Received: 8 October 2020 | Revised: 21 December 2020 | Accepted: 9 January 2021

IET Smart Grid

DOI: 10.1049/stg2.12013

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH PAPER

Smooth particle filter-based likelihood approximations for remaining useful life prediction of Lithium-ion batteries

Mo'ath El-Dalahmeh | **Maher Al-Greer** | **Ma'd El-Dalahmeh** | **Michael Short**

School of Computing Engineering and Digital Technologies, Teesside University, Middlesbrough, UK

Correspondence

Maher Al-Greer, School of Computing Engineering and Digital Technologies, Teesside University, Middlesbrough TS1 3BX, UK.
Email: M.Al-Greer@tees.ac.uk

Abstract

Accurate prediction of the remaining useful life (RUL) in Lithium-ion batteries (LiBs) is a key aspect of managing its health, in order to promote reliable and secure systems, and to reduce the need for unscheduled maintenance and costs. Recent work on RUL prediction has largely focused on refining the accuracy and reliability of the RUL prediction. The author introduces a new online RUL prediction for LiB using smooth particle filter (SPF)- based likelihood approximation method. The proposed algorithm can accurately estimate the unknown degradation model parameters and predict the degradation state by solving the optimisation problem at each iteration, rather than only taking a gradient step, that tends to lead to rapid convergence, avoids instability issues and improves predictive accuracy. From the experimental datasets published by Prognostics Centre of Excellence (PCoE) of NASA, a second order degradation model was created to explore the degradation of LiB, utilising non-linear characteristics and non-Gaussian capacity degradation. RUL prediction was tested with various predicted starting points to assess whether the amount of data and parameters' uncertainty influenced the accuracy of the prediction. Results show that the proposed prediction approach gives improved prediction accuracy and improves the convergence rate in comparison with the particle filter (PF) and other methods such as unscented particle filter (UPF). Since the maximum error of the SPF predicting approach is relatively small, RUL prediction in the best case at the prediction starting point consisting of 80 cycles is 127 cycles. The prediction relative error was approximately 0.024, and the absolute error of the proposed algorithm is around 2 cycles, which is lower than the PF (around 16 cycles). RUL prediction is close to 108 cycles and relative error is around 0.136, while the absolute error prediction is approximately 16.

1 | INTRODUCTION

Generally, electric storage devices, for example in electric vehicles (EV) and grid balancing applications, are now heavily reliant on Lithium-ion Batteries (LiBs). The widespread use of LiBs results from their being light, able to store large amounts of energy in a compact space, and long lasting capacity [1]. However, their capacity decreases over time and with use, due to degradation in material components; in turn, this leads to decreased storage capacity and energy [2]. Therefore, monitoring battery degradation, forecasting battery status, and improving maintenance have become important focal points in LiB engineering research, to improve battery performance and reliability. To do so, an accurate estimate of a battery's remaining useful life (RUL) and state of health (SOH) is required [3,4]. RUL is the term used for the quantity of cycle left between the present cycle and the cycle at which battery's End of Service (EoS) occurs, and can vary from 70% to 80% of nominal capacity [5].

In general, predictive methods can be split into machine learning (ML) and model-based methods [6]. ML techniques have recently been implemented to model battery deterioration and forecast the RUL of LiBs. For instance, the auto-regressive (AR) model was proposed in [7] to predict the diminishing capacity in LiB. The other example is the improved relevance vector machine (RVM), which is a technique for improving

certainty in RUL prediction, stability and accuracy [8]. In addition, a feedforward neural network (FNN) was applied in [9] to enhance RUL prediction. The authors in [10] proposed a framework aimed at estimating battery capacity based on multi-channel ML methods using an FNN, convolutional neural network (CNN) and long short-term memory (LSTM), to improve prediction accuracy based on the diversity of possible data from current, voltage and temperature. In all these techniques, however, the training needs to be extensive, inclusive, unbiased, and good quality. Additionally, most ML algorithms need off-line training to accurately model the battery, and they also have a high computational load for online RUL prediction [11].

With model-based methods, mathematical ageing models that capture long-term battery degradation dependencies are required. Because of the mathematical simplicity, wide validity and high flexibility, RUL can be predicted [12]. Most studies in the literature have utilised a model that is generally linear, exponential and polynomial [13–15]. Model-based approaches are also associated with advanced Bayesian, Kalman and Particle filters (KF and PF, respectively) [16]. These can update the parameters of the model as part of the diagnostic process, to ensure accurate RUL prediction. The best candidate for solving linear system problems with Gaussian noise is the KF [6]. A linear model of capacity degradation linked with two filters has also been proposed to estimate the remaining battery life [17]; however, the process of battery degradation is often non-linear, and this is where different KF algorithms, such as an unscented KF or extended KF [18], may address the above-mentioned issue. According to [19], most errors in the process of predicting RUL derive from several sources when obtaining data, and, thus, total noise often does not show Gaussian behaviour. In this context, therefore, the application of a KF algorithm leads to divergence. However, the method of health diagnosis includes solutions for non-Gaussian non-linear system-based problems. For this reason, studies have tended to consider PF algorithms, which give solutions for both non-linear and non-Gaussian issues [20]. Several papers have employed PF to determine the failure points of LiBs. For example, in [21,22], a method was proposed to predict failure using the exponential model and classical PF. In addition, although a second-order polynomial was presented in [14], which contains fewer parameters than the exponential model, this model is less accurate than the exponential model.

In general, PFs suffer from two main problems: (1) particle degeneracy and (2) particle impoverishments. The latter is due to the fact that a PF has a resampling phase that can reduce particle degeneracy, and this may also result in a loss of sample particles [23]. Accordingly, other types of PFs used to solve these problems have been considered. For example, an unscented PF (UPF) was presented in [24] to improve the sampling and reshaping of PF. The authors in [25] presented a scheme for battery capacity estimation based on the estimated capacity using a Gauss-Hermite PF algorithm to predict the failure limit for the uncertainty in the RUL prediction. Markov Chain Monte Carlo (MCMC) method has been applied in [26] to solve sample problem impoverishment in a UPF algorithm. Regularised particle filters have also been used in the resampling phase to enhance PF accuracy, as presented in [27]. A Rao-Blackwellized PF (RBPF) was suggested in [28] to limit the distribution of likelihood into a subspace of the state distribution of likelihood in the state space sample. The authors in [29] have been integrated the neural networks radial basis with a PF to the end-of-discharge prediction for a LiB. Similarly in [30], NN model and new PF algorithm know as Bat-PF has been integrated to improve the accuracy of the residual life prediction and to reduce PF impoverishment and degeneracy. Also, second-order central difference (SCD-PF) algorithm was introduced in [31] to improve the performance of the PF for RUL prediction of Limbs. However, most previous improved algorithms have extensively reduced the problems faced by the PF algorithm in terms of particle decomposition and sample diversity deficiency and obtained a strong result for LiB RUL prediction. Nevertheless, issues with particle degradation and particle diversity deficiency remain difficult for RUL prediction.

The key contribution of this work is to improve the RUL prediction of LiB battery by smoothing the PF using likelihood approximations scheme [32], combined with a second order degradation model. The proposed SPF algorithm improves the accuracy of RUL prediction by choosing the proposal distribution and the resampling weights, depending on certain current parameter estimates, thus overcoming the problem of particle impoverishment and uncertainty in the degradation model parameters. This article is structured as follows. Theoretical background and the procedures of implementation for the PF and the proposed SPF algorithms are set out in Section 2. Capacity modelling for LiB based on the experimental data collected by PCoE of NASA is demonstrated in Section 3. Obtained results are presented and discussed in Section 4. Finally, conclusions and discussion are summarised in Section 5.

2 | THEORETICAL BACKGROUND-METHODOLOGY

2.1 | PF algorithm

The PF algorithm combines two techniques: recursive Bayesian and sequential importance sampling (SIS) [30]. It also contains two important elements, which are the initialization of parameters, and the state model equation. The state-spatial dynamic model can be represented by the state transformation model and the measuring model, using (1) and (2) [33].

$$x_k = f_k(x_{k-1}, \omega_{k-1}) \tag{1}$$

$$z_k = h(x_k, v_k) \tag{2}$$

where, x_k represents the hidden state variables at k^{th} time, z_k represent the measurement system at k^{th} time, ω_k is the noise process and v_k is the noise measurement. The PF algorithm is comprised of the prediction and update stage. In the prediction stage, the previous probability distribution of state x_k is

calculated in (3), without making use of information from the measurement model z_k [21].

$$p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})dx_{k-1} \quad (3)$$

In the update stage, the previous probability distribution is altered by the measurement model z_k to achieve the end distribution of x_k at time k.

$$p(x_k|z_{1:k}) = \frac{p(z_k|x_k)p(x_k|z_{1:k-1})}{\int p(z_k|x_k)p(x_k|z_{1:k-1})dx_k} \quad (4)$$

The main stage of the PF algorithm is to estimate the end probability distribution function (PDF) with particles $\{x_k^{(i)}\}_{i=1}^N$, weighted with the associated $-\{w_k^{(i)}\}_{i=1}^N$. Thus, the PDF end in (4) may be rewritten as follows [21]:

$$p(x_{0:k}|z_{1:k}) \approx \sum_{i=1}^{N} w_k^i \delta(x_{0:k} - x_{0:k}^i) \quad (5)$$

where N is the number of particles and $\delta(\cdot)$ represents the Dirac delta function. The particles generated by the distribution $p(x_k|z_{1:k})$ represent the sample perfectly. However, from the accurate PDF posterior density, it is still difficult to take a precise sample, and so an alternative way of sampling needs to be found to sample proposal distribution $q(x_k|z_{1:k})$. The weighting of the filter may be improved using the SIS and taking another sample in the stages of the SMC algorithm. The associate weight of a random particle drawn from $q(x_k|z_{1:k})$ is represented as [33]:

$$w_k^i \propto \frac{p(x_k^i|z_{1:k})}{q(x_k^i|z_{1:k})} \quad (6)$$

Suppose the proposal distribution $q(x_k|z_{1:k})$ can be factorised into:

$$q(x_{0:k}|z_{1:k}) = q(x_{0:k-1}|z_{1:k-1})q(x_k|x_{0:k-1}, z_{1:k}) \quad (7)$$

Then the recursive form of the posterior probability density function can be expressed as:

$$\begin{aligned} p(x_{0:k}|z_{1:k}) &= \frac{p(z_{1:k}|x_{0:k}, z_{1:k-1})p(x_{0:k}|z_{1,k-1})}{p(z_{1:k}|z_{1:k-1})} \\ &= \frac{p(z_{1:k}|x_{0:k}, z_{1:k-1})p(x_k|x_{0:k-1}, z_{1:k-1})p(x_{0:k-1}|z_{1:k-1})}{p(z_{1:k}|z_{1:k-1})} \\ &= \frac{p(z_{1:k}|x_k)p(x_k|x_{k-1})p(x_{0:k-1}|z_{1:k-1})}{p(z_{1:k}|z_{1:k-1})} \\ &\propto p(z_{1:k}|x_k)p(x_k|x_{k-1})p(x_{0:k-1}|z_{1:k-1}) \end{aligned} \quad (8)$$

By substituting (7) and (8) into (6), the particle weight update equation can be expressed as:

$$\begin{aligned} w_k^i &\propto \frac{p(z_k|x_k^i)p(x_k^i|x_{k-1}^i)p(x_{0:k}^i|z_{1:k-1})}{q(x_k^i|x_{0:k-1}^i, z_{1:k})q(x_{0:k-1}^i|z_{1:k-1})} \\ &= w_{k-1}^i \frac{p(z_k|x_k^i)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{0:k-1}^i, z_{1:k})} \end{aligned} \quad (9)$$

As long as the constraint $q(x_k^i|x_{0:k-1}^i, z_{1:k}) = q(x_k^i|x_{k-1}^i, z_k)$ is satisfied, the modified weight calculation (9) can be transformed into:

$$w_k^i \propto w_{k-1}^i \frac{p(z_{1:k}|x_k^i)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{k-1}^i, z_{:k})} \quad (10)$$

The proposal distribution $q(x_k^i|x_{k-1}^i, z_{:k})$ was chosen as the prior distribution $p(x_k^i|x_{k-1}^i)$ in the PF algorithm, to simplify implementation. The summary of the PF procedure is presented in Table 1 [21]. Step 1: at $k = 0$, the prior probability distribution function $p(x_0)$ is used to produce an array of initial particles $\{x_0^i\}_{i=1}^N$ and corresponding particle weight is assigned as $w_0^{1:N} = 1/N$. Steps 2 and 3: particles are updated using Equations (1) and (2); proposal distribution function $q(x_k^i|x_{k-1}^i, z_k)$ is selected as $p(x_k^i|x_{k-1}^i)$, then the weight of the particles can be calculated at time k (see Table 1, step 3). Step 4: by copying large weights, a new sample of particles $\{x_0^i\}_{i=1}^N$ can be drawn, and the corresponding weights are reassigned to $1/N$. Step 5: the new state can be updated based on the newly obtained particles $\{x_0^i\}_{i=1}^N$ and weights $\tilde{w}_k^i$.

2.2 | The proposed SPF algorithm

When using the PF algorithm to estimate maximum probability (likelihood) parameters in the non-linear state-space model, the PF removes the light weights and copies the heavy weights in a resample phase, which results in a loss of diversity in the particle distribution [26].

As presented in [32], the main challenge is that likelihood distribution estimation and its derivatives are fundamentally noisy; the main idea of the SPF method is to choose the proposal distribution $q(x_k^i|x_{k-1}^i, z_k)$and the resampling weights w_k^n, such that it is entirely independent of parameters θ (in this application, the parameters of the degradation model). Based on this choice, it is noted that all the randomly extracted elements, such as particles $x_{0:T}^n$ and ancestor indices $a_{1:T}^n$ (furthermore are the $a_{1:T}^n$ drawn with respect to the w_k^n) in the PF algorithm, became independent of θ; this is critical in the analysis and estimation of battery degradation as the true values of the degradation model parameters are unknown, and highly influenced by uncertainty [24]. Therefore, this article combines the second-order empirical degradation model with the SPF algorithm [32] to improve the RUL prediction of LiB

TABLE 1 Procedures of the PF

Step 1	Initialisation: draw particles $x_0^i \sim p(x_0),\ i = 1, 2, 3, \ldots, N$.
Step 2	Time update: for k = 1:N generate new importance sample for $x_k^i \sim q(x_k^i \mid x_{0:k-1}^i, z_{0:k})$.
Step 3	Normalise importance weights: Assign the weight of the particle according to: $w_k^i = w_{k-1}^i \frac{p(z_k \mid x_k^i) p(x_k^i \mid x_{k-1}^i)}{q(x_k^i \mid x_{k-1}^i, z_k)}\#$ $w_k^i = \left(\frac{1}{\sqrt{2\pi}}\right) \cdot e^{-(z_k - \hat{z}_k^i)^2}/2$ Normalise the weight: $w_k^i = w_k^i / \sum_{i=1}^N w_k^i \#$
Step 4	Particles re-sampling: the effective sample size N_{eff} is calculated as $N_{eff} \approx 1/\sum_{i=1}^N (w_k^i)^2$. If $N_{eff} < \frac{2}{3}N$, the posterior samples can be generated by resampling from the current particle set, the corresponding weights are $1/N$.
Step 5	Output: State estimation: $\hat{x}_k = \sum_{i-1}^N \tilde{w}_k^i \tilde{x}_k^i \#$

battery by smoothing the PF using likelihood approximations scheme.

Now, if a certain condition is applied to the realisation of $\{x_{0:T}^n, a_{1:T}^n\}$, the $\hat{z}\theta$ estimation will convert into a deterministic function within θ, and any standard optimisation routine can then be implemented to solve (11) and find the maximum likelihood estimate of θ [32].

$$p_\theta(z_{1:T}) = \int p(x_0) \prod_{t=1}^{T} f_\theta(x_t|x_{t-1}) h_\theta(z_t|x_t) dx_{0:T} \tag{11}$$

$$\hat{\theta} = \arg max_\theta p_\theta(z_{1:T}) \tag{12}$$

where $p_\theta(z_{1:T})$ refers to the likelihood function when considered a function of θ. This follows from (1) and the initial state density $p(x_0)$. However, the strength of the PF method is the ability to construct samples sequentially over high space dimensions X^{T+1}, where the resampling phase provides valuable feedback information to discover which parts of the state space should be explored further. Based on the arbitrary decision, the weights of θ-independent resampling w_k^n will be lost, and thus missing this feature may lead to a discrepancy in the estimate obtained. The deterministic function can be ascertained in θ-independent re-sampling by allowing the algorithm to let the resampling weights w_k^n and $q(x_t|x_{k-1}, z_k)$ rely on certain current parameter predictions, θ_{k-1}, as [32],

$$q(x_t|x_{t-1}, z_t) = f_{\theta_{k-1}}(x_t|z_t) \tag{13}$$

$$w_k^n = h_{\theta_{k-1}}(z_t|x_t^n) \tag{14}$$

The SPF choice was θ_{k-1} instead of θ. If the θ value is somewhat close to the value of θ_{k-1}, the variance of the estimate of the maximum likelihood state of the particle distribution, referred to as $\hat{z}_{\theta_{k-1}}(\theta)$, may not be prohibitively large. On the other hand, if the current value of θ_{k-1} is far from the current value of $\widehat{\theta}$, then the estimate $\hat{z}_{\theta_{k-1}}(\theta)$ will not be particularly good at the $\widehat{\theta}$. For this reason, we must repeat the parameter values over k until we roughly arrive at values close to $\widehat{\theta}$. By inserting (13) and (14) into the particle filter algorithm and combining with an external optimisation loop, an innovative method is proposed and, presented in Table 2 [32]. The steps of the method are: Step 1: the initial parameters are extracted and initialised; Step 2: Run the Particle Filter method to draw particles, calculated using $\{x_0^i\}_{i=1}^N$ from the initial distribution function in line one (as we assume it is independent of θ), and the importance weight is set as $w_k^i = 1$. Moreover, in step 2, line 4, $a_{1:T}^n$ is drawn concerning the reconfiguration weights w_{t-1}^j. For particle propagation, in step 2, line 5, the new particles are generated from the proposal distribution $q(x_k^i|x_{k-1}^i, z_k)$; Step 3: a new sample of $\{x_0^i\}_{i=1}^N$ particles can be obtained by copying large weights and assigning corresponding weights $\frac{1}{N}\sum_{n=1}^N log\, w_t^n$; and Step 4: The new state is thus reached based on new particles $\{x_0^i\}_{i=1}^N$ and weights $\tilde{w}_k^i$.

It can be observed in Table 2 (Step 1, line 4) that the optimization step related to solving arg $max_\theta \hat{z}_{\theta_{k-1}}(\theta)$ has been established. Importantly, this issue is now deterministic, and any usual numerical optimisation tool can be implemented, and the experiments will show this using the general-purpose optimisation tool fminunc in Matlab. The structure of $\hat{z}_{\theta_{k-1}}(\theta)$, which is implicitly defined in the function likelihood of the proposed algorithm, might still be utilised by a more suitable optimisation scheme. Its structure can be shown as [32]:

$$\hat{z}_{\theta_{k-1}}(\theta) = \frac{1}{N} \prod_{k=1}^{T} \sum_{n=1}^{N} c_k^n w_k^n(\theta) f_\theta\left(x_k \middle| x_{k-1}^{a_k^n}\right) h_\theta(z_k|x_k^n) \tag{15}$$

where c_k^n is a constant that is independent of θ, $w_k^n(\theta)$ depends on θ but always fulfils $\sum_{n=1}^N w_k^n(\theta) = 1$, and f_θ and h_θ rely on the degradation model.

TABLE 2 Procedures of the proposed method

Step 1	1. Set $\boldsymbol{\theta}_0$ (Initial parameter of degradation model)
	2. **for k** = 1, ... **do**
	3. Call $\{\boldsymbol{x}_{0:T}^n, \boldsymbol{a}_{1:T}^n\}$ from particle filter $(\boldsymbol{\theta}_{k-1})$
	4. Solve $\boldsymbol{\theta}_k \leftarrow \mathbf{argmax}_{\boldsymbol{\theta}}\ \mathbf{log\_likelihood}(\boldsymbol{\theta}, \boldsymbol{\theta}_{k-1}, \{\boldsymbol{x}_{0:T}^n, \boldsymbol{a}_{1:T}^n\}_{n=1}^N)$
Step 2	Function particle filter $(\boldsymbol{\theta}_{k-1})$
	1. $\boldsymbol{x}_0^n$ particles are first taken from the initial distribution $\boldsymbol{p}(\boldsymbol{x}_0)$
	2. Set the importance weight $(\boldsymbol{w}_k^i = 1)$
	3. **for t** = 1, **to** $\boldsymbol{T}$ **do**
	4. Generate $\boldsymbol{a}_t^n$ from the $\boldsymbol{C}(\{\boldsymbol{w}_{t-1}^j\}_{j=1}^N)$
	5. Propagate $\boldsymbol{x}_k^i \sim \boldsymbol{f}_{\boldsymbol{\theta}_{k-1}}(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1}^{\boldsymbol{a}_k^n}, \boldsymbol{z}_k)$
	6. Set $\boldsymbol{w}_k^n \leftarrow \boldsymbol{h}_{\boldsymbol{\theta}_{k-1}}(\boldsymbol{z}_k \mid \boldsymbol{x}_k^n)$
	Return $\{\boldsymbol{x}_{0:T}^n, \boldsymbol{a}_{1:T}^n\}_{n=1}^N$
Step 3	Function $\mathbf{log\_likelihood}(\boldsymbol{\theta}, \boldsymbol{\theta}_{k-1}, \{\boldsymbol{x}_{0:T}^n, \boldsymbol{a}_{1:T}^n\}_{n=1}^N)$
	1. **for t** = 1, **to** $\boldsymbol{T}$ **do**
	2. Set $\boldsymbol{w}_t^i \leftarrow \frac{\boldsymbol{w}_{t-1}^{a_t^n} / \sum_j \boldsymbol{w}_{t-1}^{a_t^n\, j}}{\boldsymbol{h}_{\boldsymbol{\theta}_{k-1}}(\boldsymbol{x}_{t-1}^{a_t^n} \mid \boldsymbol{y}_{t-1}) / \sum_j \boldsymbol{h}_{\boldsymbol{\theta}_{k-1}}(\boldsymbol{x}_{t-1}^j \mid \boldsymbol{y}_{t-1})} \frac{\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}^{a_t^n})}{\boldsymbol{f}_{\boldsymbol{\theta}_{k-1}}(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}^{a_t^n})} \boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{y}_t \mid \boldsymbol{x}_t^n)$
	3. Set $\boldsymbol{z}_t \leftarrow \frac{1}{N} \sum_{n=1}^N \boldsymbol{log}\, \boldsymbol{w}_t^n$
	Return $\mathbf{log}\ \hat{\mathbf{z}}_{\boldsymbol{\theta}_{k-1}}(\boldsymbol{\theta}) \leftarrow \sum_{t=1}^{\mathbf{T}} \mathbf{log}\ \mathbf{z}_t$
Step 4	State prediction: $\hat{x}_k = \sum_{i-1}^{N} \tilde{w}_k^i \tilde{x}_k^i \#$

3 | CAPACITY DEGRADATION MODELLING

Subsequently, three cells were selected (B05, B06 and B18) and their experimental data were published by the Prognostics Centre of Excellence (PCoE) at NASA Ames Research Centre, to investigate the performance and accuracy of the proposed algorithm [34]. As shown in Figure 1, the dataset consist of four cells using commercial lithium cobalt oxide batteries. The stated capacity of the cells is 2 Ah, and their nominal voltage is 3.3 V. The cells are iterated through the cycle until they fail, at a room temperature of 24℃. The following steps provide a description of the test procedure and data collection conditions [34]:

1. In the charge step, Constant-Current, Current-Voltage protocol (CC-CV) is used. A CC of 1.5 Ah was applied until the cell voltage reached the maximum limit (4.2 V) and then the CV continued until the current dropped to 20 mAh.
2. In the discharge step, a CC 2 A level applied until the voltage of battery dropped to 2.7 V, 2.5 V and 2.5 V for batteries for batteries B05, B06 and B18, respectively.
3. Repeat the previous steps until the batteries reach the point of failure, here it is 30% of the nominal capacity, and so the battery's EoS is placed at a capacity threshold of $U = 1.4$ Ah.

During the degradation process, a LiB continues to decrease in capacity. As shown in Figure 1, battery capacity decreases dramatically with an increase in the number of cycles

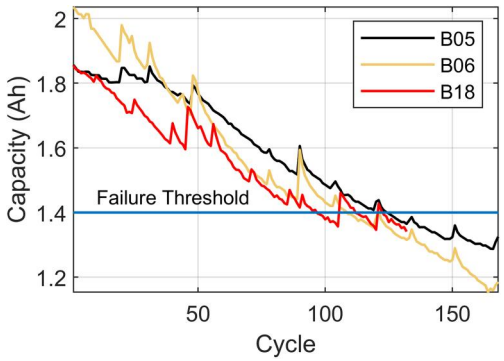

FIGURE 1 The capacity degradation curve

used. Thus, the degradation curves are well fitted using an exponential growth model as presents in (16).

$$Q = a \cdot \exp(b \cdot k) + c \cdot \exp(d \cdot k) \tag{16}$$

Here Q represents the capacity of the battery, a, b, c and d are the model parameters, and k is the number of cycles. Note that a curve fitting toolbox was used to obtain an accurate exponential degradation model. Figure 2 show the fitting results. The findings show that the degradation model is effective and can be used to predict the RUL battery. The modelled parameters of the four batteries in the study were produced at the fitting stage, to attain the starting parameters for the training data. These were used in the prediction step, as shown in Table 2.

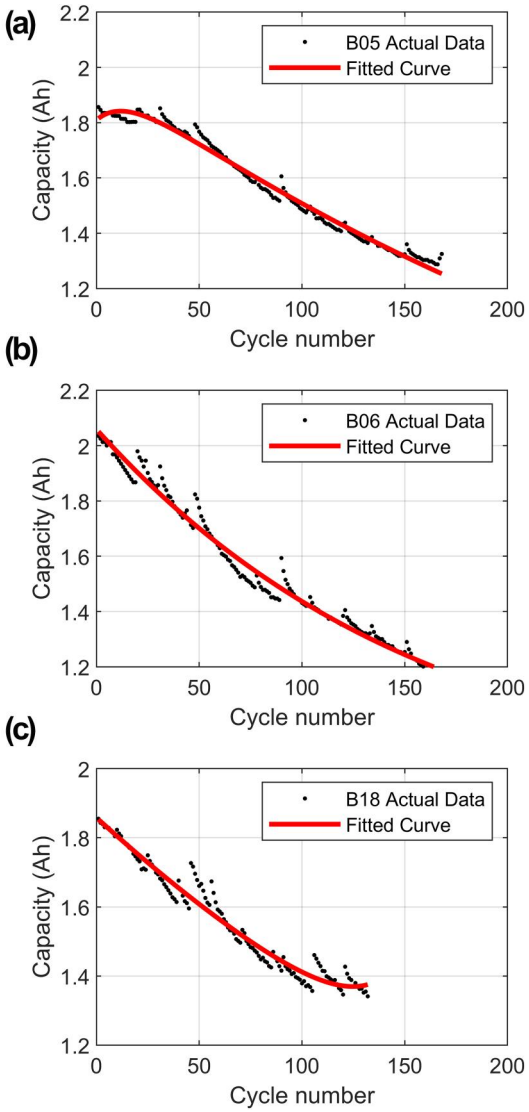

FIGURE 2 Degradation data and fitted curve

4 | EXPERIMENTAL VALIDATION AND DISCUSSION

Now, the SPF and PF algorithms in Tables 1 and 2 are implemented to demonstration the effectiveness of the proposed solution. Furthermore, in order to test the accuracy of the prediction of the PF and proposed SPF algorithm, different cycle 'starting points' were applied, such as 20, 50 and 80 cycles. Again, the two algorithms were used here for online estimation of the parameters in (16). A second order degradation model has been developed based on the fitting results explained earlier in Section 3. The performance of the prediction has been evaluated using absolute error (*AE*) of the RUL, the relative error (*RE*) of the RUL and the root-bean square error (*RMSE*) of the RUL. As given in (17), AE is defined as the difference between the number of remaining true (RUL_T) cycles and the number of predicted (RUL_P) cycles. While the *RE* is defined as presented in (18) and RMSE is defined as presented in (19).

$$AE = |RUL_T - RUL_P| \tag{17}$$

$$RE = AE/RUL_T \tag{18}$$

$$RMSE = \sqrt{\frac{1}{k_p}\sum_{k=strt}^{C}(Q_k - Q_{\hat{k}})} \tag{19}$$

where k_p is the number of cycles that must be predict and $Q_{\hat{k}}$ represent the prediction result at cycle k.

4.1 | RUL prediction

According to the capacity degradation model, the state transition of the battery system can be defined as follows:

$$x_k = [a_k; b_k; c_k; d_k] \tag{20}$$

where

$$\begin{cases} a_k = a_{k-1} + \omega_a & \omega_a \sim N(0, \sigma_a) \\ b_k = b_{k-1} + \omega_b & \omega_c \sim N(0, \sigma_b) \\ c_k = c_{k-1} + \omega_c & \omega_c \sim N(0, \sigma_c) \\ d_k = d_{k-1} + \omega_d & \omega_d \sim N(0, \sigma_d) \end{cases} \tag{21}$$

Now Equation (16) can be written as

$$Q_k = a_k \exp(b_k k) + c_k \exp(d_k k) + v_k \; v_k \sim N(0, \sigma_V) \tag{22}$$

Here, Q_k is the measurement of capacity cell at cycle k, and $N(0, \sigma)$ is the Gaussian noise with zero mean and σ is the standard deviation. Then, the measurement of capacity cell can be estimated by:

$$Q_k = \sum_{i=1}^{N} Q_k^i = \sum_{i=1}^{N}\left[a_k^i . \exp\left(b_k^i \cdot k\right) + c_k^i . \exp\left(d_k^i \cdot k\right)\right] \tag{23}$$

At cycle k, the prediction step (p-th) can be calculated by:

$$Q_{k+p} = \sum_{i=1}^{N}\left[a_k^i . \exp\left(b_k^i \cdot (k+p)\right) + c_k^i . \exp\left(d_k^i \cdot (k+p)\right)\right] \tag{24}$$

The posterior PDF can be estimated with weights on each trajectory:

$$P\left(Q_{k+p}\middle|Q_{O:k}\right) \approx \sum_{i=1}^{N} w_k^i \delta\left(Q_{k+p} - Q_{k+p}^i\right) \tag{25}$$

In this analysis, the value of the failure threshold is 70% of the nominal capacity value. Then, at cycle k the RUL distribution can be predicted by [35]:

$$0.7Q_{nominal} = a_k^i.\exp\left(b_k^i \cdot L_k^i\right) + c_k^i.\exp\left(d_k^i \cdot L_k^i\right) \quad (26)$$

$$P(L_k|Q_{0:k}) \approx \sum_{i=1}^{N} w_k^i\left(L_k - L_k^i\right) \quad (27)$$

Here, $L_k{}^i$ is the RUL at cycle k.

4.2 | RUL prediction using B05 cell

Note that the authors use the $B05$ battery cell for RUL prediction. The initial values of the PF and proposed SPF parameters were selected as: number of particles $N = 200$, and battery failure threshold 1.4 Ah. The initial parameters for the degradation model for all battery cells are shown in Table 3.

Figure 3 shows the prediction result with PF, and the proposed SPF algorithm for battery cell $B05$. It is important to mention that the first 80 cycles from the data are used as training data to update the prediction process. There are two curves of prediction, and the respective PDFs of the RUL were obtained to compare between the PF and the proposed SPF. Here, PDF indicates the possibility of the lithium-ion battery's end life in each interval, as shown in Figure 3a,b, respectively. As seen in Figure 3a, at $T_s = 80$ cycles, the final life cycle was 126 cycles, while the average predicted life cycle using the PF was 108 cycles. Thus, the AE for the PF algorithm is around16, the maximum RE was about 0.136 and error was approximately 0.1128. While for the proposed algorithm (see Figure 3b), the average number of life cycles predicted was around 127 cycles, and the prediction AE was approximately 1 cycles, the prediction RE was around 0.014 and the RMSE error was roughly 0.0198. From the prediction results, it is observed that the prediction curve obtained from the proposed SPF method is closer to the actual capacity degradation curve than the PF prediction curve, and its PDF of RUL, based on the proposed SPF algorithm, is more concentrated. Figure 3c shows the two prediction curves at the same time to check the PF algorithm and the proposed SPF algorithm performance results during the entire lifecycle, which clearly reveal the effectiveness of the proposed algorithm. Figure 4 likewise shows the process of predicting the RUL by relying on the above mentioned two methods. Here, the first 50 cycles were used to update the prediction process. As shown in Figure 4, an identical conclusion similar to the above can be deduced from Figure 3, which shows the robustness and strong accuracy of the proposed SPF algorithm.

Figure 5a,b show the prediction RE of the RUL using the PF method and the proposed SPF method; in the prediction update phase, the first 50 and 80 cycles are used, respectively. The RE curve extracted from the PF method starts with quite a small error rate but then diverges from the actual capacity values, leading to an increase in the error rate. This corresponds to how the previously mentioned particle filter suffers from particle degeneration and impoverishment. In contrast, the curve extracted from the proposed SPF method with the cycle passage converges more than the actual capacity values, indicating that the proposed method is more reliable.

Table 4 shows the error rates for the prediction RUL of the battery B05 obtained by the PF and proposed SPF methods with different starting points. The AE, RE and RMSE of the proposed SPF algorithm are significantly

TABLE 3 Initial model parameters

Cell ID	a	b	c	d
B05	1.974	−0.00027	−0.158	−0.06942
B06	1.562	−0.00557	0.4895	0.0009
B18	1.853	−0.00291	0.0002	0.0428

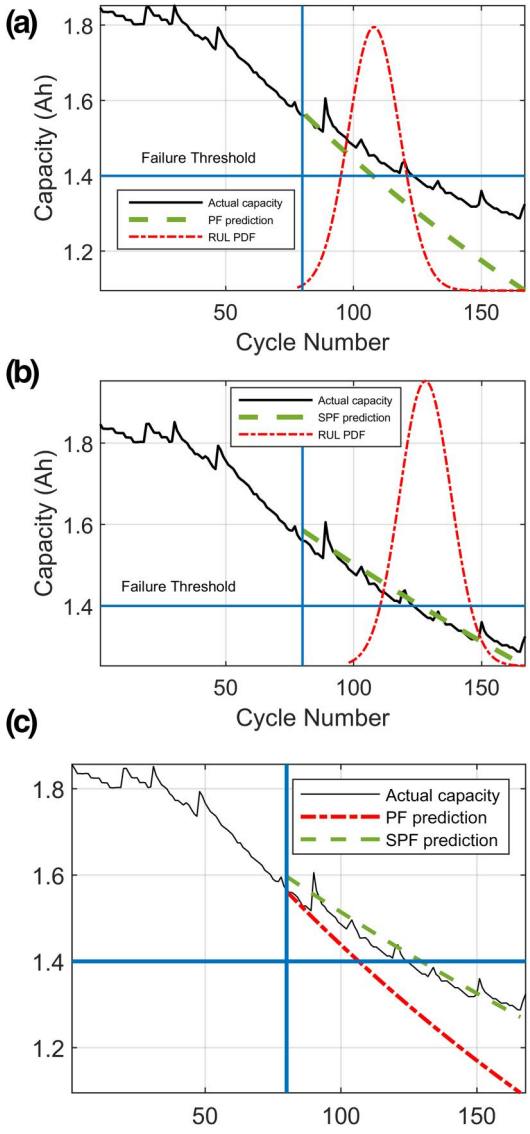

FIGURE 3 Prediction RUL results at 80 cycles for B05. (a) PF, (b) SPF and (c) comparison results

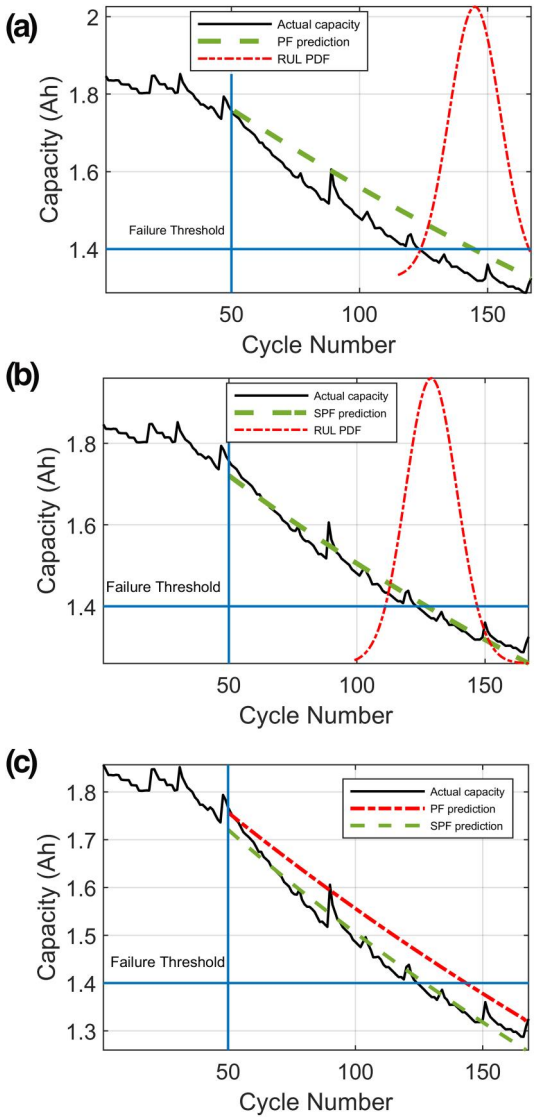

FIGURE 4 Prediction RUL results at 50 cycles for B05. (a) PF, (b) SPF and (c) comparison results

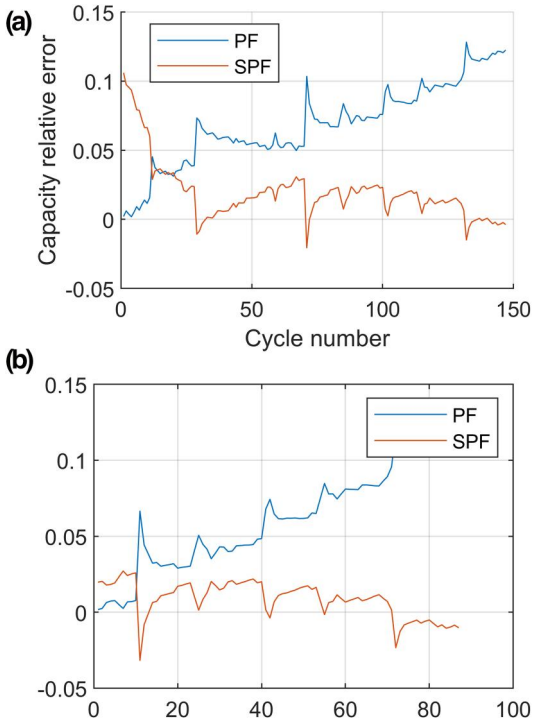

FIGURE 5 The prediction relative error B05: (a) the RE at 50 cycles and (b) the RE at 80 cycles

TABLE 4 RUL prediction results of B05

Method	T_S	RUL_{True}	RUL_{Pred}	AE	RA	RMSE
PF	20	125	94	31	0.2480	0.1385
	50	125	144	19	0.16	0.0570
	80	125	108	16	0.1360	0.1128
SPF	20	125	134	9	0.072	0.0532
	50	125	129	4	0.032	0.0209
	80	125	126	1	0.014	0.0198

TABLE 5 Comparative results

Method	Cell ID	AE	RE	RMSE
U-LOCR-PF [26]	B05	1	0.01	0.0198
SCD-PF [31]	B05	8	-	-
Proposed SPF	B05	1	0.014	0.0198
PF	B05	16	0.136	0.1128

smaller than of the PF algorithm. In addition, the findings clearly show (Figures 3 and 4) that the start point is continuously regressed, and so the prediction error becomes nearer to zero; this is in line with real-time application. The findings also show that the algorithm converges with the predicted start point, and thus much more training data can be employed for learning, and better degradation knowledge and characteristics can be revealed. It was also seen that the prediction error is lower in the reverse direction, indicating that the particles are more compacted as more capacity data are accessible. To evaluate the prediction accuracy of the proposed approach, it is necessary to compare the obtained result with the latest state-of-the-art approaches [25,30] conducted for the same case study. Table 5 presents the results of the proposed solutions. In addition, the performance of each approach was further evaluated using AE, RE and RMSE of the RUL. As shown in Table 5, the proposed SPF approach and U-LOCR-PF enhanced the prediction accuracy of the PF more than the proposed SCD-PF approach, which demonstrates the robustness of the proposed approach.

In terms of applying the RUL prediction method [26], this approach is complicated when compared to our proposed method. This is because it adopts a unscented kalman filter algorithm to obtain the proposal distribution as an

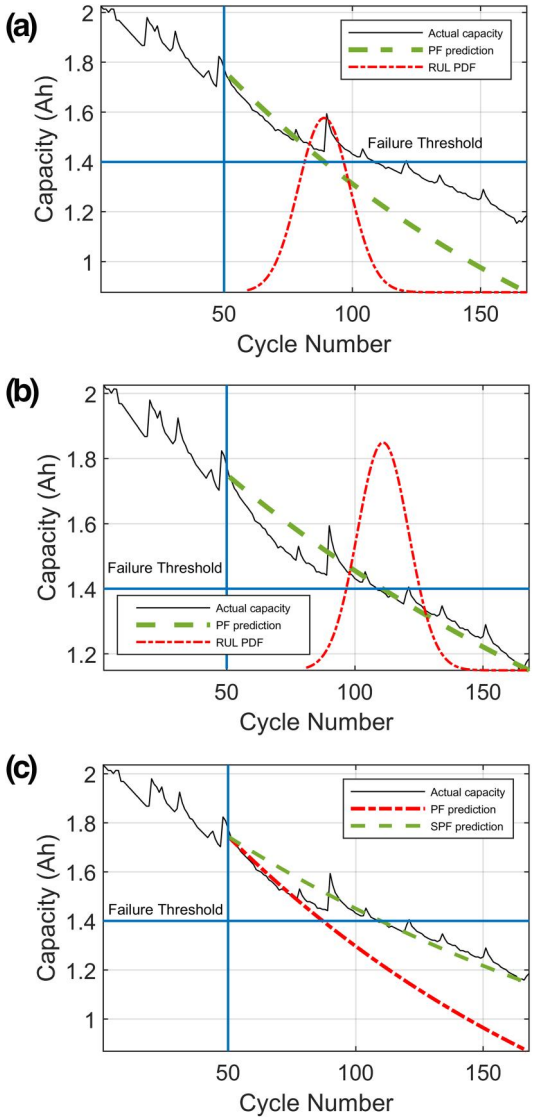

FIGURE 6 Prediction RUL results at 50 cycles for B06. (a) PF, (b) SPF and (c) comparison results

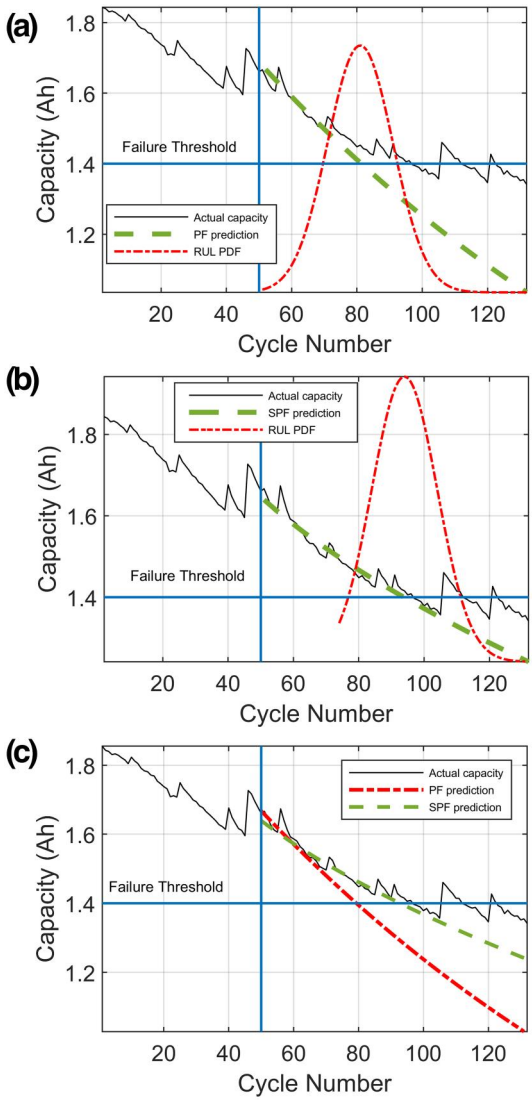

FIGURE 7 RUL prediction results at 50 cycles for B18. (a) PF, (b) SPF and (c) comparison results

essential function of the PF algorithm, and is then integrated with the linear optimisation algorithm to overcome the particle deficiency problem. The linear optimisation algorithm can be influenced by the step parameter K, and the fuzzy inference method is used to evaluate the value of the K step coefficient. As for the method proposed in reference [31], this is also complicated in an implementation; its methodology relies on presenting the second-order central difference Kalman filter (SCDKF) method to choose the proposal distribution function. After this, SCDKF is integrated with the PF technique to resolve particle impoverishment.

However, the proposed method aims to operate the SPF based on the current parameters, and then uses the output from this SPF to re-evaluate the approximation of the likelihood function. The calculation time of the proposed method is higher than the conventional PF method due to the fact that finding the maximum likelihood estimates of unknown parameters in nonlinear state-space models is generally time consuming. In addition, the degradation capacity model (16) is monotonous, although part of a lithium-ion battery's degradation pattern is typically non-monotonic and exhibits strong fluctuation. Consequently, the degradation model (16) may sometimes be insufficient to track the degradation trend.

4.3 | RUL prediction using (*B*06 and *B*18) batteries

Further, to verify the result obtained previously, capacity degradation data was used for *B*06 and *B*18 batteries to check the accuracy of the proposed method for RUL prediction. Figures 6 and 7 show the RUL result for the battery *B*06 and *B*18 data using the PF and proposed SPF algorithms at starting point 50 cycle. Similar to *B*05 dataset, the accuracy of the estimation findings and the RUL PDF gained by the suggested SPF approach is just greater than that provided by

TABLE 6 RUL prediction results of B06 and B18

Cell ID	Method	T_S	RUL_T	RUL_p	AE	RA	RMSE
B06	PF	20	109	133	24	0.2202	0.0958
		50	109	89	20	0.1835	0.1905
		80	109	98	11	0.1009	0.0708
	SPF	20	109	113	4	0.0367	0.0454
		50	109	111	2	0.0183	0.0446
		80	109	108	1	0.014	0.0414
B18	PF	20	98	74	24	0.2474	0.1455
		50	98	81	17	0.1694	0.1727
		80	98	89	8	0.0825	0.1502
	SPF	20	98	106	9	0.0928	0.0414
		50	98	93	5	0.0309	0.0610
		80	98	100	2	0.0206	0.0594

the PF. That is, the prediction error of the suggested approach is four cycles lower than the PF using 50 cycles, and the value of the *RE* and *AE* are also reduced. As shown in Table 6 both the prediction relative error and absolute error under various prediction starting point (T_s) of the proposed SPF algorithm for $B06$ and $B18$ are smaller than that of the PF algorithm, which indicates that the stability of the proposed SPF algorithm is higher than PF algorithm. For example, at $T_s = 50$ cycles, the B06 battery's *AE* prediction of SPF was approximately 2 cycles, the maximum *RE* was about 0.0183 and that *AE* of PF algorithm was around 20 cycles, and the *RE* was approximately 0.1835, which can lead to the same conclusions as discussed for B05.

5 | CONCLUSION

Here, the authors have presented an innovative online RUL prediction of LiBs known as SPF algorithm. Experimental datasets published by PCoE of NASA, were used and a second-order exponential degradation model to validate the effectiveness and stability of the proposed method was developed. The results obtained clearly indicated that the proposed SPF algorithm can improve the prediction accuracy compared with the classical PF algorithm. The average RUL errors and PDF width of the SPF approach are less than in PF methods, demonstrating that the suggested method is more accurate and steadier. In addition, RUL prediction was tested with various predicted starting points to assess whether the amount of data influenced the accuracy of the prediction. The findings clearly demonstrated that the amount of data affects the accuracy of the prediction. It has also been shown that the earlier the starting point of the prediction, the higher the prediction error rate relative to the higher starting point. In fact, the predicted curve further diverges from the actual degradation curve. Future research is planned to focus on designing robust degradation models such as Multiphysics model with an emphasis on accurate and reliable RUL prediction at a rapid convergence rate.

DATA AVAILABILITY STATEMENT

Data may available on request from the authors.

ORCID

Mo'ath El-Dalahmeh https://orcid.org/0000-0003-4696-2056

REFERENCES

1. Hu, X., et al.: Technological developments in batteries: a survey of principal roles, types, and management needs. IEEE Power Energy Mag. 15(5), 20–31 (2017)
2. Vetter, J., et al.: Ageing mechanisms in lithium-ion batteries. J. Power Source. 147(1-2), 269–281 (2005)
3. Dong, G., et al.: Battery health prognosis using Brownian motion modeling and particle filtering. IEEE Trans. Indus. Electron. 65(11), 8646–8655 (2018)
4. Sayfutdinov, T., Ali, M., Khamisov, O.: Alternating direction method of multipliers for the optimal siting, sizing, and technology selection of Li-ion battery storage. Electr. Power Syst. Res. 185, 106388 (2020)
5. El Mejdoubi, A., et al.: State-of-charge and state-of-health lithium-ion batteries' diagnosis according to surface temperature variation. IEEE Trans. Indus. Electron. 63(4), 2391–2402 (2016)
6. Wu, L., Fu, X., Guan, Y.: Review of the remaining useful life prognostics of vehicle lithium-ion batteries using data-driven methodologies. Appl. Sci. 6(6) (2016)
7. Long, B., et al.: An improved autoregressive model by particle swarm optimization for prognostics of lithium-ion batteries. Microelectron. Reliabil. 53(6), 821–831 (2013)
8. Datong, L., et al.: A health indicator extraction and optimization framework for lithium-ion battery degradation modeling and prognostics. IEEE Trans. Syst. Man Cyber. Syst. 45(6), 915–928 (2015)
9. Wu, J., Zhang, C., Chen, Z.: An online method for lithium-ion battery remaining useful life estimation using importance sampling and neural networks. Appl. Energy. 173, 134–140 (2016)
10. Choi, Y., et al.: Machine learning-based lithium-ion battery capacity estimation exploiting multi-channel charging profiles. IEEE Access. 7, 75143–75152 (2019)
11. Li, Y., et al.: Data-driven health estimation and lifetime prediction of lithium-ion batteries: A review. Renew. Sust. Energy Rev. 113 (2019)
12. Chang, Y., Fang, H.: A hybrid prognostic method for system degradation based on particle filter and relevance vector machine. Reliabil. Eng. Syst. Safety. 186, 51–63 (2019)
13. He, W., et al.: Prognostics of lithium-ion batteries based on Dempster–Shafer theory and the Bayesian Monte Carlo method. J. Power Source. 196(23), 10314–10321 (2011)
14. Micea, M.V., et al.: Online state-of-health assessment for battery management systems. IEEE Trans. Instrument. Measure. 60(6), 1997–2006 (2011)
15. Qin, Q., et al.: Adaptive and robust prediction for the remaining useful life of electrolytic capacitors. Microelectron. Reliabil. 87, 64–74 (2018)
16. Arachchige, B., Perinpanayagam, S., Jaras, R.: Enhanced prognostic model for lithium ion batteries based on particle filter state transition model modification. Appl. Sci. 7(11) (2017)
17. Burgess, W. L.: Valve regulated lead acid battery float service life estimation using a Kalman filter. J. Power Source. 191(1), 16–21 (2009)
18. Zheng, X., Fang, H.: An integrated unscented kalman filter and relevance vector regression approach for lithium-ion battery remaining useful life and short-term capacity prediction. Reliabil. Eng. Syst. Safety. 144, 74–82 (2015)

19. Saha, B., et al.: Prognostics methods for battery health monitoring using a Bayesian framework. IEEE Trans. Instrument. Measure. 58(2), 291–296 (2009)
20. An, D., Choi, J.-H., Kim, N. H.: Prognostics 101: A tutorial for particle filter-based prognostics algorithm using Matlab,. Reliabil. Eng. Syst. Safety. 115, 161–169 (2013). https://doi.org/10.1016/j.ress.2013.02.019
21. Su, X., et al.: Prognostics of lithium-ion batteries based on different dimensional state equations in the particle filtering method. Trans. Instit. Measure. Control. 39(10), 1537–1546 (2016)
22. Zhang, L., Mu, Z., Sun, C.: Remaining useful life prediction for lithium-ion batteries based on exponential model and particle filter. IEEE Access. 6, 17729–17740 (2018)
23. Omariba, Z. B., Zhang, L., Sun, D.: Remaining useful life prediction of electric vehicle lithium-ion battery based on particle filter method. In: 2018 IEEE 3rd international conference on big data analysis (ICBDA), 9–12 March, pp. 412–416 (2018)
24. Hu, Y., et al.: A particle filtering and kernel smoothing-based approach for new design component prognostics. Reliabil. Eng. Syst. Safety. 134, 19–31 (2015)
25. Hu, C., et al.: Method for estimating capacity and predicting remaining useful life of lithium-ion battery. Appl. Energy. 126, 182–189 (2014)
26. Zhang, H., et al.: An improved unscented particle filter approach for lithium-ion battery remaining useful life prediction. Microelectron. Reliabil. 81, 288–298 (2018)
27. Zhang, X., Miao, Q., Liu, Z.: Remaining useful life prediction of lithium-ion battery using an improved UPF method based on MCMC. Microelectron. Reliabil. 75, 288–295 (2017)
28. Musso, C., Oudjane, N., Le Gland, F.: Improving regularised particle filters. In: Doucet, A., de Freitas, N., Gordon, N. (eds.) Sequential Monte Carlo Methods in Practice. Springer New York, New York, NY, pp. 247–271. (2001)
29. Sbarufatti, C., et al.: Adaptive prognosis of lithium-ion batteries based on the combination of particle filters and radial basis function neural networks. J. Power Source. 344, 128–140 (2017)
30. Wu, Y., et al.: Remaining useful life prediction of lithium-ion batteries using neural network and bat-based particle filter. IEEE Access. 7, 54843–54854 (2019)
31. Chen, Y., et al.: Remaining useful life prediction and state of health diagnosis of lithium-ion battery based on second-order central difference particle filter. IEEE Access. 8, 37305–37313 (2020)
32. Svensson, A., Lindsten, F., Schön, T. B.: Learning nonlinear state-space models using smooth particle-filter-based likelihood approximations. IFAC-PapersOnLine. 51(15), 652–657, 01 January (2018)
33. Arulampalam, M.S., et al.: A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Transactions on Signal Processing. 50(2), 174–188 (2002)
34. Saha, B., Goebel, K.: Battery Data Set. NASA Ames Prognostics Data Repository. NASA Ames Research Center, Moffett Field, CA (2007)
35. Miao, Q., et al.: Remaining useful life prediction of lithium-ion battery with unscented particle filter technique. Microelectronics Reliability. 53, 805–810 (2013)

How to cite this article: El-Dalahmeh M, Al-Greer M, El-Dalahmeh M, Short M. Smooth particle filter-based likelihood approximations for remaining useful life prediction of Lithium-ion batteries. *IET Smart Grid*. 2021;1–11. https://doi.org/10.1049/stg2.12013