

Research Article

Improved Particle Filter Using Clustering Similarity of the State Trajectory with Application to Nonlinear Estimation: Theory, Modeling, and Applications

Ziquan Jiao ^{1,2}, Zhiqiang Feng ², Na Lv ³, Wenjing Liu ⁴, and Haijian Qin ²

¹School of Mechanical and Electrical Engineering, Guilin University of Electronic Technology, Guilin 541004, China

²Guangxi Engineering Technology Research Center of Ship Digital Design and Advanced Manufacturing, Beibu Gulf University, Qinzhou 535011, China

³Department of Instrument Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

⁴Faculty of Education, Saitama University, Saitama City 338-8570, Japan

Correspondence should be addressed to Zhiqiang Feng; bbgufzq@126.com

Received 8 March 2021; Revised 13 April 2021; Accepted 10 May 2021; Published 26 May 2021

Academic Editor: Ying-Ren Chien

Copyright © 2021 Ziquan Jiao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A clustering similarity particle filter based on state trajectory consistency is presented for the mathematical modeling, performance estimation, and smart sensing of nonlinear systems. Starting from an information fusion model based on the consistency principle of the spatial state trajectory, the predicted observation information of the current particle filter (original trajectory) and future multistage Gaussian particle filter (modified trajectory) are selected as the state trajectories of the sampling particles. Clustering similarity methods are used to measure the state trajectories of the sampling particles and the actual system (reference trajectory). The importance weight of a first-order Markov model is updated with the measurement results. By integrating the targeted compensation scheme of the latest measurement information into the sequential importance sampling process, the adverse effects of the particle degradation phenomenon are effectively reduced. The convergence theorems of the improved particle filter are proposed and proved. The improved filter is applied to practical cases of nonlinear process estimation, economic statistical prediction, and battery health assessment, and the simulation results show that the improved particle filter is superior to traditional filters in estimation accuracy, efficiency, and robustness.

1. Introduction

Nonlinear phenomena are common in natural engineering technology. As a popular research topic with important theoretical and practical value in solving nonlinear problems, state estimation has been applied to problems such as target tracking and navigation, fault diagnosis and detection, process feedback and control, biochemical reaction and extraction, and economic prediction and control. Nonlinear state estimation applies to a wide range of fields, especially the industrial field. Applications include longitudinal vehicle speed estimation [1], fault detection of piezoceramic actuators [2], battery health assessment [3], state detection of impending rollover [4], and state estimation of dynamic systems with hysteresis [5]. Many solutions have been proposed,

such as the Luenberger observer [6], robust observer [7], Gaussian process regression, Kalman filter [8], proportional integral observer [9], unknown input observer [10], high-gain observer [11], and nonsmooth observer [12]. For example, on the premise of satisfying the Gaussian noise distribution, the Kalman filter (KF) calculates the conditional probability density of random variables using a recursive formula and iteratively updates the linear minimum variance estimate. From this come extended KF, unscented KF, invariant extended KF [13], and adaptive extended KF [14]. The above methods have many advantages and wide applications for nonlinear system state estimation, but there is still much room for optimization and improvement in terms of nonlinear complexity and environmental noise uncertainty of different practical applications.

With the rapid development of computer technology, the particle filter (PF) algorithm based on Bayesian and Monte Carlo theories has shown many advantages and considerable potential to solve estimation problems involving nonlinear and non-Gaussian systems. The prototype algorithm, sequential importance sampling (SIS), was formed in the middle of the last century and is mainly applied to physics and automatic control applications. Due to inherent sample degradation and computer hardware limitations, the study of the PF algorithm slowed until 1993, when Gordon et al. [15, 16] introduced a resampling strategy to SIS and developed sequential importance resampling (SIR), which improved the method and laid the theoretical foundation for the PF. With the development of stochastic probability theory and Monte Carlo methods, the auxiliary particle filter (APF) [17] and Gaussian (sum) particle filter (GPF) [18, 19] were proposed. Introduced by Guarniero et al. [20], APF is based on the idea that the latest observation will approach the optimal proposed distribution if an auxiliary variable is imported to represent the prior probability of the current state. When the system noise is strong, the filtering accuracy is difficult to guarantee due to the lack of information. The GPF algorithm of Sun et al. [21] uses a Gaussian distribution to estimate the posterior probability density function (PDF) of a system state under the basic SIS framework, and the mean and variance are recursively obtained. The filtering effect depends heavily on the problem's degree of nonlinearity and is limited to the dimension of the system variables [22]. With the high complexity of current natural engineering structures, the degree of nonlinearity of systems is growing. Although these algorithms somewhat improve the performance of PFs, issues remain, such as low accuracy of filter estimation and poor stability due to particle degradation and depletion, which do not meet the needs of modern engineering. Moreover, as a probabilistic method, the nonlinear estimation of PF leads to uncertainty in the result [23].

For this reason, a clustering similarity particle filter (CSPF) based on the consistency principle of the spatial state trajectory is presented. The clustering similarity method is used to measure the state distance between the actual system and the sampling particles, including the observation information of the current state filtering and future multistage state prediction, to guide the generation and improvement of new distributions and update the weight calculation of the importance sampling process. This makes up for using the prior PDF instead of the importance function in the standard PF algorithm, which can prevent the occurrence of particle degradation and significantly improve the accuracy and robustness of estimation. The resampling strategy in the traditional PF algorithm is abandoned to eliminate particle depletion, which improves the quantization accuracy of uncertainty and efficiency of the algorithm. The above methods adopt the idea of simply modifying the proposed distribution. The designed method uses clustering theory to measure the similarity of observation information corresponding to multistage (from k to $k + L + I$) state trajectories so as to guide the generation and updating of the latest proposal distribution, which significantly improves the computational complexity of the designed method. To ensure the

efficiency of the improved method for different nonlinear state estimation applications, the following two aspects can be improved. (a) The order of the state trajectory should be selected reasonably. Theoretically, the higher the order of trajectory selection, the more accurate the corresponding observation information can express the actual state, and the more accurate the estimation result, but the computational efficiency is greatly decreased. (b) The number of sampling particles N should be reduced appropriately. With the increase of the number of particles, the sampling probability density function will gradually approach the probability distribution of the actual state. While improving the accuracy of state estimation, the computational effort will increase. For these reasons, it is necessary to coordinate the contradiction between estimation accuracy and computational efficiency. The nonlinear state estimation results are largely affected by the signal information, which involves the quality and scale of the research object dataset, appropriate parameter identification, and state tracking training methods. In different research and application objects, increasing the quality and scale of experimental datasets containing more physical model information can improve the state estimation performance of data mining; an appropriate parameter training method can ensure that the model can obtain as much useful information as possible from the dataset and help to establish an appropriate state space model. Based on the above measures, compared to traditional estimation methods under the premise of consistent preconditions, the designed method can greatly improve the accuracy of state results, and it also improves computational efficiency.

The remainder of this paper is structured as follows. Section 2 discusses nonlinear system theory. In Section 3, an improved CSPF algorithm is proposed based on the analysis of the defects of the PF. Section 4 provides a theoretical explanation of the improved algorithm and proves the relevant theorems. Section 5 compares the simulation results of the proposed algorithm and the traditional improved PF algorithm. We discuss our conclusions in Section 6.

2. Theory Statement

We summarize the basic definitions and properties of the state space and optimal Bayesian recursion theory of nonlinear systems.

2.1. State Space Model. We set (Ω, P, F) as a random probability space and define two actual vector stochastic processes: $X = \{X_k, k \in N\}$ and $Y = \{Y_k, k \in N \setminus \{0\}\}$, where sample space Ω is the set of all possible outcomes, event space F is a set of outcomes in the sample space, probability function P assigns a probability to each event in the event space, X is the state process, and Y is the observation information. Let n_x and n_y be the dimensions of the state X and observation information Y , respectively, corresponding to the state space, and define $B(\mathfrak{R}^n)$ as the Borel σ -algebra set on n -dimensional Euclidean space \mathfrak{R}^n . Most nonlinear systems can take

the form of a dynamic state space (DSS) [24]:

$$\begin{aligned} X_k &= F(X_{k-1}, \omega_k) \text{ (state model),} \\ Y_k &= H(X_k, v_k) \text{ (observation equation),} \end{aligned} \quad (1)$$

where k is the discrete-time (stage) index, $X_k \in \mathfrak{R}^n$ is the set of system states at time k , and $Y_k \in \mathfrak{R}^m$ is the observation information corresponding to state X_k . $F(\cdot)$ and $H(\cdot)$ are known state transition and observation functions, respectively, corresponding to the state transition kernel PDF $K(X_k | X_{k-1})$ and observation likelihood PDF $g(Y_k | X_k)$ in the statistical description. The system shift noise $\{\omega_k \in \mathfrak{R}^n\}$ and measurement noise $\{v_k \in \mathfrak{R}^m\}$ are independent and identically distributed (i.i.d.) sequences that obey any PDF form.

The state space X follows the first-order Markov process; i.e., the state X_k of the current moment is only related to the state X_{k-1} of the previous moment. Assuming an initial distribution $X_0 \sim \mu(dX_0)$, the probability density functions of the state transition kernel PDF $K(X_k | X_{k-1})$ and observation likelihood PDF $g(Y_k | X_k)$ are Lebesgue measures:

$$\begin{aligned} K(X_k | X_{k-1}) &= P_\omega(X_k - F(X_{k-1})), \\ g(Y_k | X_k) &= P_v(Y_k - H(X_k)), \end{aligned} \quad (2)$$

where P_ω and P_v are the probability functions under the influence of shift noise ω_k and measurement noise v_k , respectively.

2.2. Optimal Bayesian Recursion Theory. In Bayesian theory [15], the state X_k of a nonlinear system at time k is updated based on the observation information $Y_{1:k}$ to obtain a minimum mean squared error (MSE) estimate. The optimal state estimation of the system is the conditional expected value $E(X_k | Y_{1:k})$ of the posterior PDF $P(X_k | Y_{1:k})$. Based on the premise that the state variable and observation function follow a first-order Markov process, the posterior PDF $P(X_k | Y_{1:k})$ is obtained by two steps of recursive iterations: prediction and updating. $X_{k:l} \triangleq (X_k, X_{k+1}, \dots, X_l)$ and $Y_{k:l} \triangleq (Y_k, Y_{k+1}, \dots, Y_l)$ are defined as the space path information of the state process and observation likelihood from time k to l , respectively.

2.2.1. Prediction. Combined with the transition kernel PDF $K(X_k | X_{k-1})$ calculated by the state space equation, the prior PDF $P(X_k | Y_{1:k-1})$ is predicted using the Chapman-Kolmogorov equation and the posterior PDF $P(X_{k-1} | Y_{1:k-1})$ at time $k-1$:

$$\begin{aligned} P(X_k | Y_{1:k-1}) &= \int P(X_k, X_{k-1} | Y_{1:k-1}) dX_{k-1} \\ &= \int P(X_k | X_{k-1}, Y_{1:k-1}) P(X_{k-1} | Y_{1:k-1}) dX_{k-1} \\ &= \int K(X_k | X_{k-1}) P(X_{k-1} | Y_{1:k-1}) dX_{k-1}, \end{aligned} \quad (3)$$

where the state PDF $P(X_k, X_{k-1} | Y_{1:k-1})$ specifies the conditional probability of $X_k \cap X_{k-1}$ given $Y_{1:k-1}$, and $P(X_k | X_{k-1}, Y_{1:k-1})$ specifies the conditional probability of X_k given $X_{k-1} \cap Y_{1:k-1}$.

2.2.2. Updating. The prior PDF $P(X_k | Y_{1:k-1})$ is updated using the observation likelihood PDF $g(Y_k | X_k)$ at time k , and the posterior PDF $P(X_k | Y_{1:k})$ of state X_k is obtained as

$$\begin{aligned} P(X_k | Y_{1:k}) &= \frac{P(Y_k | X_k, Y_{1:k-1}) P(X_k | Y_{1:k-1})}{P(Y_k | Y_{1:k-1})} \\ &= \frac{g(Y_k | X_k) P(X_k | Y_{1:k-1})}{\int g(Y_k | X_k) P(X_k | Y_{1:k-1}) dX_k}, \end{aligned} \quad (4)$$

where the observation PDF $P(Y_k | X_k, Y_{1:k-1})$ specifies the conditional probability of Y_k given $X_k \cap Y_{1:k-1}$, and $P(Y_k | Y_{1:k-1})$ specifies the conditional probability of Y_k given $Y_{1:k-1}$.

The system state PDF measure $\pi_{k:l|m}(dX_{k:l})$ is defined as

$$\pi_{k:l|m}(dX_{k:l}) \triangleq P(X_{k:l} | Y_{1:m}). \quad (5)$$

The joint posterior PDF measure $\pi_{0:k|k}(dX_{0:k})$ can be obtained by Bayes' theorem (Equations (3) and (4)):

$$\pi_{0:k|k-1}(dX_{0:k}) = \int_{\mathfrak{R}^{n \times k}} K(dX_k | X_{k-1}) \pi_{0:k-1|k-1}(dX_{0:k-1}) \text{ (prediction),} \quad (6)$$

$$\pi_{0:k|k}(dX_{0:k}) = \frac{g(Y_k | X_k) \pi_{0:k|k-1}(dX_{0:k})}{\int_{\mathfrak{R}^{n \times k}} g(Y_k | X_k) \pi_{0:k|k-1}(dX_{0:k})} \text{ (updating).} \quad (7)$$

The marginal posterior PDF measure $\pi_{k|k-1}(dX_k)$ is obtained by

$$\pi_{k|k-1}(dX_k) = \int_{\mathfrak{R}^{n \times k}} K(dX_k | X_{k-1}) \pi_{k-1|k-1}(dX_{k-1}) \text{ (prediction),} \quad (8)$$

$$\pi_{k|k}(dX_k) = \frac{g(Y_k | X_k) \pi_{k|k-1}(dX_k)}{\int_{\mathfrak{R}^{n \times k}} g(Y_k | X_k) \pi_{k|k-1}(dX_k)} \text{ (updating).} \quad (9)$$

Similarly, $x_{k:l} \triangleq (x_k, x_{k+1}, \dots, x_l)$ and $y_{k:l} \triangleq (y_k, y_{k+1}, \dots, y_l)$ are, respectively, defined as the sampling particle path information of the state process and observation likelihood from time k to l .

Definition 1. Suppose that ν is a probability measure and φ represents an arbitrary function, Θ and Ψ are arbitrary function variables, K is the PDF of the transfer kernel satisfying the Markov process, and the following calculation method

is defined:

$$\begin{aligned} (\nu, \varphi) &\triangleq \int \varphi \nu, \\ \nu K(\Theta) &\triangleq \int \nu(dx) K(\Theta | x), \\ K\varphi(x) &\triangleq \int K(d\Psi | x) \varphi(\Psi). \end{aligned} \quad (10)$$

According to the above symbols, for any function φ , Bayesian theory (prediction and updating processes) can be redefined, using Equation (8), as

$$\begin{aligned} (\pi_{k|k-1}, \varphi) &= \int_{\mathfrak{R}^{n_x}} p(x_k | y_{1:k-1}) \varphi(x_k) dx_k \\ &= \int_{\mathfrak{R}^{n_x}} \left[\int_{\mathfrak{R}^{n_x}} K(x_k | x_{k-1}) p(x_{k-1} | y_{1:k-1}) dx_{k-1} \right] \varphi(x_k) dx_k \\ &= \int_{\mathfrak{R}^{n_x}} p(x_{k-1} | y_{1:k-1}) \left[\int_{\mathfrak{R}^{n_x}} K(x_k | x_{k-1}) \varphi(x_k) dx_k \right] dx_{k-1} \\ &= (\pi_{k-1|k-1}, K\varphi) \text{ (prediction)}. \end{aligned} \quad (11)$$

From Equation (9), it is concluded that

$$\begin{aligned} (\pi_{k|k}, \varphi) &= \int_{\mathfrak{R}^{n_x}} p(x_k | y_{1:k}) \varphi(x_k) dx_k \\ &= \frac{\int_{\mathfrak{R}^{n_x}} g(y_k | x_k) p(x_k | y_{1:k-1}) \varphi(x_k) dx_k}{\int_{\mathfrak{R}^{n_x}} g(y_k | x_k) p(x_k | y_{1:k-1}) dx_k} \\ &= \frac{(\pi_{k|k-1}, g\varphi)}{(\pi_{k|k-1}, g)} \text{ (updating)}. \end{aligned} \quad (12)$$

Except for a small number of dynamic models, it is difficult to obtain an analytic solution in Bayesian theory (Equations (6)–(9), (11), and (12)) and the exact solution of the posterior probability for general nonlinear and non-Gaussian systems.

3. Particle Filter

To solve the complex problem in the above optimal Bayesian filtering algorithm, Monte Carlo sampling is used instead of an integral operation [25]. The idea is to use a discrete distribution with a series of random samples and their corresponding weights to approximate the posterior PDF measure $\pi_{k|k}$ and calculate the expected value of the samples to estimate the actual system state X_k . The importance PDF $q(x_k | y_{1:k})$ is generally used to represent the discrete distribution to obtain the sampling particle set $\{x_{0:k}^i : i = 1, 2, \dots, N\}$ to calculate the posterior empirical measure distribution $\pi_{k|k}^N$:

$$\pi_{k|k}^N(dx_k) \triangleq \frac{1}{N} \sum_{i=1}^N \delta_{x_k^i}(dx_k), \quad (13)$$

where $\delta_{x_k^i}(\cdot)$ is the Dirac delta function. With the sampling number $N \rightarrow \infty$, the empirical measure $\pi_{k|k}^N$ is infinitely close to the actual posterior PDF measure $\pi_{k|k}$.

3.1. Sequential Importance Resampling (SIR) Filter. Since the posterior PDF distribution $p(x_k | y_{1:k})$ is unknown, it is necessary to construct the importance PDF $q(x_k | x_{k-1}, y_{1:k})$ to satisfy the requirements of the Monte Carlo sampling method and make up for the shortcoming that sampling cannot be carried out in the target distribution, and

$$q(x_k | x_{k-1}, y_{1:k}) = K(x_k | x_{k-1}), \quad (14)$$

is typically selected during the SIR process.

Assuming that the posterior PDF measure $\pi_{k-1|k-1}^N(dx_{k-1})$ at time $k-1$ is known and the particle set is $\{x_k^i\}_{i=1}^N \sim \pi_{k-1|k-1}^N(dx_{k-1})K(dx_k | x_{k-1})$ at time k , the prediction measure $\pi_{k|k-1}^N$ of the prediction stage can be obtained as

$$\pi_{k|k-1}^N(dx_k) \triangleq \frac{1}{N} \sum_{i=1}^N \delta(x_k - x_k^i). \quad (15)$$

When the number of sample particles N is large enough, the prediction measure $\pi_{k|k-1}^N$ is infinitely close to the actual state $\pi_{k|k-1}$. The Monte Carlo approximate posterior measure $\pi_{k|k}^N$ is obtained by substituting the prediction measure $\pi_{k|k-1}^N$ into Equation (9):

$$\pi_{k|k}^N(dx_k) \triangleq \frac{g(y_k | x_k) \pi_{k|k-1}^N(dx_k)}{\int_{\mathfrak{R}^{n_x}} g(y_k | x_k) \pi_{k|k-1}^N(dx_k)} = \frac{\sum_{i=1}^N g(y_k | x_k^i) \delta_{x_k^i}(dx_k)}{\sum_{i=1}^N g(y_k | x_k^i)}. \quad (16)$$

The above formula is equivalent to

$$\pi_{k|k}^N(dx_k) = \sum_{i=1}^N \omega_k^i \delta_{x_k^i}(dx_k), \quad \sum_{i=1}^N \omega_k^i = 1, \quad (17)$$

where $\omega_k^i \propto g(y_k | x_k^i)$ is the weight of the importance PDF $q(x_k | y_{1:k})$ after normalization of all sampling particles $\{x_{0:k}^i\}_{i=1}^N$, and the posterior measure $\pi_{k|k}^N$ is the weighted sum of the Dirac delta function. The above process is called SIS filtering.

After several updating iterations, the weights of some particles in the SIS process may be small enough to ignore, which cannot be avoided due to the shortcomings of the algorithm. To overcome this, resampling is usually used to solve the degradation problem of the standard PF algorithm. By duplicating particles with higher weights and discarding those with smaller weights, the particle set $\{\tilde{x}_{0:k}^i : i = 1, 2, \dots, N\}$ is gathered in the high-probability posterior region to obtain the approximate value of the unweighted empirical

Step 1. Particle initialization
 At time $k = 0$, for $i = 1, 2, \dots, N$
 Sample $x_0^i \sim \pi_{0|0}(dx_0)$
 At time $k \geq 1$, for $i = 1, 2, \dots, N$
Step 2. Sequential importance sampling
 Predict $\pi_{k|k-1}^N$, sample $x_k^i \sim \pi_{k-1|k-1}^N K(dx_k)$
 Evaluate the normalized importance weights ω_k^i
 $\omega_k^i \propto g(y_k | x_k^i)$, and $\sum_{i=1}^N \omega_k^i = 1$
Step 3. Resampling strategy
 Calculate effective samples $N_{\text{eff}} = (\sum_{i=1}^N (\omega_k^i)^2)^{-1}$
 Compare resampling threshold N_{th}
 if $N_{\text{eff}} < N_{\text{th}}$
 $[\tilde{x}_k^i, \tilde{\omega}_k^i] = \text{RESAMPLE}[x_k^i, \omega_k^i]$, and $\tilde{\omega}_k^i = 1/N$
 else $[\tilde{x}_k^i, \tilde{\omega}_k^i] = [x_k^i, \omega_k^i]$
Step 4. State estimation
 $x_k = \sum_{i=1}^N \tilde{\omega}_k^i \tilde{x}_k^i$

ALGORITHM 1: Standard particle filter.

distribution measure $\tilde{\pi}_{k|k}^N$:

$$\tilde{\pi}_{k|k}^N = \frac{1}{N} \sum_{i=1}^N \delta_{\tilde{x}_k^i}(d\tilde{x}_k). \quad (18)$$

It can be inferred that the essence of resampling is realized using N sampling iterations in the empirical distribution measure $\pi_{k|k}^N$, and the new particle set obtained by this method approximates the actual posterior measure $\pi_{k|k}$. Common resampling methods are random, system, polynomial, and residual resampling. The process of the standard PF algorithm is shown as Algorithm 1.

3.2. Clustering Similarity Particle Filter (CSPF). The standard PF algorithm is simple in structure and easy to execute. Under the optimal estimation, the approximate estimated value of the algorithm converges to the actual state value. However, there are some issues in practical engineering applications.

3.2.1. Particle Degradation Phenomenon. The standard PF introduces the importance PDF distribution in the SIS process, which causes the variance of the particle weight to accumulate with each iteration. The importance weights corresponding to most particles tend to zero, resulting in a particle degradation phenomenon [26]. The above effects lead to a significant waste of computing resources, with the result that the approximate estimation cannot accurately describe the posterior distribution of the actual state. This degradation phenomenon cannot be avoided due to defects of the algorithm.

3.2.2. Particle Depletion Problem. A resampling strategy is an effective and important method to improve particle degradation. By resampling the discrete approximate posterior PDF distribution obtained by the importance sampling process,

samples with larger weights are duplicated many times under the guidance of the particle motion and the distribution of the state at the previous moment so that the number of effective particles increases and degradation is suppressed. However, resampling is likely to cause the abandonment or loss of some low-weight particles, which causes the resampled particles to prematurely move away from the actual state posterior region. This results in sample dilution [27] and eventually in the increase of state estimation variance, which greatly diminishes filtering performance.

In view of the above problems, our improved PF algorithm relies on the consistency principle of the spatial state trajectory [28]; i.e., the closer the state trajectory of a particle is to the actual state trajectory, the more likely the particle state represents the actual state. By using clustering similarity theory to measure the degree of trajectory consistency, the higher the degree of consistency similarity, the closer it is to the actual state, and the particle weights of the SIS process are updated to improve particle degradation. The improved algorithm abandons the resampling strategy, which can fundamentally eliminate the particle depletion problem.

The particle set $\{X_k^\bullet(i)\}_{i=1}^N = \{x_j^\circ(i) : j = k, \dots, k+L; x_j'(i) : j = k+L+1, \dots, k+L+l\}_{i=1}^N$ from time k to time $k+L+l$ is selected as the state trajectory at time k , where L and l are predefined constants. The original trajectory set $x_j^\circ(i)$ follows the SIS filtering process [15], and the modified trajectory set $x_j'(i)$ complies with the GPF prediction algorithm [18]. Because the actual state is unknown, the observation likelihood information is used to represent the consistency parameter of the state trajectory. Depending on the particle state trajectory, the corresponding observed likelihood trajectory set $\{Y_k^\bullet(i)\}_{i=1}^N = \{y_j^\circ(i) : j = k, \dots, k+L; y_j'(i) : j = k+L+1, \dots, k+L+l\}_{i=1}^N$ is determined as

$$Y_k^\bullet(i) = H(X_k^\bullet(i), v_k), \quad (19)$$

where the measurement noise is $v_k = 0$; i.e., the observation equation $H(\ast)$ is a known function determined by the specific research objects without noise interference. The observed likelihood trajectory corresponding to the actual state (reference trajectory) is $\{Y_k\} = \{y_j : j = k, \dots, k+L+l\}$. In this work, a clustering method using distance-based similarity is selected to analyze the trajectory consistency, and the distance similarity measurement $d_k^\bullet(i)$ [29] of the observed likelihood trajectories of the actual state and sampling particles is calculated as

$$\begin{aligned} d_k^\bullet(i) &= \text{dis}\left(\{Y_k\}, \{Y_k^\bullet(i)\}_{i=1}^N\right) \\ &= \sqrt{\sum_{j=k}^{k+L} |y_j - y_j^\circ(i)|^s + \sum_{j=k+L+1}^{k+L+l} |y_j - y_j'(i)|^s}, \end{aligned} \quad (20)$$

where $\text{dis}(\ast)$ is the distance similarity function, $\text{dis}(\ast) \geq 0$, and s is the measurement type parameter. To increase the

reliability of the algorithm, the distance similarity function is transformed to an exponential similarity function:

$$d_k(i) = e^{\lambda \times d_k^*(i)}, \quad i = 1, \dots, N, \quad (21)$$

where λ is the gradient factor. The importance weights $w_k^*(i)$ and $w_{k+L+l}^*(i)$ corresponding to times k and $k+L+l$ can be calculated as

$$w_k^*(i) = d_k^{-1}(i) \times p_v(Y_k - H(X_k^*, v_k)), \quad i = 1, \dots, N, \quad (22)$$

$$w_{k+L+l}^*(i) = d_k^{-1}(i) \times p_v(Y_{k+L+l} - H(X_{k+L+l}^*, v_{k+L+l})), \quad i = 1, \dots, N, \quad (23)$$

where $p_v(\cdot)$ is the PDF of the observed likelihood noise. Using the above algorithm, the original trajectory set $\{x_k^o(i)\}_{i=1}^N$ of the SIS process and the corresponding importance weights $\{w_k^*(i)\}_{i=1}^N$ at time k represent the posterior PDF $\pi_{k|k}^N$ of the system state, modified trajectory set $\{x'_{k+L+l}(i)\}_{i=1}^N$ of the GPF-predicted distribution, and corresponding importance weights $\{w_{k+L+l}^*(i)\}_{i=1}^N$ at time $k+L+l$, which can approximately represent the predicted PDF $\pi_{k+L+l|k}$. Therefore, the state estimate value x_k can be obtained by the filtering operation, and the state estimate x_{k+L+l} can be calculated by the prediction step. The implementation of the improved PF algorithm is as follows.

(1) *Estimation*. This step is consistent with the SIS(GPF) estimation process used to extract the particle distribution set $\{X_k^*(i)\}_{i=1}^N$.

(2) *Updating*. The weights $\{w_k^*(i)\}_{i=1}^N$ and $\{w_{k+L+l}^*(i)\}_{i=1}^N$ are determined and normalized to $\{w_k(i)\}_{i=1}^N$ and $\{w_{k+L+l}(i)\}_{i=1}^N$, respectively, to estimate and predict system states x_k and x_{k+L+l} :

$$x_k \approx \tilde{x}_k = \sum_{i=1}^N w_k(i) x_k^o(i) \text{ (filtering)}, \quad (24)$$

$$x_{k+L+l} \approx \tilde{x}_{k+L+l} = \sum_{i=1}^N w_{k+L+l}(i) x'_{k+L+l}(i) \text{ (prediction)}.$$

The above steps constitute an iterative process of the improved algorithm. Unlike the standard PF, this method uses an estimate-update-filter (prediction) process without resampling. The improved PF algorithm (CSPF) is shown as Algorithm 2.

4. Convergence Proof

The proposed algorithm is based on bootstrap filtering theory, and Bayesian state estimation can be realized by a weighted bootstrap method [15, 30]. It is assumed that the sampling particle set $\{X_k^*(i)\}_{i=1}^N$ is derived from a continuous

```

Step 1. Particle initialization
At time  $k = 0$ , for  $i = 1, 2, \dots, N$ 
Sample  $x_0^i \sim \pi_{0|0}(dx_0)$ 
At time  $k \geq 1$ , for  $i = 1, 2, \dots, N$ 
Step 2. Importance sampling
Predict  $\pi_{k|k-1}^N$ , sample  $x_k^i \sim K(x_k | x_{k-1}^i)$ 
Step 3. Similarity measurement
Particle state trajectories  $\{X_k^*(i)\}$ 
Draw  $x_j^o(i) \sim \text{SIS}, j = k, k+1, \dots, k+L$ 
 $x'_j(i) \sim \text{GPF}, j = k+L+1, \dots, k+L+l$ 
Calculate  $d_k^*(i) = \text{dis}(\{Y_k\}, \{Y_k^*(i)\}_{i=1}^N)$ 
Exponential transformation  $d_k(i) = e^{\lambda \times d_k^*(i)}$ 
Step 4. Recursive importance weights
Revised proposal distribution
 $q(x_k | x_{k-1}, y_k) = d_k(i) K(x_k | x_{k-1}^i)$ 
Update weights  $w_k^*(i) \propto d_k^{-1}(i) g(y_k | x_k^i)$ 
Normalize  $w_k(i) = w_k^*(i) / \sum_{i=1}^N w_k^*(i)$ 
Step 5. State estimation
 $x_k = \sum_{i=1}^N w_k(i) x_k^i$ 

```

ALGORITHM 2: Clustering similarity particle filter.

PDF, $R(x)$. The posterior PDF $\pi_{k|k}$ and $R(x)S(x)$ are proportional, and $S(x)$ is a known function. If the sample number $N \rightarrow \infty$, then the discrete distribution of particles composed of $\{X_k^*(i)\}_{i=1}^N$ and its corresponding weights $\{S(X_k^*(i)) / \sum_{i=1}^N S(X_k^*(i))\}_{i=1}^N$ can be regarded as approaching the actual posterior PDF $\pi_{k|k}$. Referring to Equation (16), the posterior PDF $\pi_{k|k}$ of system state x_k is proportional to the product of the observation likelihood function and prior PDF $\pi_{k|k-1}$, which can be equivalent to $R(x)$, and the weight $w_k^*(i)$ in Equation (22) can be regarded as the observation likelihood function equivalent to $S(x)$, which follows bootstrap filtering theory and is reasonable and effective.

4.1. *Convergence of the Improved Algorithm*. Suppose that the probability density measure space on set $E = \Gamma(\mathfrak{R}^{n_x})$ is the probability measure set on the largest-dimensional Euclidean space \mathfrak{R}^{n_x} with convergence topology and set (E, d) is the measure space. $(a_k)_{k=1}^\infty$ and $(b_k)_{k=1}^\infty$ are two continuous function sequences: $E \rightarrow E$. In the stochastic filtering setup, space E will be all probability measure spaces on n -dimensional Euclidean space \mathfrak{R}^{n_x} .

Definition 2. a_k and b_k , respectively, represent the mapping relationships of measure $\pi_{k-1|k-1} \rightarrow \pi_{k|k-1}$ and of measure $\pi_{k|k-1} \rightarrow \pi_{k|k}$. We define a_k as the mapping relation (prediction) satisfied on measure set $\Gamma(\mathfrak{R}^{n_x}) \rightarrow \Gamma(\mathfrak{R}^{n_x})$:

$$a_k(v)(dx_k) \triangleq vK(dx_k) = \int_{\mathfrak{R}^{n_x}} vK(dx_k | x_{k-1})(dx_{k-1}). \quad (25)$$

This holds for any measure v . Therefore, substituting the continuous function $\varphi \in \mathcal{C}_b(\mathfrak{R}^{n_x})$ in the prediction Equation

(11), we obtain

$$(a_k(\nu), \varphi) = \iint_{R^{n_x}} \nu K(dx_k | x_{k-1}) \varphi(x_k) (dx_{k-1}) = (\nu, K\varphi). \quad (26)$$

The prediction measure expression can be obtained as

$$\pi_{k|k-1} = a_k(\pi_{k-1|k-1}). \quad (27)$$

Definition 3. Referring to Equation (12), we define b_k as the mapping relation (updating) satisfied on measure set $\Gamma(\mathfrak{R}^{n_x}) \rightarrow \Gamma(\mathfrak{R}^{n_x})$:

$$(b_k(\nu), \varphi) = \frac{(\nu, g\varphi)}{(\nu, g)}. \quad (28)$$

The Bayesian filtering process can be expressed as

$$\pi_{k|k} = b_k(\pi_{k|k-1}) \triangleq b_k \odot a_k(\pi_{k-1|k-1}), \quad (29)$$

where the operator “ \odot ” represents the composite mapping function.

Definition 4. Setting h_k and $h_{1:k}$ as the conversion functions of measure $\pi_{k-1|k-1} \rightarrow \pi_{k|k}$ and of measure $\pi_{0|0} \rightarrow \pi_{k|k}$, respectively, the Bayesian filtering process can be expressed as

$$\begin{aligned} h_k &\triangleq b_k \odot a_k, \\ h_{1:k} &\triangleq h_k \odot h_{k-1} \odot \cdots \odot h_1. \end{aligned} \quad (30)$$

In an abstract environment, the PF algorithm uses the Monte Carlo method to solve a problem for which it is difficult to obtain the exact analytical integral solution in Bayesian theory. The principle is to generate a series of samples from the target distribution to approximately estimate the partial characteristics of the actual state, and the estimation result is only the expectation of a “good performance” function, which can be approximated as the average value:

$$E_\pi(\varphi(x)) \approx \frac{\varphi(x_1) + \varphi(x_2) + \cdots + \varphi(x_N)}{N} = \bar{E}_{\varphi, N}. \quad (31)$$

When $N \rightarrow \infty$, the estimated value $\bar{E}_{\varphi, N}$ converges to the expected value $E_\pi(\varphi(x))$. It can be assumed that

$$\begin{aligned} E_\pi(\varphi(x)) &= \mu \text{ (expectation),} \\ \text{Var}(\varphi(x)) &= \sigma^2 \text{ (variance),} \end{aligned} \quad (32)$$

where μ is the most basic digital feature to measure the centralized position or average level of a random variable x , and σ^2 is a numeric characteristic of the dispersion of the random variable x .

Based on the law of large numbers and the central limit theorem [31], it can be concluded that

$$\bar{E}_{\varphi, N} \rightarrow \mu, \quad (33)$$

$$\Pr \left\{ \lim_{N \rightarrow \infty} |\mu - \bar{E}_{\varphi, N}| = 0 \right\} = 1, \quad (34)$$

where $\Pr \{*\}$ is the probability function.

Therefore, for the analytical solution of the integral operation, the disturbance caused by the Monte Carlo sampling method is inevitable, mainly because the estimated value is based on a random and limited sample set. However, under the guarantee of the law of large numbers and the central limit theorem, when the number of particles tends to infinity, the disturbance is minimal and satisfies the following Gaussian distribution $N(*|*)$:

$$\bar{E}_{\varphi, N} \sim N(\bar{E}_{\varphi, N} | \mu, \sigma^2/N). \quad (35)$$

When $N \rightarrow \infty$, the state estimate $\bar{E}_{\varphi, N}$ converges to the real expected value μ , and the estimation variance decreases with the increase of the number of sample particles.

From the above analysis, it can be concluded that the particle filter is based on Bayesian filtering and can be combined with the Monte Carlo sampling method to generate a sampling disturbance function c^N [32]. The perturbation Equations (33) and (34) can be expressed as

$$\lim_{N \rightarrow \infty} c^N(\bar{E}_{\varphi, N}) = \mu. \quad (36)$$

The process formulas (29) and (30) of the particle filter algorithm can be expressed as

$$\pi_{k|k}^N = b_k \odot c^N \odot a_k(\pi_{k-1|k-1}^N) = h_k^N(\pi_{k-1|k-1}^N) = h_{1:k}^N(\kappa), \quad (37)$$

where κ is the initial value $\pi_{0|0}$. Our improved algorithm uses clustering analysis to measure the similarity of multistage measurement information [33] as the proposed distribution $q(x_k | x_{k-1}, y_k)$ to replace the prior PDF $p(x_k | x_{k-1})$ in the SIS process:

$$q(x_k | x_{k-1}, y_k) = d_k(i) K(dx_k | x_{k-1}). \quad (38)$$

The importance weight calculation is updated and modified as follows:

$$w_k(i) \propto g(y_k | x_k) d_k^{-1}(i) = \widehat{g}(y_k | x_k). \quad (39)$$

Substituting this in Equation (28), the updating formula of

the improved algorithm becomes

$$\left(\widehat{b}_k(v_N), \varphi\right) = \frac{\left(v_N, \widehat{g}\varphi\right)}{\left(v_N, \widehat{g}\right)}, \quad (40)$$

where \widehat{b}_k represents the mapping relationship of the improved algorithm measure $\pi_{k|k-1}^N \rightarrow \pi_{k|k}^N$ and Monte Carlo measure $\lim_{N \rightarrow \infty} v_N = v$. Referring to Equations (37) and (30), the improved PF can be expressed as

$$\widehat{\pi}_{k|k}^N = \widehat{b}_k \odot c^N \odot a_k \left(\pi_{k-1|k-1}^N\right) = \widehat{h}_k^N \left(\pi_{k-1|k-1}^N\right) = \widehat{h}_{1:k}^N(\kappa). \quad (41)$$

Theorem 5. *It is assumed that the state transition kernel function K satisfies the first-order Markov process, and the observation likelihood function g is continuous, bounded, and strictly positive in $x_k \in \mathfrak{R}^{n_x}$. Under the condition of the Monte Carlo sampling disturbance c^N , the improved PF algorithm measure $\widehat{\pi}_{k|k}^N$ converges to the theoretical value (actual state value) $\pi_{k|k}$ of Bayesian optimal filtering:*

$$\lim_{N \rightarrow \infty} \widehat{\pi}_{k|k}^N = \pi_{k|k}. \quad (42)$$

Proof. In the PF algorithm, the Monte Carlo sampling disturbance is random and uncertain. $c^{N,\vartheta}$ is set as a random disturbance, the sample number $N > 0$, and the independent variable $\vartheta \in \Omega$. For all measures $v \in \Gamma(\mathfrak{R}^{n_x})$,

$$c^{N,\vartheta}(v) = \frac{1}{N} \sum_{j=1}^N \delta\{V_j; \vartheta\}, \quad (43)$$

where $V_j : \Omega \rightarrow \mathfrak{R}^d$ is an i.i.d. random variable with measure v . According to the algorithm and the simplification of Equations (29) and (41), we can obtain

$$\pi_{k|k} = b_k \left(\pi_{k|k-1}\right) = b_k \odot a_k \left(\pi_{k-1|k-1}\right), \quad (44)$$

$$\widehat{\pi}_{k|k}^N = \widehat{b}_k \left(\widehat{\pi}_{k|k-1}^N\right) = \widehat{b}_k \odot c^N \odot a_k \left(\widehat{\pi}_{k-1|k-1}^N\right) = \widehat{b}_k \odot c^N(a_k). \quad (45)$$

At time k , the measure of the Bayesian prediction stage is a_k , and the sampling disturbance measure of the PF prediction stage is $c^N(a_k)$. Using the i.i.d. random variables $V_j, \forall \varphi \in \Lambda$,

and $\Lambda = \{\varphi_i\}_{i>0} \in C_b(\mathfrak{R}^{n_x})$, we can obtain

$$\begin{aligned} E\left[\left((c^N(a_k), \varphi_i) - (a_k, \varphi_i)\right)^4\right] &= \frac{1}{N^4} E\left[\left(\sum_{j=1}^N (\varphi_i(V_j) - (a_k, \varphi_i))\right)^4\right] \\ &= \frac{1}{N^4} \sum_{j=1}^N E\left[\left(\varphi_i(V_j) - (a_k, \varphi_i)\right)^4\right] \\ &= \frac{4}{N^4} \sum_{j_1, j_2=1, j_1 \neq j_2}^N E\left[\left(\varphi_i(V_{j_1}) - (a_k, \varphi_i)\right)^2 \left(\varphi_i(V_{j_2}) - (a_k, \varphi_i)\right)^2\right] \\ &\leq \frac{N\|2\varphi_i\|^4 + 3N(N-1)\|2\varphi_i\|^4}{N^4} \leq \frac{48\|\varphi_i\|^4}{N^2}, \end{aligned} \quad (46)$$

where $E[*]$ is the solution function of set expectation, $\|*\|$ represents the supremum norm in the domain $C_b(\mathfrak{R}^{n_x})$, and $\|\varphi\| \triangleq \sup_{x \in \mathfrak{R}^{n_x}} |\varphi(x)|$. The summed expectation of the number of sampled particles from 1 to ∞ is

$$E\left[\sum_{N=1}^{\infty} \left((c^N(a_k), \varphi_i) - (a_k, \varphi_i)\right)^4\right] \leq 48\|\varphi_i\|^4 \sum_{N=1}^{\infty} \frac{1}{N^2} < \infty. \quad (47)$$

Hence,

$$\sum_{N=1}^{\infty} \left((c^N(a_k), \varphi_i) - (a_k, \varphi_i)\right)^4 < \infty. \quad (48)$$

This implies that for the prediction stage, the measure at a certain time can be expressed as

$$\lim_{N \rightarrow \infty} \left((c^N(a_k), \varphi_i) - (a_k, \varphi_i)\right) = 0. \quad (49)$$

Referring to Equation (40), for $v \in \Gamma(\mathfrak{R}^{n_x})$ and any function φ , the updating stage measure can be obtained as

$$\begin{aligned} \lim_{N \rightarrow \infty} \left(\widehat{b}_k(v_N), \varphi\right) &= \frac{\lim_{N \rightarrow \infty} \left(v_N, \widehat{g}\varphi\right)}{\lim_{N \rightarrow \infty} \left(v_N, \widehat{g}\right)} \\ &= \frac{\lim_{N \rightarrow \infty} d_k^{-1}(i)(v_N, g\varphi)}{\lim_{N \rightarrow \infty} d_k^{-1}(i)(v_N, g)} \\ &= \frac{(v, g\varphi)}{(v, g)} = (b_k(v), \varphi). \end{aligned} \quad (50)$$

This result is compared with Equations (28) and (50), and it is concluded that the improved algorithm has the same measures as the Bayesian filter in the updating stage, i.e.,

$$\lim_{N \rightarrow \infty} \widehat{b}_k(v_N) = b_k(v). \quad (51)$$

Combining Equations (44), (45), (49), and (51), we can

obtain

$$\lim_{N \rightarrow \infty} \widehat{\pi}_{k|k}^N = \pi_{k|k}^N. \quad (52)$$

Therefore, the improved PF algorithm (CSPF) based on the clustering similarity of the state trajectories still converges to the actual state under the interference of Monte Carlo sampling disturbances. Theorem 5 is proved. \square

4.2. Convergence of the Mean Squared Error of Results. Combined with the conditions and conclusions of Section 4.1, we analyze the convergence of the results by calculating the boundary of the MSE of the improved algorithm [34]. We demonstrate that the convergence of the reasoning process is related to the number of sample particles at each stage of the algorithm. Suppose that in the neighborhood of $E = \Gamma(\mathfrak{R}^{n_x})$, $(\mathbf{v}_N^w)_{N=1}^\infty$ represents a measure sequence of random probability and satisfies $\lim_{N \rightarrow \infty} \mathbf{v}_N = \mathbf{v}$. For any function $\varphi \in C_b(\mathfrak{R}^{n_x})$, it can be obtained from Theorem 5 that

$$\lim_{N \rightarrow \infty} E[(\mathbf{v}_N, \varphi) - (\mathbf{v}, \varphi)]^2 = 0. \quad (53)$$

Theorem 6. *It is assumed that the state transition kernel function K satisfies the first-order Markov process, and the observation likelihood function g is continuous, bounded, and strictly positive in $x_k \in \mathfrak{R}^{n_x}$. For any function φ , there must be a real constant $C_{k|k}$ satisfying*

$$E\left[\left(\widehat{\pi}_{k|k}^N, \varphi\right) - \left(\pi_{k|k}, \varphi\right)\right]^2 \leq C_{k|k} \frac{\|\varphi\|^2}{N}, \quad (54)$$

where $\|\varphi\| \triangleq \sup_{x \in \mathfrak{R}^{n_x}} |\varphi(x)|$.

Proof. According to the improved PF algorithm, the proof is divided into prediction and updating parts. \square

Lemma 7. *Refer to the prediction stage in Algorithm 2 (steps 1 and 2) and assume that the conditions set by Theorem 6 are met. When $k \geq 0$, there must be a real constant $C_{k|k-1}$, and the prediction measure $\pi_{k|k-1}^N$ satisfies*

$$E\left[\left(\left(\pi_{k|k-1}^N, \varphi\right) - \left(\pi_{k|k-1}, \varphi\right)\right)^2\right] \leq C_{k|k-1} \frac{\|\varphi\|^2}{N}. \quad (55)$$

We use induction to complete the proof. When time $k = 0$, for any function φ , there must be a real constant $C_{0|0}$ satisfying

$$E\left[\left(\left(\pi_{0|0}^N, \varphi\right) - \left(\pi_{0|0}, \varphi\right)\right)^2\right] \leq C_{0|0} \frac{\|\varphi\|^2}{N}. \quad (56)$$

From step 1 in Algorithm 2, when $k = 0$, N i.i.d. particles are sampled from the prior PDF measure $\pi_{0|0}$, $x_0^i \sim \pi_{0|0}(dx_0)$. Using the Marcinkiewicz–Zygmund inequality [35], we

obtain

$$\begin{aligned} E\left[\left(\left(\pi_{0|0}^N, \varphi\right) - \left(\pi_{0|0}, \varphi\right)\right)^2\right] &= E\left[\left(\frac{1}{N} \sum_{i=1}^N \varphi(x_i) - \left(\pi_{0|0}, \varphi\right)\right)^2\right] \\ &= \frac{1}{N^2} E\left[\left(\sum_{i=1}^N \varphi(x_i) - N\left(\pi_{0|0}, \varphi\right)\right)^2\right] \\ &\leq C_{0|0} \frac{\|\varphi\|^2}{N}. \end{aligned} \quad (57)$$

Thus, when $k = 0$, Equation (55) is proved, and $\pi_{0|0}^N$ converges to $\pi_{0|0}$.

At time $k - 1$, for any function φ , there must be a real constant $C_{k-1|k-1}$ satisfying

$$E\left[\left(\left(\pi_{k-1|k-1}^N, \varphi\right) - \left(\pi_{k-1|k-1}, \varphi\right)\right)^2\right] \leq C_{k-1|k-1} \frac{\|\varphi\|^2}{N}. \quad (58)$$

At time k , step 2 in Algorithm 2 can be derived:

$$E\left[\left(\left(\pi_{k|k-1}^N, \varphi\right) - \left(\pi_{k|k-1}, \varphi\right)\right)^2\right] \leq C_{k|k-1} \frac{\|\varphi\|^2}{N}. \quad (59)$$

By substituting the prediction stage formula (11) in the above formula, we obtain

$$\begin{aligned} \left| \left(\pi_{k|k-1}^N, \varphi\right) - \left(\pi_{k|k-1}, \varphi\right) \right| &= \left| \left(\pi_{k|k-1}^N, \varphi\right) - \left(\pi_{k-1|k-1}^N, K\varphi\right) \right. \\ &\quad \left. + \left(\pi_{k-1|k-1}^N, K\varphi\right) - \left(\pi_{k-1|k-1}, K\varphi\right) \right| \\ &\leq \left| \left(\pi_{k|k-1}^N, \varphi\right) - \left(\pi_{k-1|k-1}^N, K\varphi\right) \right| \\ &\quad + \left| \left(\pi_{k-1|k-1}^N, K\varphi\right) - \left(\pi_{k-1|k-1}, K\varphi\right) \right|. \end{aligned} \quad (60)$$

Setting ζ_{k-1} as the σ -field generated by particle set $\{x_k^i\}_{i=1}^N$ and combining this with the Monte Carlo method, we obtain

$$E\left[\left(\pi_{k|k-1}^N, \varphi\right) | \zeta_{k-1}\right] = \left(\pi_{k-1|k-1}^N, K\varphi\right). \quad (61)$$

Substituting $\|k\varphi\| \leq \|\varphi\|$ in the above formula,

$$\begin{aligned} E\left[\left(\left(\pi_{k|k-1}^N, \varphi\right) - \left(\pi_{k-1|k-1}^N, K\varphi\right)\right)^2\right] &= E\left[\left(\left(\pi_{k|k-1}^N, \varphi\right) - E\left[\left(\pi_{k|k-1}^N, \varphi\right) | \zeta_{k-1}\right]\right)^2 | \zeta_{k-1}\right] \\ &= E\left[\left(\pi_{k|k-1}^N, \varphi\right)^2 | \zeta_{k-1}\right] - 2E^2\left[\left(\pi_{k|k-1}^N, \varphi\right) | \zeta_{k-1}\right] \\ &\quad + E^2\left[\left(\pi_{k|k-1}^N, \varphi\right) | \zeta_{k-1}\right] = E\left[\left(\pi_{k|k-1}^N, \varphi\right)^2 | \zeta_{k-1}\right] \\ &\quad - E^2\left[\left(\pi_{k|k-1}^N, \varphi\right) | \zeta_{k-1}\right] \leq \frac{\|\varphi\|^2}{N}. \end{aligned} \quad (62)$$

Referring to Equation (58), there must be a real constant:

$$E \left[\left(\left(\pi_{k-1|k-1}^N, K\varphi \right) - \left(\pi_{k-1|k-1}, K\varphi \right) \right)^2 \right] \leq C_{k-1|k-1} \frac{\|K\varphi\|^2}{N} \leq C_{k-1|k-1} \frac{\|\varphi\|^2}{N}. \quad (63)$$

Using the Minkowski inequality, we obtain

$$\begin{aligned} E \left[\left(\left(\pi_{k|k}^N, \varphi \right) - \left(\pi_{k|k-1}, \varphi \right) \right)^2 \right]^{1/2} &\leq E \left[\left(\left(\pi_{k|k-1}^N, \varphi \right) - \left(\pi_{k-1|k-1}^N, K\varphi \right) \right)^2 \right]^{1/2} \\ &\quad + E \left[\left(\left(\pi_{k-1|k-1}^N, K\varphi \right) - \left(\pi_{k-1|k-1}, K\varphi \right) \right)^2 \right]^{1/2} \\ &\leq \sqrt{C_{k|k-1}} \frac{\|\varphi\|}{\sqrt{N}}, \end{aligned} \quad (64)$$

where $C_{k|k-1} = (1 + \sqrt{C_{k-1|k-1}})^2$. Lemma 7 is proved.

Lemma 8. Refer to the updating stage in Algorithm 2 (steps 3–5) and assume that the conditions set by Theorem 6 and Lemma 7 are met. When $k \geq 0$, for any function φ , there must be a real constant $C_{k|k}$, and the prediction measure $\widehat{\pi}_{k|k}^N$ sat-

isfies

$$E \left[\left(\pi_{k|k-1}^N, \varphi \right) | \zeta_{k-1} \right] = \left(\pi_{k-1|k-1}^N, K\varphi \right), \quad (65)$$

where $\|\varphi\| \triangleq \sup_{x \in \mathfrak{R}^n} |\varphi(x)|$, setting ζ_{k-1} as the σ -field generated by particle set $\{x_k^i\}_{i=1}^N$.

Combined with the updating stage, we can obtain the following using Equation (12):

$$\begin{aligned} \left(\widehat{\pi}_{k|k}^N, \varphi \right) - \left(\pi_{k|k}, \varphi \right) &= \frac{\left(\pi_{k|k-1}^N, \widehat{g}\varphi \right)}{\left(\pi_{k|k-1}^N, \widehat{g} \right)} - \frac{\left(\pi_{k|k-1}, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} \\ &= \frac{\left(\pi_{k|k-1}^N, \widehat{g}\varphi \right)}{\left(\pi_{k|k-1}^N, \widehat{g} \right)} - \frac{\left(\pi_{k|k-1}^N, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} \\ &\quad + \frac{\left(\pi_{k|k-1}^N, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} - \frac{\left(\pi_{k|k-1}, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)}. \end{aligned} \quad (66)$$

Substituting Equation (39) in this result yields

$$\begin{aligned} \left| \frac{\left(\pi_{k|k-1}^N, \widehat{g}\varphi \right)}{\left(\pi_{k|k-1}^N, \widehat{g} \right)} - \frac{\left(\pi_{k|k-1}^N, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} \right| &= \left| \frac{\left(\pi_{k|k-1}^N, \widehat{g}\varphi \right) \left(\pi_{k|k-1}, g \right) - \left(\pi_{k|k-1}^N, g\varphi \right) \left(\pi_{k|k-1}^N, \widehat{g} \right)}{\left(\pi_{k|k-1}^N, \widehat{g} \right) \left(\pi_{k|k-1}, g \right)} \right| \\ &= \left| \frac{\left(\pi_{k|k-1}^N, d_k^{-1} g\varphi \right) \left(\pi_{k|k-1}, g \right) - \left(\pi_{k|k-1}^N, g\varphi \right) \left(\pi_{k|k-1}^N, d_k^{-1} g \right)}{\left(\pi_{k|k-1}^N, d_k^{-1} g \right) \left(\pi_{k|k-1}, g \right)} \right| \\ &= \frac{\left(\pi_{k|k-1}^N, g\varphi \right)}{\left(\pi_{k|k-1}^N, g \right) \left(\pi_{k|k-1}, g \right)} \left| \left(\pi_{k|k-1}, g \right) - \left(\pi_{k|k-1}^N, g \right) \right| \\ &\leq \frac{\|\varphi\|}{\left(\pi_{k|k-1}, g \right)} \left| \left(\pi_{k|k-1}, g \right) - \left(\pi_{k|k-1}^N, g \right) \right|. \end{aligned} \quad (67)$$

Similarly, using the Minkowski inequality and Equation (64), we obtain

where $(2\|g\|/(\pi_{k|k-1}, g))\sqrt{C_{k|k-1}} = \sqrt{C_{k|k}}$. Lemma 8 is proved, and the MSE of the improved PF algorithm is convergent.

5. Practical Applications

The factors that affect the performance of the state estimation method based on the PF are mainly determined by the construction of the spatial state model and the quality of the algorithm. The accurate architecture of the spatial state model is

mainly realized by the reasonable selection of prior knowledge of the physical mechanism, as reflected in empirical physical equations, scientific parameter identification and training, multifeature search, and optimization of noise distribution. The method's advantages and disadvantages are limited by the particle degradation caused by an unreasonable proposed distribution and particle depletion caused by the resampling strategy. Therefore, to improve and obtain the best performance of state estimation methods for a fair comparison depends on the following. (a) The architecture of the spatial state model is

consistent. The same empirical physical equation and noise distribution are used in the application cases. The state tracking

where the measurement noise $v_k \sim N(0, R_k)$, and $R_k = 1$. The initial state was $x_0 = 0.1$, $\alpha = 0.5$, $\beta = 25$, $\gamma = 8$, $\delta = 1.2$, and ε

$$\begin{aligned}
E \left[\left(\left(\widehat{\pi}_{k|k}^N, \varphi \right) - \left(\pi_{k|k}, \varphi \right) \right)^2 \right]^{1/2} &\leq E \left[\left(\frac{\left(\pi_{k|k-1}^N, \widehat{g}\varphi \right)}{\left(\pi_{k|k-1}^N, \widehat{g} \right)} - \frac{\left(\pi_{k|k-1}^N, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} \right)^2 \right]^{1/2} + E \left[\left(\frac{\left(\pi_{k|k-1}^N, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} - \frac{\left(\pi_{k|k-1}, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} \right)^2 \right]^{1/2} \\
&\leq E \left[\left(\frac{\left(\pi_{k|k-1}^N, \widehat{g}\varphi \right)}{\left(\pi_{k|k-1}^N, \widehat{g} \right)} - \frac{\left(\pi_{k|k-1}^N, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} \right)^2 \right]^{1/2} + E \left[\left(\frac{\left(\pi_{k|k-1}^N, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} - \frac{\left(\pi_{k|k-1}, g\varphi \right)}{\left(\pi_{k|k-1}, g \right)} \right)^2 \right]^{1/2} \\
&\leq \frac{\|\varphi\|}{\left(\pi_{k|k-1}, g \right)} E \left[\left(\left(\pi_{k|k-1}, g \right) - \left(\pi_{k|k-1}^N, g \right) \right)^2 \right]^{1/2} + \frac{1}{\left(\pi_{k|k-1}, g \right)} E \left[\left(\left(\pi_{k|k-1}^N, g\varphi \right) - \left(\pi_{k|k-1}, g\varphi \right) \right)^2 \right]^{1/2} \\
&\leq \frac{2\|\varphi\|}{\left(\pi_{k|k-1}, g \right)} \sqrt{C_{k|k-1}} \frac{\|\varphi\|}{\sqrt{N}} \leq \sqrt{C_{k|k}} \frac{\|\varphi\|}{\sqrt{N}},
\end{aligned} \tag{68}$$

ability of each method is fully used, and the model parameter identification is trained. (b) The advantages of different methods are fully exploited. Commonly used and effective strategies are random, polynomial, system, and residual resampling. The applicability and accuracy of resampling strategies vary for different complex nonlinear systems. Considering the advantages of the above state estimation methods, according to the application case studied in this paper, through the analysis of simulation test results, it is concluded that the system resampling strategy performs best.

Therefore, based on the performance improvements of the method, the improved method can be flexibly applied to practical application fields to solve engineering problems on the premise of application scenario guarantees, such as the accurate architecture of the state space model for the research object, scientific identification and training of physical parameters, multi-feature search, and optimization of noise distribution. We compared the clustering similarity particle filter (CSPF) to several traditional filtering algorithms on two examples. All methods were implemented in MATLAB, and the root mean squared error (RMSE) was used to evaluate accuracy.

Example 1. The advantages of the improved algorithm can be seen in both the Gaussian and non-Gaussian nonlinear systems. A typical univariate nonstationary growth model (UNGM) [36] with highly nonlinear and bimodal characteristics of the state distribution is selected, which is widely used in social and economic fields such as the evaluation of urban development and short-term predictions of insurance stocks and bank interest. The following model was selected to verify the effectiveness of the CSPF algorithm:

$$\begin{aligned}
x_k &= \alpha x_{k-1} + \beta \frac{x_{k-1}}{1 + x_{k-1}^2} + \gamma \cos(\delta * (k-1)) + \omega_k, \\
y_k &= \frac{x_k^2}{\varepsilon} + v_k,
\end{aligned} \tag{69}$$

= 20. The number of particles was $N = 100$. The simulation steps were $k = 1, 2, \dots, T$, where $T = 5000$. The correlation coefficients were $L = 2$ and $l = 1$, and the gradient factor was $\lambda = 1.3$. SIR, APF, GPF, and the proposed algorithm were implemented in MATLAB, and 100 trials were conducted. The computer processor speed was 2.7 GHz, and the memory capacity was 8 GB. The RMSE,

$$\text{RMSE} = \sqrt{\frac{1}{T} \sum_{k=1}^T (x_k^{\text{tur}} - x_k^{\text{est}})^2}, \tag{70}$$

was used to evaluate the performance of the algorithms, where x_k^{tur} and x_k^{est} are, respectively, the real and state estimation values of the system.

The basic framework of the particle filter has state estimation and update parts. State estimation realizes state prediction of the current time based on the state prediction value of the previous time and the system state shift equation (including shift noise). Because the state variables are inevitably disturbed by various noises in the working environment, it is necessary to modify the estimated values based on actual observations to be as close to the real state values as possible so as to realize the update process of state estimation according to the observation values and measurement noise. Furthermore, the purpose of constructing the probability density function of the unknown system state using prior knowledge and actual observation data is realized. It can be seen that the shift and measurement noise will directly and greatly affect the nonlinear estimation method. This paper takes this case as an example, setting different kinds of shift noise to analyze the performance level of the nonlinear estimation methods. The two types of shift noise were Gaussian noise $\omega_{k_G} \sim N(\omega_k; 0, 1)$ and non-Gaussian noise $\omega_{k_NG} \sim \xi N(\omega_k; 0, \sigma_{\omega_1}^2) + (1 - \xi)N(\omega_k; 0, \sigma_{\omega_2}^2)$, where $\xi = 0.8$, $\sigma_{\omega_1}^2 = 1$, and $\sigma_{\omega_2}^2 = 10$. The non-Gaussian noise ω_{k_NG} was a heavy-

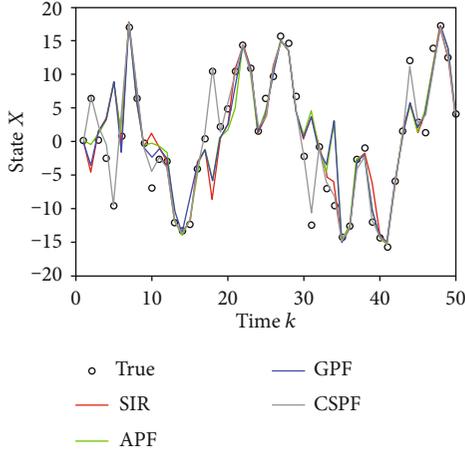


FIGURE 1: Comparison of state estimation based on different algorithms in the Gaussian noise environment. Note: SIR=sequential importance resampling; APF=auxiliary particle filter; GPF=Gaussian particle filter; CSPF=clustering similarity particle filter.

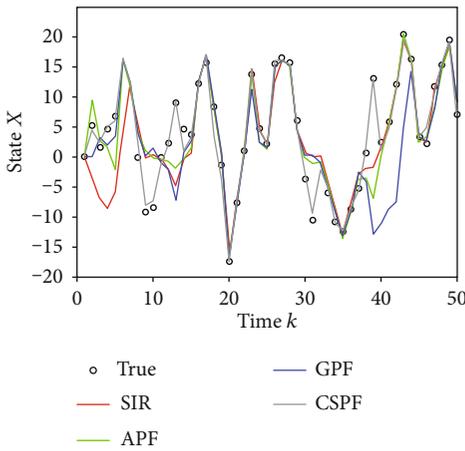


FIGURE 2: Comparison of state estimation based on different algorithms in the non-Gaussian noise environment. Note: SIR=sequential importance resampling; APF=auxiliary particle filter; GPF=Gaussian particle filter; CSPF=clustering similarity particle filter.

tailed distribution, and the acceptance-rejection sampling procedure used in Monte Carlo simulations [25] was adopted to achieve noise sampling with a confidence level of 97.5%. The simulation results are shown in Figures 1–3 and Tables 1 and 2.

The consistency of the state trajectory of the improved PF algorithm (clustering similarity particle filter (CSPF)) was measured by the Euclidean spatial distance (ECSPF), $s = 2$, and Chebyshev spatial distance (CCSPF), $s \rightarrow \infty$. Figures 1 and 2 compare various PF algorithms for a single operation of the nonlinear system in Gaussian and non-Gaussian noise environments (the improved algorithm is represented by ECSPF), which can directly evaluate the effect of state estimation. Figure 3 compares the RMSE of the simulation results

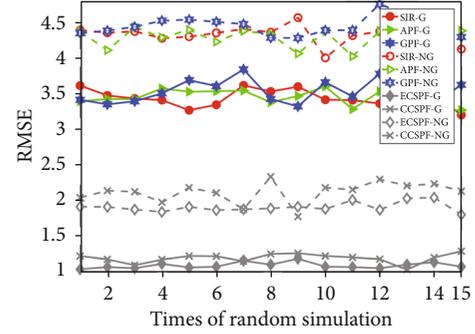


FIGURE 3: RMSE of SIR, APF, GPF, ECSPF, and CCSPF for 15 random simulation runs. Note: SIR=sequential importance resampling; APF=auxiliary particle filter; GPF=Gaussian particle filter; ECSPF=Euclidean distance clustering similarity particle filter; CCSPF=Chebyshev distance clustering similarity particle filter.

after 15 simulation runs. Algorithm labels ending in the letter G are denoted by solid curves and correspond to Gaussian system noise ω_{k-G} , and those ending in the letters NG are denoted by dotted curves and correspond to non-Gaussian system noise ω_{k-NG} . Tables 1 and 2 show the RMSE mean and variance of state estimation results after 100 simulation runs to compare the accuracy of the improved algorithm and other algorithms.

Figures 1 and 2 show that the state estimation accuracy of CSPF was significantly better than that of the other algorithms in the two noise environments, and the fluctuation trends of the CSPF estimated curve at any time continued moving and approaching the fluctuation trends of the actual state curve. By contrast, the other algorithms were not consistent at each time point. As shown in Figure 3 and Tables 1 and 2, the RMSE means and variances of the two improved algorithms were lower than those of SIR, APF, and GPF. For example, the prediction accuracy (variance) of CSPF was improved by 65%–69% (67%–89%) in the Gaussian noise environment and by 52%–57% (14%–54%) in the non-Gaussian noise environment. Therefore, the improved algorithm had greater accuracy and more stability, and the performance of ECSPF was particularly significant.

Under the same running time, the accuracy of the state estimation of the different algorithms was determined by adjusting the number of sampling particles N and the order of the state trajectory. From this, the efficiency of the improved algorithm was verified by comparing the corresponding operational cost (see Table 3 for details). In Gaussian and non-Gaussian noise environments, the improved algorithm with Euclidean and Chebyshev distance similarity measures was compared with SIR, APF, and GPF. For the same operational cost, i.e., for the same computing and simulation times, the improved algorithm not only used the fewest sampling particles but also had significantly better accuracy than the other three algorithms. The RMSE was reduced by factors of about 3 and 2 in Gaussian and non-Gaussian noise environments, respectively. This verified that

TABLE 1: Mean and variance of RMSE for five nonlinear algorithms running 100 times under both the Gaussian and non-Gaussian noise models.

Filter estimation	SIR	APF	GPF	ECSPF	CCSPF
RMSE mean (Gaussian)	3.4771	3.4368	3.5677	1.1003	1.2027
RMSE variance (Gaussian)	0.0171	0.0135	0.0221	0.0025	0.0045
RMSE mean (non-Gaussian)	4.3332	4.3939	4.4726	1.9273	2.0941
RMSE variance (non-Gaussian)	0.0222	0.0155	0.0235	0.0108	0.0133

Note: SIR = sequential importance resampling; APF = auxiliary particle filter; GPF = Gaussian particle filter; ECSPF = Euclidean distance clustering similarity particle filter; CCSPF = Chebyshev distance clustering similarity particle filter; RMSE = root mean squared error.

TABLE 2: Accuracy improvement percentage of mean and variance of RMSE for five nonlinear algorithms running 100 times under both the Gaussian and non-Gaussian noise models.

Filter	RMSE	Gaussian noise			Non-Gaussian noise		
		SIR	APF	GPF	SIR	APF	GPF
ECSPF	Mean	68.4%	68.0%	69.2%	55.5%	56.1%	56.9%
	Variance	85.4%	81.5%	88.7%	51.4%	30.3%	54.0%
CCSPF	Mean	65.4%	65.0%	66.3%	51.7%	52.3%	53.2%
	Variance	73.7%	66.7%	79.6%	40.1%	14.2%	43.4%

Note: SIR = sequential importance resampling; APF = auxiliary particle filter; GPF = Gaussian particle filter; ECSPF = Euclidean distance clustering similarity particle filter; CCSPF = Chebyshev distance clustering similarity particle filter; RMSE = root mean squared error.

the improved algorithm had higher operational efficiency and substantial advantages in computation time.

Example 2. We assessed the health status of a lithium-ion battery. Through the state tracking and capacity training of historical samples, the physical model parameters of the empirical degradation and distribution information of the noise were identified and optimized [3]. Different nonlinear methods were used to establish state tracking and remaining useful life (RUL) prediction models for the battery. The performances of various algorithms were evaluated based on the state tracking effect and prediction accuracy.

In order to solve the problem regarding battery health assessment, it is particularly important to ensure the accurate structure of the decay physical model (state space model) and scientific parameter identification; this is done under the premise of the reasonable selection of the state tracking training set (noise distribution) and initial parameters. In this paper, parameter identification is carried out based on the attenuation information of state tracking historical samples to construct the RUL prediction model [37]. Since the accuracy of state tracking and RUL prediction largely depends on the physical model of the battery capacity degradation, the development of the model requires physical knowledge of the system [38]. This is usually represented by information collected by sensors, including battery parameters (e.g., charge-discharge voltage and current, power, electrochemical impedance spectroscopy, frequency, and temperature), to build an equivalent circuit model [39] to characterize the degradation trend of battery capacity, as shown in Figure 4, which includes the open-circuit voltage (OCV), electrolyte resistance R_E , polarization current I_R , double-layer voltage

TABLE 3: Computation time analysis for five nonlinear algorithms running 100 times under Gaussian and non-Gaussian noise models.

Filter	N	Gaussian noise ($T = 5000$)		Filter	N	Non-Gaussian noise ($T = 1000$)	
		RMSE mean	AVG time (s)			RMSE mean	AVG time (s)
SIR	500	3.2556	1.5	SIR	320	4.0904	2.3
APF	250	3.3211	1.5	APF	160	4.2697	2.3
GPF	800	3.3093	1.5	GPF	340	4.1536	2.3
ECSPF	100	1.1003	1.5	CCSPF	100	2.1215	2.3

Note: AVG = average; RMSE = root mean squared error; SIR = sequential importance resampling; APF = auxiliary particle filter; GPF = Gaussian particle filter; ECSPF = Euclidean distance clustering similarity particle filter; CCSPF = Chebyshev distance clustering similarity particle filter.

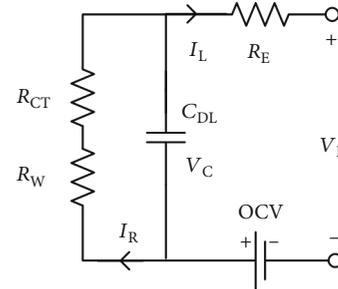


FIGURE 4: Battery equivalent circuit model.

V_C , capacitance C_{DL} , charge transfer resistance R_{CT} , Warburg impedance R_W , load current I_L , and terminal voltage V_L .

Relying on the attenuation mechanism of the electrochemical characteristics, the relationship between capacity degradation and internal impedance of the battery was determined using statistical regression. The simulated attenuation characteristic of the impedance increased with the number of charge-discharge cycles to obtain a double-exponential empirical degradation model [40, 41],

$$C_{ap} = a * \exp(b * k) + c * \exp(d * k), \quad (71)$$

where C_{ap} is the battery capacity ($A \cdot h$) and k is the number of charge-discharge cycles. The unknown model parameters a and c are related to the battery impedance, b and d are related to the rate of capacity degradation, and $\exp(*)$ is the exponential function.

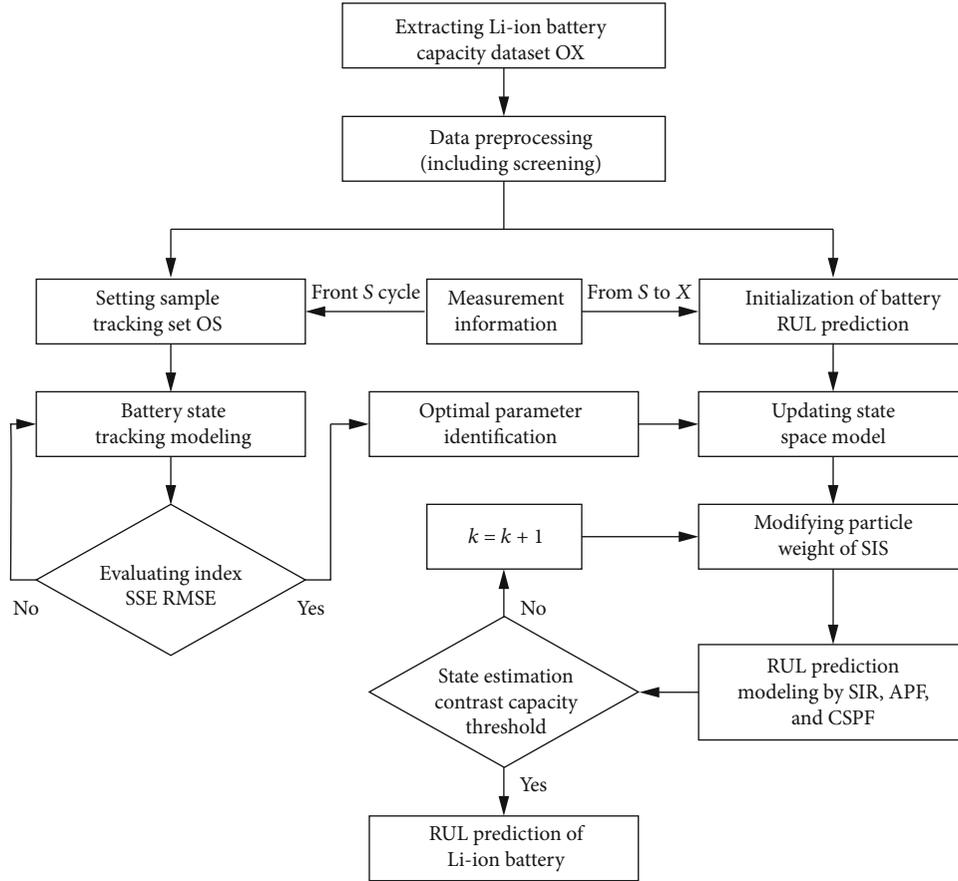


FIGURE 5: Battery state tracking and remaining useful life (RUL) prediction method for the Li-ion battery based on different algorithms.

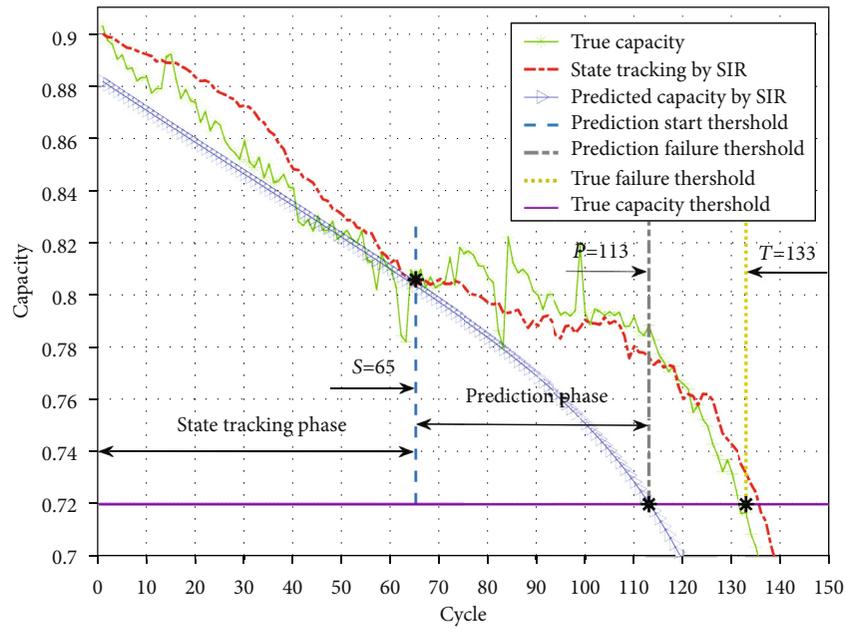
TABLE 4: Experimental test conditions of CALCE dataset M8.

Battery ID	Experimental platform	Temperature	Charging	End condition	Discharge	End condition
M8	Arbin BT2000	20-25°C	Constant current/voltage	4.2 V/0.05 A	Constant current	2.7 V

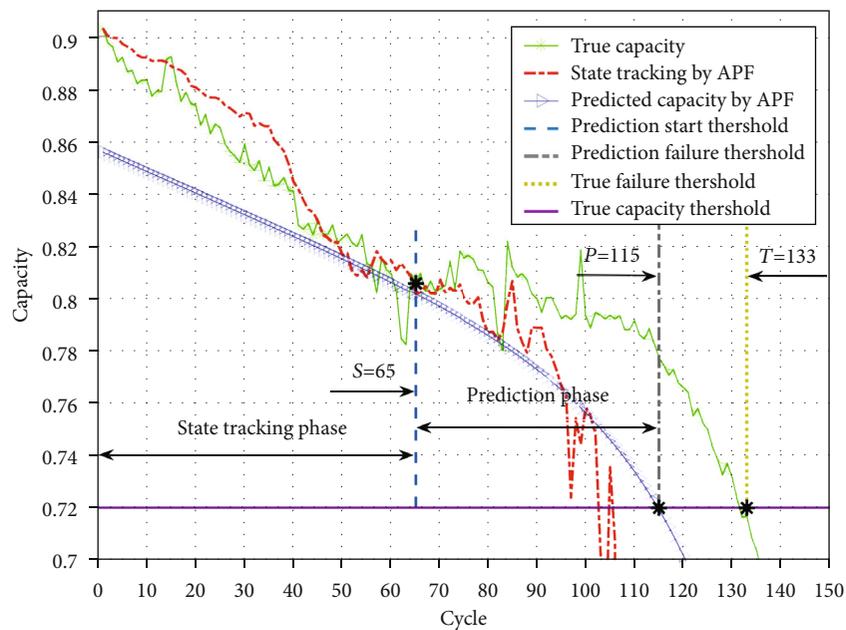
A tracking and training model was constructed using historical samples of the battery capacity to estimate the model parameters of the empirical degradation in real time and optimize the multicharacteristic noise [42]. The identified physical model effectively converged to the gradual trend of actual battery degradation, which provided reasonable identification parameters and effective initial values to establish an RUL prediction model. This enabled the updating of the state space model composed of the state transition equation (physical model of the battery system), which can represent the recursion law of the system, and the observation equation (data characteristic measurement relationship), which can transform the implicit information of the system to visible output. The state estimation of the current time is obtained by the state transfer process and the prediction results of the previous time. Using the error between the actual measurement (noise interference) and the estimated observation information at the current moment, a weighted correction term is generated to realize the updating process. This allows the state estimation and prediction system model to be

obtained with high reliability, and the process of battery state tracking and RUL predictive evaluation can be realized. A flowchart is shown in Figure 5.

Using the state tracking process to identify the model parameters and battery RUL prediction, the SIR, APF, and CSPF algorithms were compared (since the application of the algorithm is limited by the dimension of the system variables, the GPF is not suitable for this example). State tracking parameters were identified before the charge-discharge cycle S ; i.e., we estimated the optimal model parameters that can minimize the error between the predicted value of the algorithm and the actual value of the experiment. After the charge-discharge cycle S , the battery capacity was predicted to determine whether the failure threshold was exceeded. The original data of the capacity degradation of the lithium-ion battery were from the open-source experimental data of the Center for Advanced Life Cycle Engineering (CALCE) of the University of Maryland. An Arbin BT2000 battery system was the experimental platform, and the test data were stored in Excel format. A data sample with a

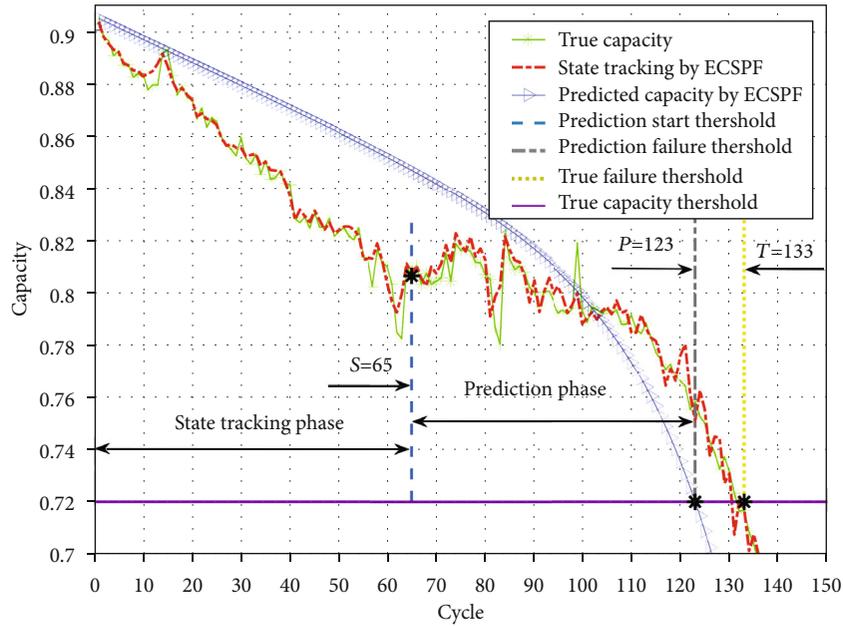


(a) SIR

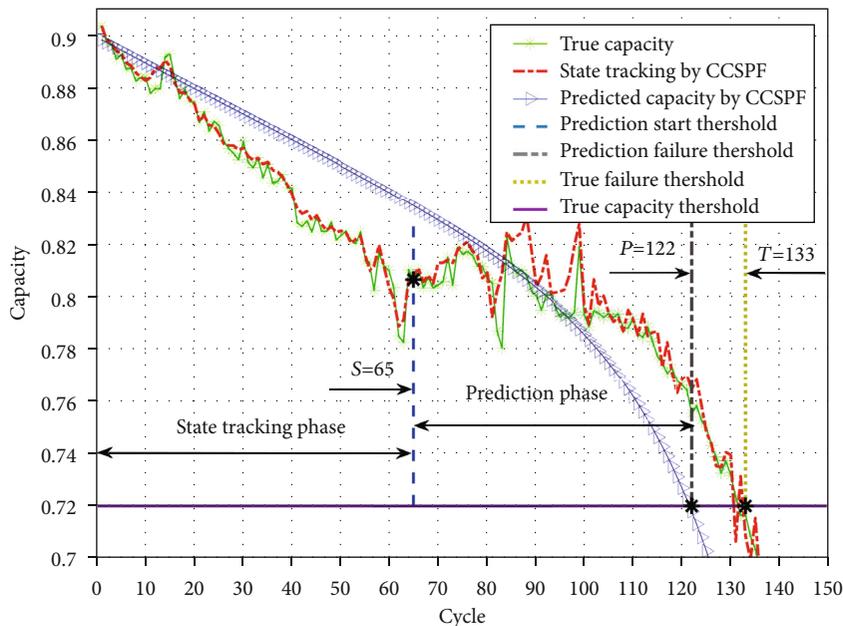


(b) APF

FIGURE 6: Continued.



(c) ECSPF



(d) CCSPF

FIGURE 6: State tracking and RUL prediction results for the battery based on SIR, APF, ECSPF, and CCSPF. Note: SIR = sequential importance resampling; APF = auxiliary particle filter; ECSPF = Euclidean distance clustering similarity particle filter; CCSPF = Chebyshev distance clustering similarity particle filter.

normalized capacity of 0.90 Ah, M8, was selected as the test set for comparison of the state assessment methods. The battery was charged at the standard constant current of 0.5 C until it reached 4.2 V and was switched to constant voltage charging until the current decayed to 0.05 A, and the discharge was considered complete when the voltage dropped to 2.7 V during the charge-discharge cycle [43], as shown in Table 4. While collecting experimental data, due to the error from the accuracy of measuring equipment and human misoperation, a small amount of abnormal data was extracted

from the battery dataset, as it would have affected the quality of the history dataset. The size of the dataset would also affect the operational cost, and simplifying the data (e.g., taking one point from each n point) would reduce this cost. Therefore, the original data were preprocessed, filtered, and simplified.

Generally, 80% of the rated capacity was taken as the end-of-life threshold T , and the actual failure threshold was 133 cycles according to open-source data, where the state tracking set was $S = 65$ and the number of particles was $N = 100$. The simulation results are shown in Figure 6 and Tables 5

TABLE 5: Comparison of the average state tracking effect for four nonlinear algorithms running 1000 times.

Filter	SSE	MSE	RMSE	R^2
SIR	1.3526×10^{-5}	2.0810×10^{-7}	1.4328×10^{-5}	7.8318×10^{-1}
APF	1.8923×10^{-5}	2.9112×10^{-7}	1.6408×10^{-5}	6.4347×10^{-1}
ECSPF	8.7368×10^{-7}	1.3441×10^{-8}	3.6547×10^{-6}	9.8419×10^{-1}
CCSPF	1.0425×10^{-6}	1.6038×10^{-8}	3.9975×10^{-6}	9.8281×10^{-1}

Note: SIR = sequential importance resampling; APF = auxiliary particle filter; ECSPF = Euclidean distance clustering similarity particle filter; CCSPF = Chebyshev distance clustering similarity particle filter; SSE = sum of squared error; MSE = mean squared error; RMSE = root mean squared error; R^2 = coefficient of determination.

TABLE 6: Comparison of RUL prediction results based on four nonlinear algorithms running 100 times.

Comparison items	SIR	APF	CSPF	
			ECSPF	CCSPF
Mean	113	115	123	122
Variance	156.3	476.6	5.9	6.2
Error	15.0%	13.5%	7.5%	8.3%
RMSE	23.5%	26.8%	10.6%	11.1%

Note: SIR = sequential importance resampling; APF = auxiliary particle filter; ECSPF = Euclidean distance clustering similarity particle filter; CCSPF = Chebyshev distance clustering similarity particle filter; RMSE = root mean squared error.

and 6. Figure 6 can be used to compare the effect of state tracking and the accuracy of capacity prediction of different algorithms. Tables 5 and 6 compare the state tracking and prediction results obtained by multiple simulation calculations. The evaluation indicators of the state tracking effect are the sum of squared error (SSE), MSE, RMSE, and coefficient of determination (R^2). The closer SSE, MSE, and RMSE were to 0, the closer R^2 was to 1, and the better was the effect of state tracking. The evaluation indicators of the prediction accuracy were the mean, variance, relative error, and RMSE of the prediction failure threshold.

Figure 6 and Table 5 show that compared with the SIR and APF algorithms, the state tracking indicators and effect of the proposed CSPF algorithm were superior, and the Euclidean distance measure yielded the best results. The excellent state tracking effect of the CSPF method also guarantees that the attenuation mechanism of state tracking in the early period is the same as in later-stage predictions over the whole life cycle of the battery. As shown in Figure 6 and Table 6, after 100 simulations using SIR, APF, ECSPF, and CCSPF, the average failure thresholds of RUL prediction were 113, 115, 123, and 122 cycles, respectively, and the relative error was within 15.0%. Taking ECSPF as an example, the variance, relative error, and RMSE of the RUL prediction failure threshold were significantly lower than those of SIR and APF, and the prediction trend was relatively closer to the actual capacity degradation curve. The relative error of ECSPF was better than those of SIR and APF by about 50% and 45%, respectively, and the threshold variances were lower by factors of about 26 and 80, respectively. RMSE was about 55% and 60% greater, respectively. The results show that the prediction accuracy of CSPF was relatively

higher than those of the other algorithms, the discrete degree of the state particle set of prediction results was the minimum, the degree of uncertainty expression was the lowest, and the algorithm was more robust.

6. Conclusions

Facing the difficulties of the standard PF algorithm in nonlinear system state estimation, such as low precision, instability, and low computational efficiency, we proposed an improved PF algorithm (CSPF) based on the consistency principle of the spatial state trajectory. Relying on the model construction of spatial trajectories between sampled particles and actual states, current and future multistage measurement information was predicted by SIS and GPF to form trajectories combining the original and modified trajectories. The similarity of the combined trajectories was measured by clustering analysis to guide the generation of new distributions and update the particle importance weights to mitigate the particle degradation phenomenon. Because resampling is not adopted in the improved algorithm, the problem of particle depletion is fundamentally eliminated. The convergence theorem of the improved algorithm of the CSPF and MSE of the results was proved. The effectiveness of CSPF was verified by comparison with current methods in cases of socioeconomic prediction and battery health assessment. The application experiments showed that, compared to SIR, APF, and GPF, CSPF had higher accuracy and better robustness for the state estimation of nonlinear systems under the influence of Gaussian or non-Gaussian noise. Primary conclusions are summarized as follows:

- (1) In Gaussian and non-Gaussian noise environments, the prediction accuracy of CSPF was improved significantly, more than 50% in typical cases
- (2) The latest measurement information was used to update the new proposal distribution, and the computational cost of CSPF was reduced by adjusting the number of sampling particles and the order of the state trajectory. In a typical case, the RMSE of CSPF was reduced by a factor of more than 2 in the same computing time
- (3) The first-order Markov model was modified by the clustering similarity of state trajectories so that the indicators of state tracking effects were relatively

higher, which provided an accuracy guarantee for the prediction model. In the battery application, the prediction accuracy of CSPF was the best, and the relative error of the RUL failure threshold was maintained within 8%

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors sincerely thank the Guangxi Engineering Technology Research Center of Ship Digital Design and Advanced Manufacturing and the Intelligentized Robotic Welding Technology Laboratory of Shanghai Jiao Tong University for their intelligence support and valuable comments in performing this research. We appreciate the generous financial support of the National Natural Science Foundation of China (Nos. 51969001 and 61763006), the Guangxi Natural Science Foundation of China (Nos. 2018GXNSFAA138080 and 2021GXNSFBA075023), the Guangxi Science and Technology Plan Project of China (No. GuikeAD18281007), and the Innovation Project of Guangxi Graduate Education (YCBZ2019050).

References

- [1] X. Ding, Z. Wang, L. Zhang, and C. Wang, "Longitudinal vehicle speed estimation for four-wheel-independently-actuated electric vehicles based on multi-sensor fusion," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 12797–12806, 2020.
- [2] Z. Zhou, Y. Tan, and R. Dong, "Fault detection of piezoceramic actuator using non-smooth observer," *International Journal of Applied Electromagnetics and Mechanics*, vol. 47, no. 4, pp. 975–991, 2015.
- [3] Q. Wang, Z. Wang, L. Zhang, P. Liu, and Z. Zhang, "A novel consistency evaluation method for series-connected battery systems based on real-world operation data," *IEEE Transactions on Transportation Electrification*, vol. 7, no. 2, pp. 437–451, 2021.
- [4] C. Wang, Z. Wang, L. Zhang, D. Cao, and D. G. Dorrell, "A vehicle rollover evaluation system based on enabling state and parameter estimation," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4003–4013, 2021.
- [5] Z. Zhou, Y. Tan, and X. Liu, "State estimation of dynamic systems with sandwich structure and hysteresis," *Mechanical Systems and Signal Processing*, vol. 126, pp. 82–97, 2019.
- [6] D. Luenberger, "An introduction to observers," *IEEE Transactions on Automatic Control*, vol. 16, no. 6, pp. 596–602, 1971.
- [7] Z. Zhou, Y. Tan, R. Dong, and L. Zhang, "Fault detection for sandwich systems with hysteresis based on robust observer," *International Journal of Applied Electromagnetics and Mechanics*, vol. 49, no. 4, pp. 577–595, 2015.
- [8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [9] Z. Zhou and X. Liu, "State and fault estimation of sandwich systems with hysteresis," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 13, pp. 3974–3986, 2018.
- [10] Y. Guan and M. Saif, "A novel approach to the design of unknown input observers," *IEEE Transactions on Automatic Control*, vol. 36, no. 5, pp. 632–635, 1991.
- [11] H. K. Khalil and L. Praly, "High-gain observers in nonlinear feedback control," *International Journal of Robust and Nonlinear Control*, vol. 24, no. 6, pp. 991–992, 2014.
- [12] Z. Zhou, Y. Tan, Y. Xie, and R. Dong, "State estimation of a compound non-smooth sandwich system with backlash and dead zone," *Mechanical Systems and Signal Processing*, vol. 83, pp. 439–449, 2017.
- [13] K. S. Phogat and D. E. Chang, "Discrete-time invariant extended Kalman filter on matrix Lie groups," *International Journal of Robust and Nonlinear Control*, vol. 30, no. 12, pp. 4449–4462, 2020.
- [14] Q. Meng, Y. Sun, and Z. Cao, "Adaptive extended Kalman filter (AEKF)-based mobile robot localization using sonar," *Robotica*, vol. 18, no. 5, pp. 459–473, 2000.
- [15] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEEE Proceedings F: Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.
- [16] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [17] M. K. Pitt and N. Shephard, "Filtering via simulation: auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, 1999.
- [18] J. H. Kotecha and P. M. Djuric, "Gaussian particle filtering," *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2592–2601, 2003.
- [19] J. H. Kotecha and P. M. Djuric, "Gaussian sum particle filtering," *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2602–2612, 2003.
- [20] P. Guarniero, A. M. Johansen, and A. Lee, "The iterated auxiliary particle filter," *Journal of the American Statistical Association*, vol. 112, no. 520, pp. 1636–1647, 2017.
- [21] Y. Sun, X. Ran, Y. Li, G. Zhang, and Y. Zhang, "Thruster fault diagnosis method based on Gaussian particle filter for autonomous underwater vehicles," *International Journal of Naval Architecture and Ocean Engineering*, vol. 8, no. 3, pp. 243–251, 2016.
- [22] M. L. Psiaki, J. R. Schoenberg, and I. T. Miller, "Gaussian sum reapproximation for use in a nonlinear filter," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 2, pp. 292–303, 2015.
- [23] X. Tang, K. Liu, X. Wang, B. Liu, F. Gao, and W. D. Widanage, "Real-time aging trajectory prediction using a base model-oriented gradient-correction particle filter for lithium-ion batteries," *Journal of Power Sources*, vol. 440, pp. 227118.1–227118.11, 2019.
- [24] X. Fu and Y. Jia, "An improvement on resampling algorithm of particle filters," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5414–5420, 2010.
- [25] A. T. Cemgil, "A tutorial introduction to Monte Carlo methods, Markov Chain Monte Carlo and particle filtering,"

- Academic Press Library in Signal Processing*, vol. 1, pp. 1065–1114, 2014.
- [26] M. Ahwiadi and W. Wang, “An adaptive particle filter technique for system state estimation and prognosis,” *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 9, pp. 6756–6765, 2020.
- [27] R. Havangi, “Robust evolutionary particle filter,” *ISA Transactions*, vol. 57, pp. 179–188, 2015.
- [28] A. Salarpour and H. Khotanlou, “Direction-based similarity measure to trajectory clustering,” *IET Signal Processing*, vol. 13, no. 1, pp. 70–76, 2019.
- [29] H. He and Y. Tan, “Unsupervised classification of multivariate time series using VPCA and fuzzy clustering with spatial weighted matrix distance,” *IEEE Transactions on Cybernetics*, vol. 50, no. 3, pp. 1096–1105, 2020.
- [30] J. Candy, “Bootstrap particle filtering,” *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 73–85, 2007.
- [31] L. Zhang, Y. Zhu, P. Shi, and Y. Zhao, “Resilient asynchronous Hoo filtering for Markov jump neural networks with unideal measurements and multiplicative noises,” *IEEE Transactions on Cybernetics*, vol. 45, no. 12, pp. 2840–2852, 2015.
- [32] Z. Zhou, Y. Tan, and P. Shi, “Fault detection of a sandwich system with dead-zone based on robust observer,” *Systems & Control Letters*, vol. 96, pp. 132–140, 2016.
- [33] H. He, Y. Tan, and J. Xing, “Unsupervised classification of 12-lead ECG signals using wavelet tensor decomposition and two-dimensional Gaussian spectral clustering,” *Knowledge-Based Systems*, vol. 163, pp. 392–403, 2019.
- [34] Z. Zhou, Y. Tan, Y. Xie, and R. Dong, “Soft measurement of states of sandwich system with dead zone and its application,” *Measurement*, vol. 87, no. 1, pp. 219–234, 2016.
- [35] H. Zhou, Z. Deng, Y. Xia, and M. Fu, “A new sampling method in particle filter based on Pearson correlation coefficient,” *Neurocomputing*, vol. 216, pp. 208–215, 2016.
- [36] M. Rhif, A. B. Abbes, I. Farah, B. Martínez, and Y. Sang, “Wavelet transform application for/in non-stationary time-series analysis: a review,” *Applied Sciences*, vol. 9, no. 7, 2019.
- [37] K. Liu, Y. Shang, Q. Ouyang, and W. D. Widanage, “A data-driven approach with uncertainty quantification for predicting future capacities and remaining useful life of lithium-ion battery,” *IEEE Transactions on Industrial Electronics*, vol. 68, no. 4, pp. 3170–3180, 2021.
- [38] A. Wen, J. Meng, J. Peng, L. Cai, and Q. Xiao, “Online parameter identification of the lithium-ion battery with refined instrumental variable estimation,” *Complexity*, vol. 2020, Article ID 8854618, 12 pages, 2020.
- [39] B. Saha, K. Goebel, and J. Christophersen, “Comparison of prognostic algorithms for estimating remaining useful life of batteries,” *Transactions of the Institute of Measurement and Control*, vol. 31, no. 3-4, pp. 293–308, 2009.
- [40] R. B. Wright, C. G. Motloch, J. R. Belt et al., “Calendar- and cycle-life studies of advanced technology development program generation 1 lithium-ion batteries,” *Journal of Power Sources*, vol. 110, no. 2, pp. 445–470, 2002.
- [41] W. He, N. Williard, M. Osterman, and M. Pecht, “Prognostics of lithium-ion batteries based on Dempster-Shafer theory and the Bayesian Monte Carlo method,” *Journal of Power Sources*, vol. 196, no. 23, pp. 10314–10321, 2011.
- [42] Z. Wei, J. Zhao, D. Ji, and K. J. Tseng, “A multi-timescale estimator for battery state of charge and capacity dual estimation based on an online identified model,” *Applied Energy*, vol. 204, pp. 1264–1274, 2017.
- [43] Z. Jiao, X. Fan, X. Zhang, Y. Luo, and Y. Liu, “State tracking and remaining useful life predictive method of Li-ion battery based on improved particle filter algorithm,” *Transaction of China Electrotechnical Society*, vol. 35, no. 18, pp. 3979–3993, 2020.