation results under HPPC conditions

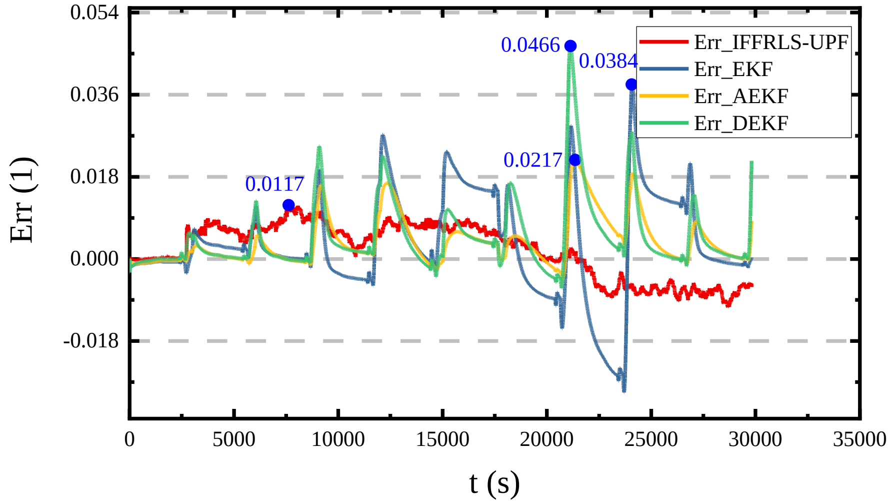
Algorithm	Max	MAE	RMSE
FFRLS-UKF	0.0334	0.007986	0.011042
FFRLS-PF	0.0287	0.009891	0.011668
FFRLS-UPF	0.0198	0.005562	0.007382
IFFRLS-UPF	0.0117	0.005531	0.006241

From Table 4, we can see that the maximum error value, MAE value, and RMSE value of the IFFRLS-UPF algorithm proposed in this paper are the smallest among the four algorithms. The error maximum of the IFFRLS-UPF algorithm proposed in this paper can reach 0.0117, its MAE value is 0.005531, and its RMSE value is 0.006241. Therefore, the accuracy of the IFFRLS-UPF algorithm proposed in this paper is better than the other three algorithms, whether in terms of maximum error value or root mean square error value.

The SOC estimation of the proposed algorithm is better overall than other algorithms that are widely used today, and we can observe and analyze the specific estimation curves.



(a) SOC result curves of IFFRLS-UPF algorithm and other existing algorithms



(b) Error result curves of IFFRLS-UPF algorithm and other existing algorithms

Figure 11 Comparison of IFFRLS-UPF algorithm with other existing algorithms

Figure 11 shows the comparison curve between the algorithm proposed in this paper and several algorithms that are often used at present. The EKF algorithm, AEKF algorithm, and DEKF algorithm are all extended and improved algorithms of the EKF algorithm, and they are also several algorithms that are widely used, we can analyze that the SOC estimation result of the algorithm proposed in this paper is more in line with the actual result, its error is the smallest, the robustness is the best, and the oscillation degree of the other algorithms is more severe than that of the algorithm proposed in this paper. Through calculation, the performance indicators of several algorithms are shown in Table 5.

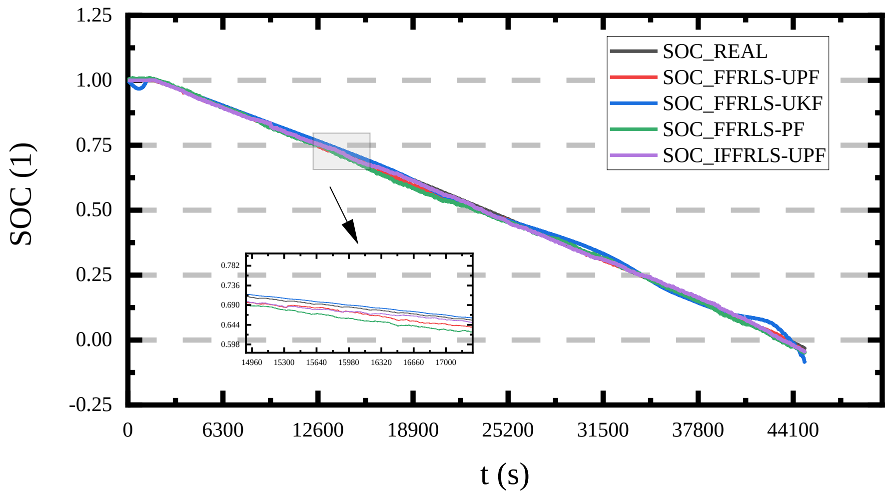
Table 5 Performance Indicators of IFFRLS-UPF Algorithm and Several Other Algorithms

Algorithm	Max	MAE	RMSE
EKF	0.0384	0.00809	0.0114
AEKF	0.0217	0.00422	0.0065
DEKF	0.0466	0.00519	0.00895
IFFRLS-UPF	0.0117	0.005531	0.006241

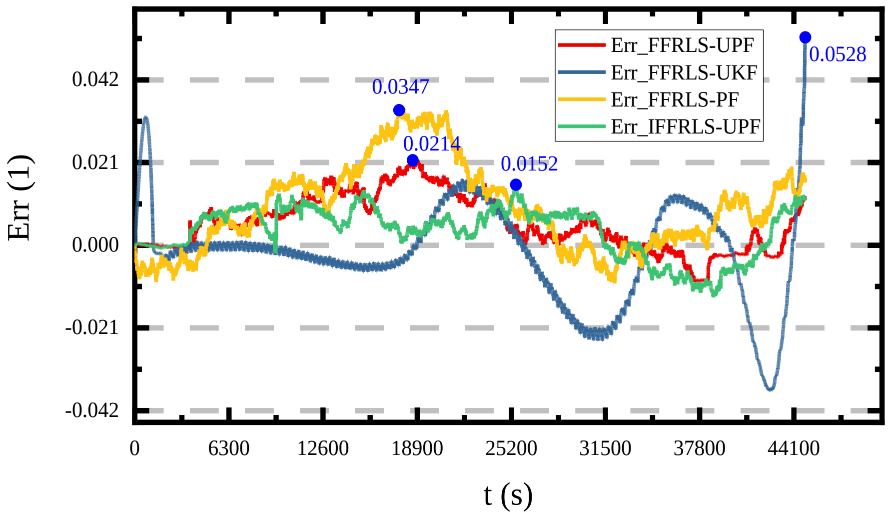
Table 5 calculates the performance index comparison between the IFFRLS-UPF algorithm and several other algorithms. Among them, the mean absolute errors of the EKF algorithm and the DEKF algorithm are larger than those of the proposed algorithm. Although the mean absolute error of the AEKF algorithm is slightly better than the proposed algorithm, However, its root mean square error is larger than the proposed algorithm. In addition, the root mean square error of the EKF algorithm and the DEKF algorithm is much larger than that of the proposed algorithm. Therefore, it can be shown that the proposed algorithm is better than other algorithms as a whole, and its maximum error value is indeed the smallest, which further proves the superiority of the proposed algorithm.

3.4 Experimental results of BBDST working condition

Through the IFFRLS online parameter identification experiment, a second-order RC equivalent circuit model is performed. It can verify the model validity with the actual data and estimated data in other working conditions. According to the constructed second-order RC equivalent circuit model, combined with the particle filter algorithm and the unscented particle filter algorithm, the SOC estimation of the BBDST condition is executed. The comparison between the estimated value and the actual value is plotted in the following Figure 12.



(a) SOC estimate under BBDST working conditions



(b) SOC estimate error under BBDST working conditions

Figure 12 SOC estimation curves and error curves under BBDST condition

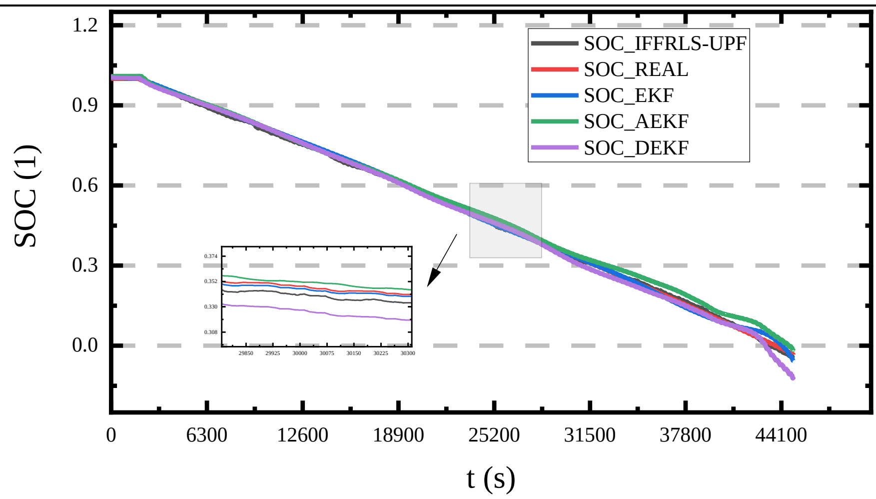
Under the Beijing Bus Dynamic Stress Test (BBDST) condition, Figure 12 reflects the SOC estimation of the four algorithms. And the error curves of the four algorithms are plotted in Figure 12(b). In the later stage, the error curve of the FFRLS-UKF algorithm among the four algorithms fluctuates mightily, even appears divergence. The curve fluctuation of the FFRLS-PF algorithm is relatively large, but its convergence is better than that of the FFRLS-UKF algorithm. Compared with the first two algorithms, the error curve of the FFRLS-UPF algorithm fluctuates less, indicating that it has higher stability. However, the IFFRLS-UPF algorithm proposed in this paper has less fluctuation in the error curve, higher robustness and convergence, and higher accuracy than the other three algorithms. The maximum error value, mean absolute error value, and root means square error value of the four algorithms are calculated in Table 6.

Table 6 Comparison of SOC estimation results under BBDST conditions

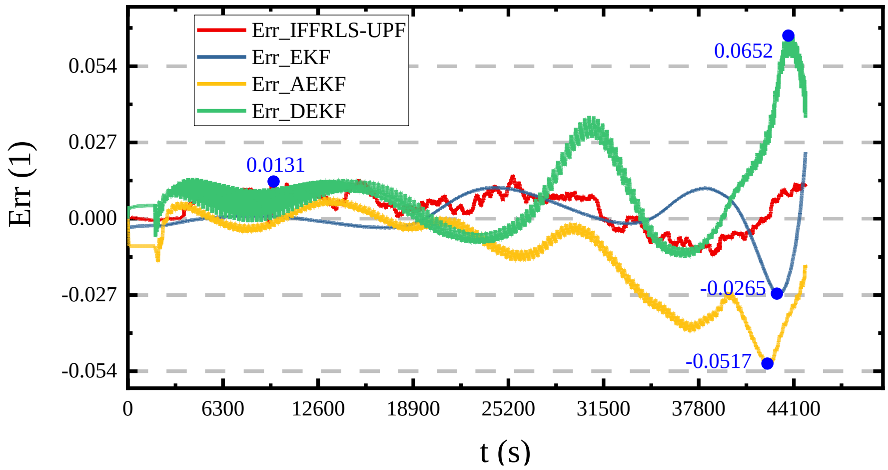
Algorithm	Max	MAE	RMSE
FFRLS-UKF	0.0528	0.009038	0.012586
FFRLS-PF	0.0347	0.011254	0.014494
FFRLS-UPF	0.0214	0.00699	0.009001
IFFRLS-UPF	0.0152	0.006365	0.007262

Under the BBDST condition, Table 6 introduces several performance indicators of the four algorithms. Among them, the maximum error value of the IFFRLS-UPF algorithm is only 0.0152, which is the highest accuracy among the four algorithms. Its MAE value reaches 0.006365, which is better than the other three algorithms. Its RMSE value achieves 0.007262, which is the smallest among the four algorithms. The smaller the RMSE value, the higher the accuracy. From this table, we can learn that the proposed algorithm is better than the other three algorithms, with the superiority of the IFFRLS-UPF algorithm.

The proposed algorithm in this paper is compared with the previously frequently used algorithms to obtain the SOC estimation curves of several algorithms under the BBDST operating conditions.



(a) SOC result curves of IFFRLS-UPF algorithm and other existing algorithms



(b) Error result curves of IFFRLS-UPF algorithm and other existing algorithms

Figure 13 Result curves of IFFRLS-UPF algorithm and other existing algorithms under BBDST condition

Figure 13 shows the SOC comparison results between the proposed algorithm and several commonly used algorithms under BBDST conditions. From the error curve, it can be seen that the curve of the proposed algorithm is relatively flat as a whole, without severe fluctuations, indicating that the proposed algorithm has good stability. Several other algorithms are very stable in the early stage, but there will be large fluctuations at the end of the experiment, indicating that the robustness of the other algorithms is not as good as the proposed algorithm, and can be analyzed by calculating the performance indicators of several algorithms.

Table 7 Performance Indicators of IFFRLS-UPF Algorithm and Several Common Algorithms under BBDST Condition

Algorithm	Max	MAE	RMSE
EKF	0.0265	0.00477	0.00721
AEKF	0.0517	0.01307	0.01905
DEKF	0.0652	0.00759	0.01702

IFFRLS-UPF	0.0152	0.006365	0.007262

Table 7 calculates the performance indicators of the proposed algorithm and several commonly used SOC estimation algorithms. Although the MAE and RMSE of the EKF algorithm are slightly smaller than the proposed algorithm, its maximum error is much larger than the proposed error. The mean absolute error and root mean square error of the proposed algorithm are also very small, which is better than the EKF algorithm as a whole. The performance indicators of the other two algorithms, the AEKF algorithm and the DEKF algorithm, are worse than the proposed algorithm. Therefore, the proposed algorithm is better than other algorithms.

From the experimental results of different working conditions, the proposed algorithm has high accuracy, good stability, and high robustness, however, due to its computational complexity, it takes a longer time, on the whole, it is a new idea for SOC estimation, which can be explored in depth to make up for its shortcomings in the subsequent research.

4. Conclusions

In this paper, the unscented Kalman filter and the particle filter algorithm are combined, then employing the PSO algorithm to optimize the FFRLS algorithm, and appears the IFFRLS algorithm. Moreover, the IFFRLS algorithm identifies the parameters in the early stage, then an IFFRLS-UPF algorithm is proposed. The SOC estimation result of the IFFRLS-UPF algorithm under HPPC conditions is better than other algorithms. The presented algorithm improves the accuracy of SOC estimation, with a percent of 2.17, compared with the FFRLS-UKF algorithm. Compared with the FFRLS-PF algorithm, it can reach a percent of 1.7. And a percent of 0.81, compared with the FFRLS-UPF algorithm. Under the BBDST working condition, the accuracy of the proposed algorithm is a percent of 3.76 higher than the FFRLS-UKF algorithm, a percent of 1.95 better than the FFRLS-PF algorithm, and a percent of 0.62 superior to the FFRLS-UPF algorithm when estimating SOC. The results display that the IFFRLS-UPF algorithm can reckon the state of charge of lithium batteries well, and the algorithm has extremely high robustness and convergence. Of course, the algorithm proposed in this paper is not perfect. The PSO algorithm used to optimize the FFRLS algorithm has certain limitations and defects in the optimization, so further research and exploration can be carried out. Secondly, the PF algorithm itself is not constrained by system factors. However, the improved formed UPF algorithm introduces the UKF algorithm, which makes Gaussian assumptions on the system, thus leading to the proposed algorithm may be constrained by Gaussian models, which need to investigate further in depth next.

Acknowledgments

The work was supported by the National Natural Science Foundation of China (No. 62173281).

References

1. Huang, K., Y. Guo, and Z. Li, *Review of state of charge estimation methods for power lithium-ion battery.* Chinese Journal of Power Sources, 2018. **42**(9): p. 1398-1401.
2. Chen, L., et al., *Remaining useful life prediction of lithium-ion battery with optimal input sequence selection and error compensation.* Neurocomputing, 2020. **414**: p. 245-254.
3. Li, H., X. Zhang, and W. Zhang, *SOC estimate method and application of high capacity lithium-ion battery.* Chinese Journal of Power Sources, 2015. **39**(5): p. 1100-1102.
4. Yang, J., et al., *Adaptive State-of-Charge Estimation Based on a Split Battery Model for Electric Vehicle Applications.* Ieee Transactions on Vehicular Technology, 2017. **66**(12): p. 10889-10898.
5. Zhang, L., et al., *Battery heating for lithium-ion batteries based on multi-stage alternative currents.* Journal of Energy Storage, 2020. **32**: p. 101885.
6. Xiong, R., et al., *A data-driven multi-scale extended Kalman filtering based parameter and state estimation approach of lithium-ion polymer battery in electric vehicles.* Applied Energy, 2014. **113**: p. 463-476.
7. Fan, Y., et al., *A Novel Adaptive Function—Dual Kalman Filtering Strategy for Online Battery Model Parameters and State of Charge Co-Estimation.* Energies, 2021. **14**(8): p. 2268.
8. Hannan, M.A., et al., *A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations.* Renewable & Sustainable Energy Reviews, 2017. **78**: p. 834-854.
9. Shen, P., et al., *The Co-estimation of State of Charge, State of Health, and State of Function for Lithium-Ion Batteries in Electric Vehicles.* Ieee Transactions on Vehicular Technology, 2018. **67**(1): p. 92-103.
10. Wei, Z.B., et al., *Noise-Immune Model Identification and State-of-Charge Estimation for Lithium-Ion Battery Using Bilinear Parameterization.* Ieee Transactions on Industrial Electronics, 2021. **68**(1): p. 312-323.
11. Zheng, Y.J., et al., *Investigating the error sources of the online state of charge estimation methods for lithium-ion batteries in electric vehicles.* Journal of Power Sources, 2018. **377**: p. 161-188.
12. Yue, W., et al., *A nonlinear fractional-order H-infinity observer for SOC estimation of battery pack of electric vehicles.* Proceedings of the Institution of Mechanical Engineers Part D-Journal of Automobile Engineering, 2021. **235**(9): p. 2484-2495.
13. Ren, Z., et al., *A comparative study of the influence of different open circuit voltage tests on model-based state of charge estimation for lithium-ion batteries.* International Journal of Energy Research, 2021. **45**(9): p. 13692-13711.
14. Zhang, L., et al., *A Sparse Learning Machine for Real-Time SOC Estimation of Li-ion Batteries.* Ieee Access, 2020. **8**: p. 156165-156176.
15. Xing, Y., et al., *State of charge estimation of lithium-ion batteries using the open-circuit voltage at various ambient temperatures.* Applied Energy, 2014. **113**: p. 106-115.
16. Lee, S., et al., *State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge.* Journal of Power Sources, 2008. **185**(2): p. 1367-1373.
17. Shi, H., et al., *A Novel Dual Correction Extended Kalman Filtering Algorithm for The State of Charge Real-Time Estimation of Packing Lithium-Ion Batteries.* International Journal of Electrochemical Science, 2020. **15**(12): p. 12706-12723.

18. Xu, W., et al., *A novel adaptive dual extended Kalman filtering algorithm for the Li-ion battery state of charge and state of health co-estimation.* International Journal of Energy Research, 2021. **45**(10): p. 14592-14602.

19. He, Z., et al., *State of charge estimation of power Li-ion batteries using a hybrid estimation algorithm based on UKF.* Electrochimica Acta, 2016. **211**: p. 101-109.

20. Wei, X., Y. Mo, and Z. Feng, *Lithium-ion Battery Modeling and State of Charge Estimation.* Integrated Ferroelectrics, 2019. **200**(1): p. 59-72.

21. Li, S., et al., *Fractional-order modeling and SOC estimation of lithium-ion battery considering capacity loss.* International Journal of Energy Research, 2019. **43**(1): p. 417-429.

22. Lee, K.-T., M.-J. Dai, and C.-C. Chuang, *Temperature-Compensated Model for Lithium-Ion Polymer Batteries With Extended Kalman Filter State-of-Charge Estimation for an Implantable Charger.* Ieee Transactions on Industrial Electronics, 2018. **65**(1): p. 589-596.

23. Zheng, Y.J., et al., *State-of-charge inconsistency estimation of lithium-ion battery pack using mean-difference model and extended Kalman filter.* Journal of Power Sources, 2018. **383**: p. 50-58.

24. Misyris, G.S., et al., *State-of-Charge Estimation for Li-Ion Batteries: A More Accurate Hybrid Approach.* Ieee Transactions on Energy Conversion, 2019. **34**(1): p. 109-119.

25. Al-Gabalawy, M., et al., *State of charge estimation of a Li-ion battery based on extended Kalman filtering and sensor bias.* International Journal of Energy Research, 2021. **45**(5): p. 6708-6726.

26. Zhang, S., X. Guo, and X. Zhang, *An improved adaptive unscented kalman filtering for state of charge online estimation of lithium-ion battery.* Journal of Energy Storage, 2020. **32**: p. 101980.

27. Xu, W., et al., *Novel reduced-order modeling method combined with three-particle nonlinear transform unscented Kalman filtering for the battery state-of-charge estimation.* Journal of Power Electronics, 2020. **20**(6): p. 1541-1549.

28. Jiang, C., et al., *A state-of-charge estimation method of the power lithium-ion battery in complex conditions based on adaptive square root extended Kalman filter.* Energy, 2021. **219**.

29. Huang, C., et al., *Robustness Evaluation of Extended and Unscented Kalman Filter for Battery State of Charge Estimation.* Ieee Access, 2018. **6**: p. 27617-27628.

30. Xu, C., L. Li, and Y. Yang, *State of Health Estimation of Lithium-Ion Battery Based on Improved Particle Filter.* Automobile Technology, 2020(12): p. 19-24.

31. Ye, M., et al., *A double-scale and adaptive particle filter-based online parameter and state of charge estimation method for lithium-ion batteries.* Energy, 2018. **144**: p. 789-799.

32. Miao, Q., et al., *Remaining useful life prediction of the lithium-ion battery using particle filtering.* Journal of Chongqing University. Natural Science Edition, 2013. **36**(8): p. 47-52,60.

33. Xie, Y., et al., *A new method of unscented particle filter for high-fidelity lithium-ion battery SOC estimation.* Energy Storage Science and Technology, 2021. **10**(2): p. 722-731.

34. Walker, E., S. Rayman, and R.E. White, *Comparison of a particle filter and other state estimation methods for prognostics of lithium-ion batteries.* Journal of Power Sources, 2015. **287**: p. 1-12.

35. Ji, Y.-j., S.-l. Qiu, and G. Li, *Simulation of second-order RC equivalent circuit model of lithium battery based on variable resistance and capacitance.* Journal of Central South University, 2020. **27**(9): p. 2606-2613.

36. Xiong, R., et al., *An estimation method for lithium-ion battery SOC of special robots based on Thevenin model and improved extended Kalman.* Energy Storage Science and Technology, 2021. **10**(2): p. 695-704.

37. Wu, X. and X. Zhang, *Parameters identification of second order RC equivalent circuit model for lithium batteries.* Journal of Nanjing University. Natural Sciences, 2020. **56**(5): p. 754-761.

38. Ding, Z., et al., *SOC Estimation of Lithium-ion Battery Based on Ampere Hour Integral and Unscented Kalman Filter.* China Mechanical Engineering, 2020. **31**(15): p. 1823-1830.

39. Hu, M., et al., *Lithium-ion battery modeling and parameter identification based on fractional theory.*

Energy, 2018. **165**: p. 153-163.

40. Zhang, W., et al., *Joint State-of-Charge and State-of-Available-Power Estimation Based on the Online Parameter Identification of Lithium-Ion Battery Model.* Ieee Transactions on Industrial Electronics, 2022. **69**(4): p. 3677-3688.

41. Hu, X.-s., F.-c. Sun, and Y. Zou, *Online model identification of lithium-ion battery for electric vehicles.* Journal of Central South University of Technology, 2011. **18**(5): p. 1525-1531.

42. Chen, Z., et al., *A New Method of Insulation Detection on Electric Vehicles Based on a Variable Forgetting Factor Recursive Least Squares Algorithm.* Ieee Access, 2021. **9**: p. 73590-73607.

43. Liu, Z.F., et al., *Dynamic economic emission dispatch considering renewable energy generation: A novel multi-objective optimization approach.* Energy, 2021. **235**.

44. Li, L.L., et al., *Improved tunicate swarm algorithm: Solving the dynamic economic emission dispatch problems.* Applied Soft Computing, 2021. **108**.

45. Li, W.H., et al., *Digital twin for battery systems: Cloud battery management system with online state-of-charge and state-of-health estimation.* Journal of Energy Storage, 2020. **30**.

46. Tan, C.M., P. Singh, and C. Chen, *Accurate Real Time On-Line Estimation of State-of-Health and Remaining Useful Life of Li ion Batteries.* Applied Sciences, 2020. **10**(21): p. 7836.

47. Jiang, C., et al., *A state-of-charge estimation method of the power lithium-ion battery in complex conditions based on adaptive square root extended Kalman filter.* Energy, 2021. **219**: p. 119603.

48. Dong, G., et al., *An online model-based method for state of energy estimation of lithium-ion batteries using dual filters.* Journal of Power Sources, 2016. **301**: p. 277-286.

49. Wang, Y.J., et al., *A comprehensive review of battery modeling and state estimation approaches for advanced battery management systems.* Renewable & Sustainable Energy Reviews, 2020. **131**.

50. He, L., et al., *State of charge estimation by finite difference extended Kalman filter with HPPC parameters identification.* Science China-Technological Sciences, 2020. **63**(3): p. 410-421.