The development of machine learning-based remaining useful life prediction for lithium-ion batteries

Li, Xingjun; Yu, Dan; Vilsen, Søren Byg; Stroe, Daniel-Ioan

Published in:
Journal of Energy Chemistry

DOI (link to publication from Publisher):
10.1016/j.jechem.2023.03.026

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

Journal of Energy Chemistry 82 (2023) 103–121

Contents lists available at ScienceDirect

Journal of Energy Chemistry

journal homepage: www.elsevier.com/locate/jechem

http://www.journals.elsevier.com/journal-of-energy-chemistry/

Review

The development of machine learning-based remaining useful life prediction for lithium-ion batteries

Xingjun Li [a], Dan Yu [a,*], Vilsen Søren Byg [a,b], Store Daniel Ioan [a,*]

[a] Department of Energy, Aalborg University, Aalborg 9220, Denmark
[b] Department of Mathematical Sciences, Aalborg University, Aalborg 9220, Denmark

ARTICLE INFO

Article history:
Received 31 January 2023
Revised 16 March 2023
Accepted 18 March 2023
Available online 31 March 2023

Keywords:
Lithium-ion batteries
Remaining useful lifetime prediction
Machine learning
Lifetime extension

ABSTRACT

Lithium-ion batteries are the most widely used energy storage devices, for which the accurate prediction of the remaining useful life (RUL) is crucial to their reliable operation and accident prevention. This work thoroughly investigates the developmental trend of RUL prediction with machine learning (ML) algorithms based on the objective screening and statistics of related papers over the past decade to analyze the research core and find future improvement directions. The possibility of extending lithium-ion battery lifetime using RUL prediction results is also explored in this paper. The ten most used ML algorithms for RUL prediction are first identified in 380 relevant papers. Then the general flow of RUL prediction and an in-depth introduction to the four most used signal pre-processing techniques in RUL prediction are presented. The research core of common ML algorithms is given first time in a uniform format in chronological order. The algorithms are also compared from aspects of accuracy and characteristics comprehensively, and the novel and general improvement directions or opportunities including improvement in early prediction, local regeneration modeling, physical information fusion, generalized transfer learning, and hardware implementation are further outlooked. Finally, the methods of battery lifetime extension are summarized, and the feasibility of using RUL as an indicator for extending battery lifetime is outlooked. Battery lifetime can be extended by optimizing the charging profile serval times according to the accurate RUL prediction results online in the future. This paper aims to give inspiration to the future improvement of ML algorithms in battery RUL prediction and lifetime extension strategy.

Contents

* Corresponding authors.
E-mail addresses: dayu@energy.aau.dk (D. Yu), dis@energy.aau.dk (S. Daniel Ioan).

https://doi.org/10.1016/j.jechem.2023.03.026

1. Introduction

Thanks to their advantages of high energy density, long lifetime and low self-discharging, lithium-ion batteries have been used in many fields like electric vehicles (EV) and portable electronic devices [1]. However, complex operation conditions including temperature variation and different charging-discharging cycles can accelerate negative changes in internal electrochemical constituents of lithium-ion batteries like the loss of lithium inventory (LLI) and active material (LAM) and the increase of internal resistance. Serious negative changes in the battery are caused by complex and coupled side reactions affected by internal mechanical stresses and external factors like operation temperature, depth of discharge (DOD), charge and discharge current magnitude, and ambient humidity, which will lead to capacity degradation and performance deterioration. Specifically, LLI is caused by the side reactions consuming Li-ions like solid electrolyte interface (SEI) formation, lithium plating, and electrolyte decomposition. LAM is related to the active particle loss caused by the chemical decomposition and dissolution reaction of transition metals. The increase of internal resistance is caused by the SEI film formation and cracks in particles [2]. Usually, when the capacity of the battery reaches 80% of its initial value, it is regarded as being at its end of life (EOL) [3]. If the batteries are used beyond the EOL criterion, it might lead to poor system performance and sometimes catastrophic events. Therefore, to ensure reliable operation and battery safety, remaining useful life (RUL) prediction is necessary. A battery management system (BMS) can reference the RUL prediction result to control the operation of the batteries. Users can get timely maintenance or replacement of the battery [4]. Furthermore, the RUL prediction based on data from early cycles can reduce the cost and time of aging tests, which is beneficial for battery design, production, and optimization. Besides, RUL prediction is important for the evaluation of batteries retired from EVs due to the deteriorated capacity and complex internal characteristics [5]. The RUL prediction result describes when the battery will fail (i.e., it will no longer meet the requirement of the application).

The battery RUL can be expressed as:

$$\mathrm{RUL} = T_{\mathrm{EOL}} - T_{\mathrm{C}} \tag{1}$$

where T_{EOL} represents the battery life obtained from the battery life experiment. T_{C} is the current usage time of the battery. Equation (1) considers calendar aging and cycle aging at the same time. Most research usually defines the RUL based on cycle aging only. Another definition that can reflect RUL is expressed as:

$$\mathrm{RUL_i} = \frac{C_{\mathrm{i}} - C_{\mathrm{EOL}}}{C_{\mathrm{nominal}} - C_{\mathrm{EOL}}} * 100\% \tag{2}$$

where C_{i}, C_{nominal} and C_{EOL} represent the present capacity, nominal capacity, and end-of-life capacity respectively.

Generally, the RUL prediction is carried out by either physics-based or data-driven methods. Physics-based methods mainly include different types of electrochemical models and equivalent circuit models, while data-driven methods use less physical knowledge to reflect degradation information replacing it with historical and operational data. The key to RUL prediction for lithium-ion batteries is figuring out how to accurately learn the long-term degradation characteristics over hundreds of cycles and thousands of hours of operation based on limited historical data. Machine learning (ML) methods are a class of data-driven methods that perform well in modeling non-linear systems. For example, Guo et al. used different ML methods to predict electrochemical impedance spectroscopy (EIS) based on the battery charging voltage curve with less predicted error. In the context of RUL prediction, ML methods have been researched extensively. Luo et al. demonstrated that ML methods can be used to construct fast and accurate data-driven models for the prediction of battery performance [6]. However, many review articles about the general ML-based RUL prediction have been published without paying attention to the evolution of the research core of common ML algorithms [3,7-9]. Therefore, it is not easy for readers to find potential research directions. For this reason, this work tries to track the research core of the most used ML algorithms for lithium-ion battery RUL prediction based on the related literature form the past decade. The literature filter process and review goal are shown in Fig. 1.

Specifically, the keywords 'remaining useful life' & 'lithium-ion batteries' were input in the Web of Science and the publication time was set from 2013 to 2022 with review articles excluded, obtaining 580 relevant publications. This review work is mainly focused on the specific application of common ML algorithms in RUL prediction of battery, so the literature review articles were downloaded and analyzed individually. It is found that most of the articles involve state of health (SOH) estimation and RUL prediction at the same time. Many ML algorithms can be used to estimate SOH and predict RUL with different input variables. Through reading and filtering the title and abstract of 580 papers according to the keywords, 380 relevant journal and conference papers were identified to be concentrated on RUL prediction using ML methods. The other database was also used to filter the missed and important research articles. Based on the selected literature, the RUL prediction flow of commonly used ML algorithms and their research cores are analyzed. Fig. 2 shows the number of papers utilizing the top ten most used ML algorithms for RUL prediction from 2013 to 2022. The recurrent neural network (RNN) is the most frequently used algorithm appearing in 61 papers. Here RNNs mainly include the basic RNN method, echo state network (ESN), long short-term memory (LSTM), and gated recurrent unit (GRU). The quantity of articles on the ANN algorithm follows, mainly referring to the classical feed-forward neural network (FFNN) except RNN and convolutional neural network (CNN) of all NN algorithms. Support vector regression (SVR), relevance vector machine (RVM), auto-regression (AR), and gaussian process regression (GPR) were all used in more than 25 papers. CNN and extreme learning machine (ELM) were found to be researched 15 times, while random forest regression (RFR) and auto-encoder (AE) were both used in less than 10 papers.

In addition to reflecting the battery life, RUL prediction can also be used for battery lifetime extension. Several methods to extend the battery lifetime have been investigated, such as charging profile optimization [10–12], discharging profile optimization [13], thermal management optimization [14,15], the introduction of additional energy storage devices like supercapacitors [16], driving behavior optimization [17,18], and integration with grid [19]. And other methods to reduce the current peak or shift the current peak and fill the current valley [20,21] also attracted researchers' attention. For example, Wu et al. proposed an artificial potential field

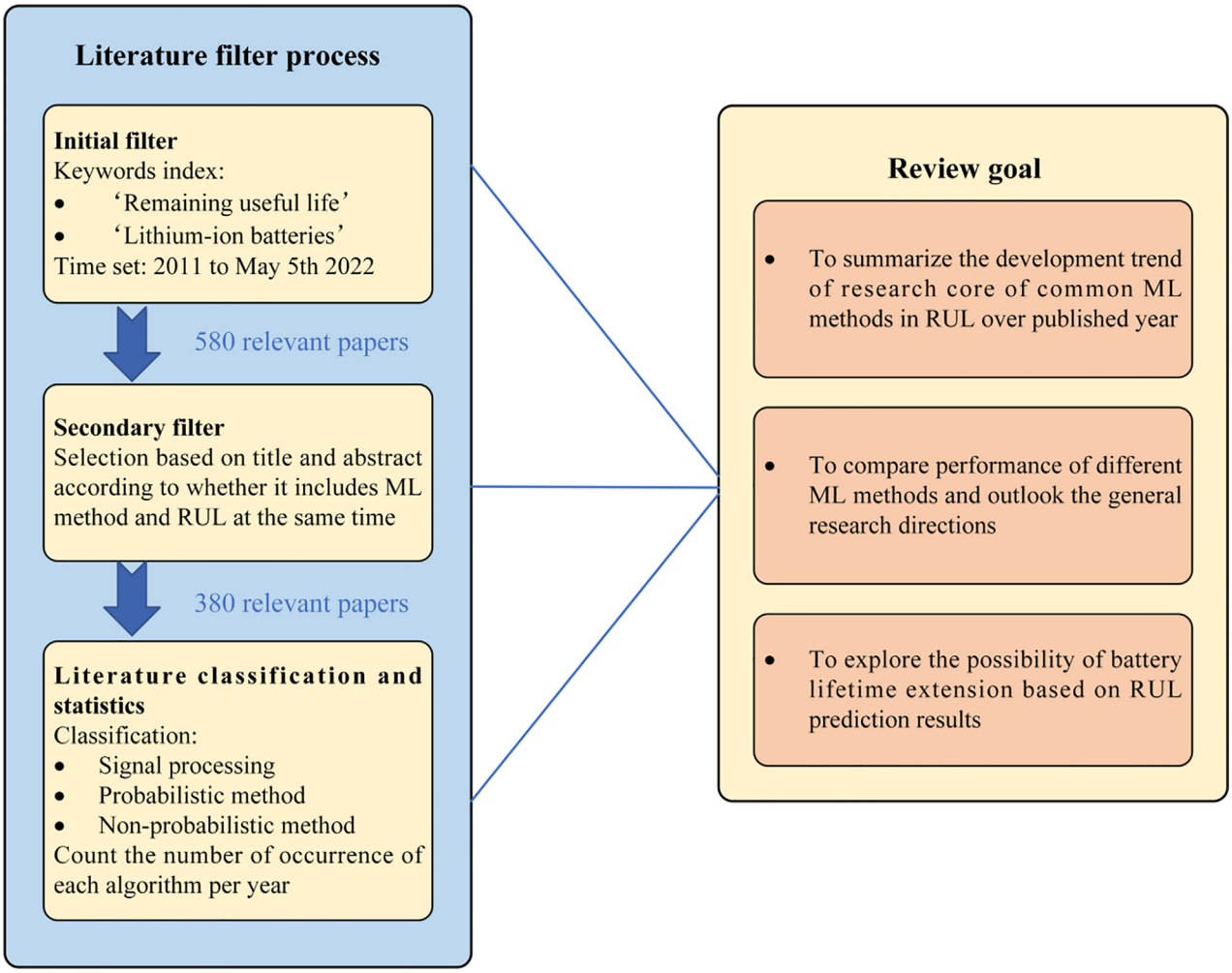

Fig. 1. The flowchart of the literature filter process and review goal of this study.

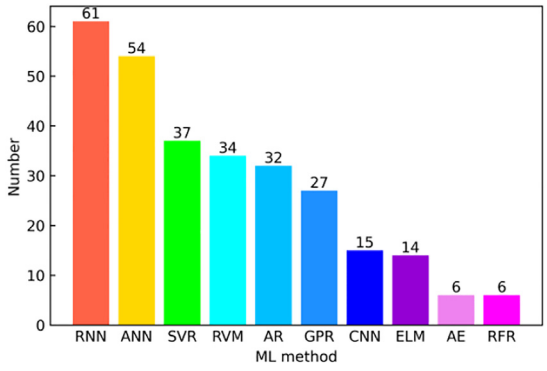

Fig. 2. The total number of occurrences of various algorithms during a given time.

strategy for a battery/supercapacitor hybrid energy storage system to reduce the battery current. The battery capacity loss can be reduced by more than 15% under urban driving conditions [16]. Furthermore, Huang et al. demonstrated battery lifetime extension using pulsed current charging [10]. However, most of the available literature nowadays do not consider using RUL result to extend battery lifetime which is of vital importance for battery application. Therefore, this review article also tries to explore this possibility from the concept and literature perspective.

The main contribution of this work mainly includes:

(1) The application and development of the most used signal processing method for RUL prediction are summarized.
(2) The evolution, over the years, of the research core regarding ML-based RUL prediction is summarized in a unified form, which helps find the recent research cores needed to be improved and combine the advantages of different ML methods.
(3) The methods of battery lifetime extension through optimizing charge profile are summarized and the possibility of battery lifetime extension based on RUL prediction results is explored briefly from literature analysis.

The reminder of this article is organized as follows. Section II presents the general flow of battery RUL prediction, relevant signal pre-processing, and different ML methods as well as their developmental trends. The performances of the different algorithms are compared comprehensively and the limits of each algorithm are discussed in Section III. Section IV gives an outlook on the future direction of the ML method for RUL prediction from a new perspective and explores the possibility of extending battery lifetime based on the predicted RUL results. Finally, Section V concludes this article.

2. Battery RUL prediction

2.1. General flow

The general structure of the RUL prediction based on ML includes three steps as shown in Fig. 3. In the first step, data is collected from battery aging tests or real operations for extracting effective features. The aging features can be classified into direct and indirect types. Temperature, charging or discharging current, and output voltage are common direct features. The time interval of an equal charge voltage difference (TIECVD), incremental capacity analysis (ICA), differential voltage analysis (DVA), and differential thermal voltammetry (DTV) are examples of indirect features; for a thorough review of extracting features, the reader is referred to [22]. In particular, extracting features from EIS data as the input of ML algorithms is becoming popular in recent years. Three common extraction ways are utilized based on EIS data: extracting features from all frequency points of EIS, extracting features from equivalent circuit models which need to obtain the model parameters from the EIS first, and extracting from fixed-frequency impedance [23]. Using ML methods like AE to extract features from EIS is another good way [24]. EIS response is sensitive to temperature, state of charge (SOC), and relaxation effect, which is one research focus and needs to be considered in RUL prediction [25]. Mona et al. quantified the influence of temperature and SOC on the SOH estimation accuracy based on the GPR model and EIS test. In their research, the model without considering both SOC and temperature had the poorest performance [26]. Similar research about the degree of dependence of model accuracy on SOC and temperature was also analyzed in [27]. Correlation analysis is always used to filter noneffective features. Signal pre-processing is adopted sometimes to smooth the capacity or state of health (SOH) curves or enhancement the linear relationship between the capacity and features. Secondly, ML models will be trained to find the relationship between extracted features and the battery RUL. Another way is using the ML model to estimate SOH first and then using the estimated SOH value in the sliding window to further recurse the next SOH until the SOH reaches the predefined threshold value. In fact, almost all algorithms used in SOH estimation can be used in RUL predictions. The main difference between them is the input variable. The input of SOH is usually the features related to SOH, while most of the input of RUL is the previous SOH or capacity which is estimated by the features first (i.e., the latter way we describe). For the first way we introduced, it needs a lot of real RUL data from experiments. The feature extraction is also skillful, and the application conditions are limited. One typical research in this way can be seen in [28], which uses discharge voltage curves from early cycles to predict cycle life based on 124 fast-charging lithium iron phosphate/graphite batteries. Finally, the model will be evaluated to quantify its performance: absolute error (AE), mean absolute percentage error (MAPE), mean square error (MSE), root mean square error (RMSE), mean absolute error (MAE), and max absolute error (MaxAE) are common performance indexes based on the test data set.

2.2. Signal pre-processing before machine learning

For RUL prediction of battery, most researchers mainly focused on the degradation trend of battery. It is easy to track the RUL point of smooth capacity or SOH curve compared with the noised curve. The signal pre-processing method can separate the main trend curve from the noise and fluctuant curve. Besides, the extracted features for RUL prediction often have a linear relationship with the capacity or SOH. If the linearity of them can be enhanced, it will be helpful to predict RUL fast and accurately. Therefore, the main signal pre-processing methods are introduced in the following part.

2.2.1. Empirical mode decomposition (EMD)

EMD is an adaptive and automatic time–frequency signal processing method presented by N. E. Huang et al. in 1998 [29]. EMD decomposes non-stationary signal data into a finite number of high-frequency intrinsic mode functions (IMF) and a low-frequency residual sequence (RES) [29]. In RUL prediction models, the IMF components are usually used to reflect capacity regeneration and local fluctuation in the battery capacity degradation, while the residual is used to reflect the global capacity degradation trend [30]. The typical decomposed result of original capacity degradation data by EMD is shown in Fig. 4. Matti et al. compared the SOH estimation results by seven ML methods through EMD. The gradient boosting has the best result with 0.28 RMSE, while the extra randomized tree has the worst result with 0.43 RMSE [31].

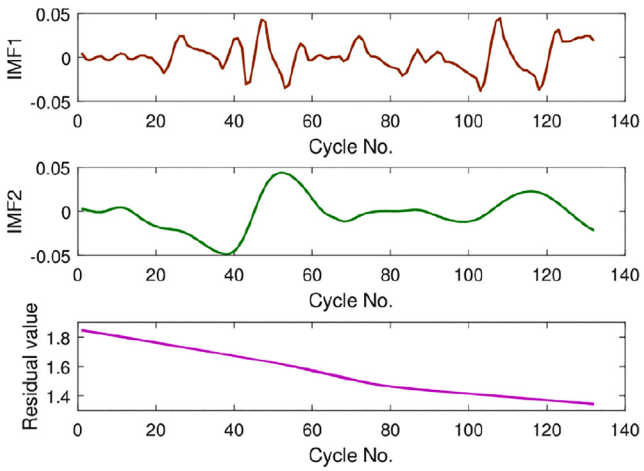

Fig. 4. Typical IMFs and residual result through EMD decomposition [30].

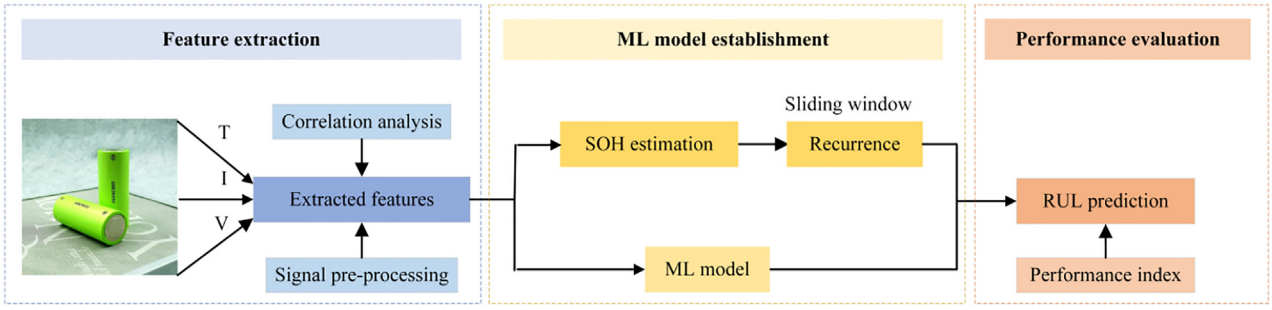

Fig. 3. The general flow of RUL prediction.

The applications of EMD can be classified into two categories. One is for eliminating the problem of capacity regeneration, i.e., ignoring the IMFs. The residue and the original data show a high degree of correlation. For example in [32], Pearson correlation coefficients between the raw capacity and the residue of different batteries were all above 0.99. Therefore, the changes in the predicted residue can truly reflect the changes in battery capacity and remove noise. Another is by considering capacity regeneration and random fluctuation. Hybrid ML methods are usually used. The first ML method (ML1) is applied to fit the IMFs for reflecting the local regression and fluctuation, and the residual sequence will be captured by the second ML method (ML2). The final RUL result is the combination of two ML method results. The common pattern is shown in Fig. 5. Although the latter case may improve the accuracy, the increased calculation resources caused by extra IMFs should be considered.

Determining the optimal number of IMFs to reflect the noise is not easy [33]. The commonly used method to choose the IMF is correlation analysis between IMF components and the original series, and then high relative IMFs are combined to reconstruct. In research by Hao Yang et al., the thresholding denoising method is applied to sift the optimal IMFs, where the amplitude of the IMF is compared with the predefined threshold referring to the denoise condition [34]. The reverse combination method is proposed to determine the optimal IMFs in the research of Yun et al. The difference between RES and the original data was analyzed first in this method. If the difference is small only RES was used. Otherwise, adding the last one or two IMFs on RES [35].

The decomposed signals by EMD mainly include extracted features for RUL prediction (like the raw SOH series, and voltage difference of equal time interval) and preliminary prediction error series (like the error between real data and basic RUL algorithm prediction value). The applications of EMD with different decomposed signals, ML methods, and decomposed result utilization are summarized in Table S1.

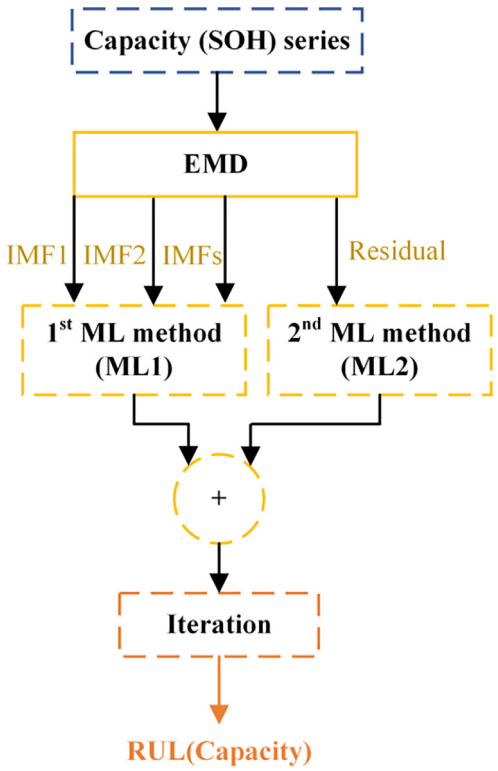

Fig. 5. RUL prediction flowchart considering EMD decomposed results.

Although EMD has the advantages of high efficiency with no need to predefine basis functions, mode mixing of IMFs often occurs in its practical application. It is common to see similar time scales in different IMFs or disparate time scales of oscillations in a single IMF, which lose the physical meaning of IMFs. To solve this problem, ensemble empirical mode decomposition (EEMD), a noise-assisted data analysis method, is proposed. In EEMD, the true IMF components are the average of ensemble trials, where each trial is an EMD result with additional white noise. EEMD can reduce the mode aliasing effect due to the introduction of white noise to the original signal. Its application in RUL prediction is also shown in Table S1.

To solve the mode aliasing problem of EMD and overcome the low decomposition efficiency of EEMD, the complementary ensemble empirical mode decomposition (CEEMD) is proposed by Torres [36]. Complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) is also used to decompose the original data, which adds adaptive white noise at each stage of the decomposition and calculates a unique residual signal to obtain the mode of each component. It is worth noting that the Pearson correlation of EMD result sometimes is higher than that of CEEMDAN [37], so the selection of EMD and its variant should be considered from correlation and efficiency comprehensively.

2.2.2. Variational modal decomposition (VMD)

VMD algorithm adaptively decomposes the signal into a series of sub-signal IMFs, where the summation of the IMFs is the original signal (not the IMFs and RES like EMD) and each IMF has a limited bandwidth in the spectral domain. It overcomes the disadvantages of the end effect and IMFs aliasing in the EMD. In RUL prediction, VMD is used to decompose the battery capacity degradation data into the degradation trend series and other fluctuation series. Through the VMD, the prediction accuracy of RUL cycles is improved from 78% to 93% for NASA 5# battery before denoising and after denoising [38]. The application of VMD is summarized in Table S2.

2.2.3. Box-Cox transformation (BCT)

Owing to its ability to improve the additivity, normality, and homoscedasticity of observations, BCT is widely used to enhance the linear relationship between the features and RUL. The basic form of BCT is:

$$y^{(\lambda)} = \begin{cases} \frac{y^{\lambda}-1}{\lambda}, \lambda \neq 0 \\ lny, \lambda = 0 \end{cases} \quad (3)$$

where y is the observation. The parameter λ is commonly determined by using profile likelihood and a goodness-of-fit test. After transformation, Pearson correlation analysis is used to quantitatively evaluate the linear relationship between the independent variable (extracted features) and observation (capacity in most cases). In most cases, capacity is the transformed variable. Another pattern for using BCT in RUL prediction is shown in Fig. 6, where BCT is used to transform features, and then improve the linear relationship between features and capacity. The new HI is now used in an ML method. The applications of BCT are summarized in Table S3 including the independent variable, the transformed variable, and the prediction variables in the prediction algorithm.

2.2.4. Wavelet decomposition technology (WDT)

Wavelet analysis, also related to wavelet decomposition technology (WDT), can be used to decompose the non-stationary signal into an approximate part and a detail part through the scaling function and wavelet function [39]. The wavelet function serving as a high-pass filter can generate the detailed version of the given signal (like the local regeneration trend and fluctuations in the SOH

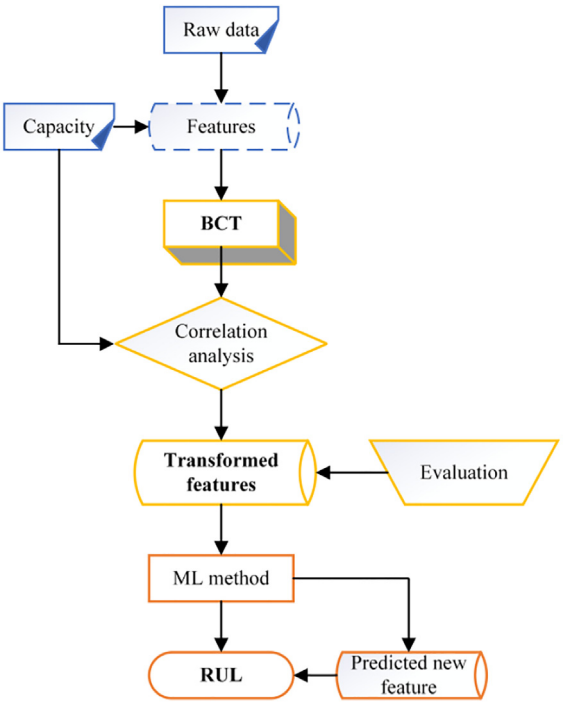

Fig. 6. RUL prediction flowchart considering BCT.

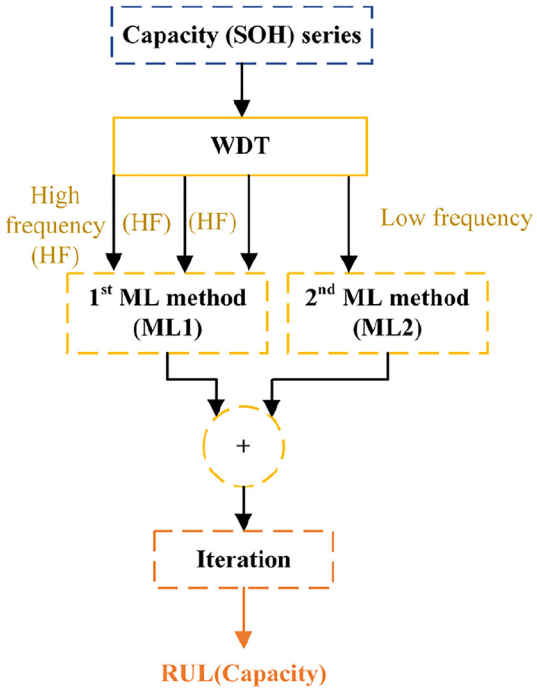

Fig. 7. RUL prediction flowchart considering WDT.

series), while the scaling function serving as a low-pass filter can generate the approximated version of the given signal (like the global degradation trend in SOH series). In RUL prediction, WDT is used to separate the global degradation and local regeneration of a battery capacity series. The approximated part is treated as a global degradation trend and the detailed part at different scales corresponds to local regeneration trends and fluctuations. For the improvement of the prediction performance, the approximated part is used in the first ML method, and the detailed part is used in the second ML method. Then the results from two ML methods are combined and iterated to RUL finally, as shown in Fig. 7. It can be found that the general idea of using WDT to predict RUL in Fig. 7 is similar to that of using EMD in Fig. 5. The two methods are different in the decompose way. EMD is an adaptive decompose method and easy to generate the mode mixing phenomenon when decomposing the capacity or SOH, while WDT has fixed bandwidth and its speed is fast.

In practical application, the discrete wavelet transform (DWT) is used, where the original signal can be decomposed into several signals with different scales (i.e. global degradation trend and local regeneration trend and fluctuations in SOH series) and can reconstruct the signals using inverse DWT. Wang et al. proposed an RUL prediction function based directly on the decomposed terminal voltage using DWT [39]. It is worth noting that the type of wavelet function and the number of decomposition levels will affect the performance of the model.

Wavelet packet decomposition (WPD), able to eliminate the noise of charge–discharge cycle data is an upgrade of the wavelet transform [40]. The main idea of the algorithm is based on discrete wavelet transformation, in each level of the decomposed signal. That is, in addition to the decomposition of the low-frequency sub-band, the high-frequency sub-band is also decomposed. The theory of wavelet packet energy entropy (WPEE), proposed by Chen et al., is the combination of WPD and information entropy theory [41]. The application of WDT and its variant is summarized in Table S4.

2.3. Machine learning methods

Fig. 8 shows the developmental trend of each ML algorithm (used for RUL estimation) and their popularity year-by-year. It is obvious that RNN algorithms were first applied to battery RUL prediction in 2013, and then it has gradually become the most frequently used method. The year that the ANN algorithm was first used in RUL prediction is 2013, it also gains attention. However, its usage drops in 2020, then its usage frequency goes up again. SVR and AR are the first used algorithms in RUL prediction occurring in 2012. The algorithms for which the total number of occurrences is larger than 10 in the given time (high popularity) will be further reviewed in the next subsections.

2.3.1. Non-probabilistic method

2.3.1.1. Recurrent neural network. RNN algorithm is suitable for processing time series data because its hidden layer makes the current output related to the previous output. As shown in Fig. 9, its basic

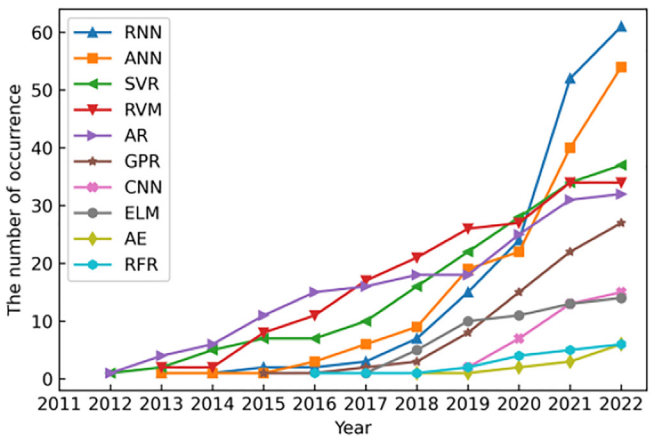

Fig. 8. Evolution trend of ML algorithms for RUL prediction.

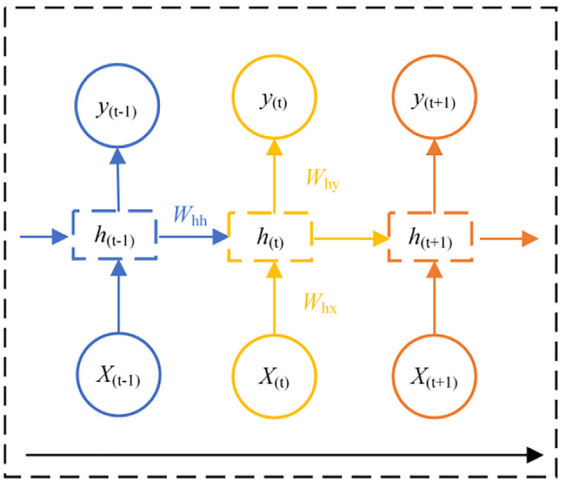

Fig. 9. The basic structure of RNN. W_{hx} is the weight matrix connecting the input layer $X_{(t)}$ and hidden layer $h_{(t)}$. W_{hh} is the weight matrix connecting the hidden layers at different times. W_{hy} is the weight matrix connecting the hidden layer $h_{(t)}$ and output layer $y_{(t)}$.

structure contains an input layer $X_{(t)}$, hidden layer $h_{(t)}$ and output layer $y_{(t)}$. Weight matrix W_{hx} connects the input layer $X_{(t)}$ and hidden layer $h_{(t)}$. Weight matrix W_{hh} connects the hidden layers at different times. Weight matrix W_{hy} connects the hidden layer and output layer. This structure makes sure that RNN can use past and current information to predict the future. The battery degradation data collected from charging and discharging cycles can be seen as time series data, making RNN suitable for RUL prediction. Kwon et al. used RNN to predict the battery RUL by learning internal resistance [42]. Ansari et al. proposed multi-channel input RNN to predict RUL based on the combined datasets comprising ten samples of voltage, current, and temperature for each cycle [43]. There are many variants in RNN, and their evolution over the published year is shown in Fig. 10**.** ESN is a fast and efficient type due to its simple structure. Besides, due to the memory part of the sequence, the performance and accuracy of basic RNN on a long sequence are poor. Therefore, LSTM is proposed to use a gate mechanism to control the information flow and solve the problem of the vanishing gradient in the RNN. The vanishing gradient means the gradient will disappear during the back propagation process, which slows the weight training down and is unable to learn the long-term memory effect. Further, GRU is a variant of LSTM, which integrates the forget gate and input gate of LSTM into an update gate and introduces an additional reset gate to control the flow of information. Various RNN models, including basic RNN, LSTM, and GRU are built to forecast the RUL of lithium-ion batteries. It is found that the LSTM layer has the optimal RUL prediction results based on the same datasets [44]. More details about ESN, LSTM, and GRU are discussed in the remainder of this subsection.

Echo state network:

The ESN is composed of an input layer, a reservoir layer, and an output layer, as shown in Fig. 11. In the figure, W^{in} is the input weight matrix. W^{re} is the internal connection weight matrix. W^{back} is the feedback matrix returning the output of ESN to the reservoir layer. W^{out} is the output weight matrix. The reservoir layer consists of many sparsely connected neurons and thus has

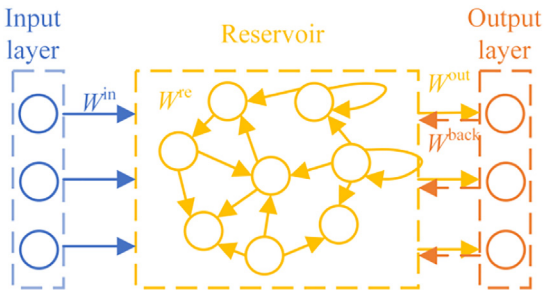

Fig. 11. The structure of ESN. W^{in} is the input weight matrix. W^{re} is the internal connection weight matrix. W^{back} is the feedback matrix. The input layer for RUL prediction is always extracted features and the output layer is RUL.

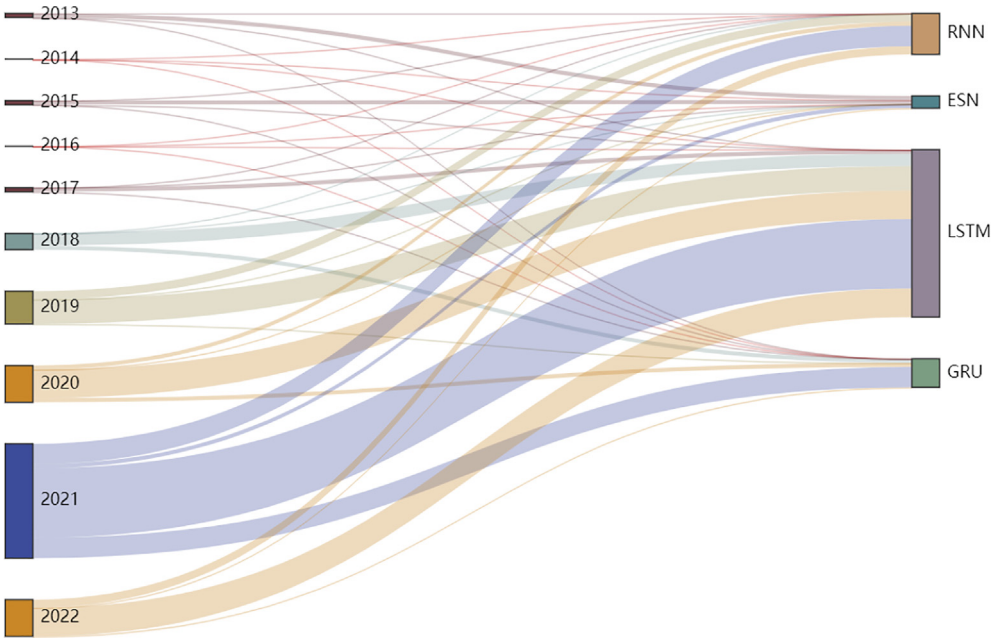

Fig. 10. The evolution of RNN variants over the published year.

the function of short-term memory by adjusting these neurons [45]. The ESN is similar to the ELM because both have a very simple and the weights of the input layer and the reservoir layer are random and fixed after initialization. That is, only the output weights between the reservoir and output layer need to be trained.

The research core of ESN includes deep ESN, monotonic ESN (MONESNs), and ensemble leaning. Catelani et al. proposed the deep ESN, where the typical ESN adds multiple reservoir layers to improve the ability to capture the nonlinear trend of batteries. In their research, state-space estimation was used first to generate a big training dataset (capacity and cycles) for deep ESNs. A genetic algorithm was employed to optimize the number of nodes of the reservoir and the linear-weighted PSO was utilized to optimize the network parameters [46]. Ji's team and Liu's team both presented the monotonic ESNs in their research by adding monotonic constraints in the ESN output weight training process to make the output of ESN decrease with the decrease of the input data. In Ji et al.'s research, a self-adaptive differential evolution (SADE) algorithm was used to optimize the parameters of the ESN [45]. Liu et al. created an ensemble of many MONESN sub-models to provide uncertainty on the predictions made by the MONESN and increase the stability of the network [47,48].

If the readers are interested in the theory and applications behind the ESN, one guide [49] is dedicated to this method.

Long short-term memory:

LSTM uses gate structure, a method that can control the information flow, to solve the vanishing gradient problem. LSTM consists of three gates: input gate, output gate and forget gate, its structure is shown in Fig. 12. x_t, h_t and C_t is the input, output, and cell memory of the current time step. h_{t-1} and C_{t-1} is the output and cell memory of the previous time step. Sigmoid is the activation function. LSTM was introduced to capture the long-term battery degradation tendency in 2017 [50] and 2018 [51] by Zhang et al. They found that LSTM can achieve early RUL prediction because LSTM only needs 20%-25% of the entire battery degradation data for accurate RUL prediction. A similar conclusion was also obtained by Tong et al [52]. The training method, over-fitting, and uncertainties of LSTM were mentioned in the publications of the above two teams. The most common training methods of LSTM include adaptive moment estimation (ADAM) [53] and resilient mean square back-propagation (RMSprop). The training method for LSTM is chosen based mainly on the convergence speed. The common methods to avoid over-fitting include L2 regularization and dropout. Furthermore, the main method often quantifying the uncertainty of the prediction made by LSTM is via Monte Carlo (MC) simulations.

In the input features of LSTM, the one-to-one structure is replaced by a many-to-one structure [54]. The many-to-one structure refers to the use of multiple input vectors simultaneously, such as current, voltage, and temperature, while the one-to-one structure uses a single input vector of historic capacity. It has been proven that many-to-one structures can improve RUL prediction accuracy even in the presence of capacity regeneration [55]. Zhang et al. proposed a BLS-LSTM model which combined LSTM and Broad Learning System (BLS) which has the additional enhancement nodes from the initial feature input. Based on this, the more effective feature can be extracted [56]. Furthermore, with the adoption of the adaptive sliding window, LSTM can learn the local fluctuations and the long-term dependencies simultaneously [57].

Besides, another type of LSTM that has been used in RUL prediction is the Bidirectional LSTM (Bi-LSTM) [34]. Bi-LSTM uses 2 independent LSTM layers to learn in both the forward and backward direction. Yang et al. thought that the battery degradation series signals were related to the current time both in forward and backward directions, so they used Bi-LSTM information to extract more information from the same signal [34]. Yuhuang et al. also used Bi-LSTM in the discriminator of the generative adversarial network (GAN) to predict RUL [58].

The research core of LSTM centers around gate optimization, algorithm combinations (RUL prediction based on the signal decomposition, measurement variable as filter algorithm, combination with CNN), ensemble learning, and transfer learning. Its development trend is shown in Fig. 13.

Gate optimization of LSTM is an effective strategy to improve RUL accuracy. The information controlled by the gates is an essential part of LSTM. Li et al. coupled the input and forget gate through a fixed connection, which selects old and new information at the same time. They also conducted the element-wise product between the new inputs and historical cell state and added a peephole connection to the output gate to mine more information and

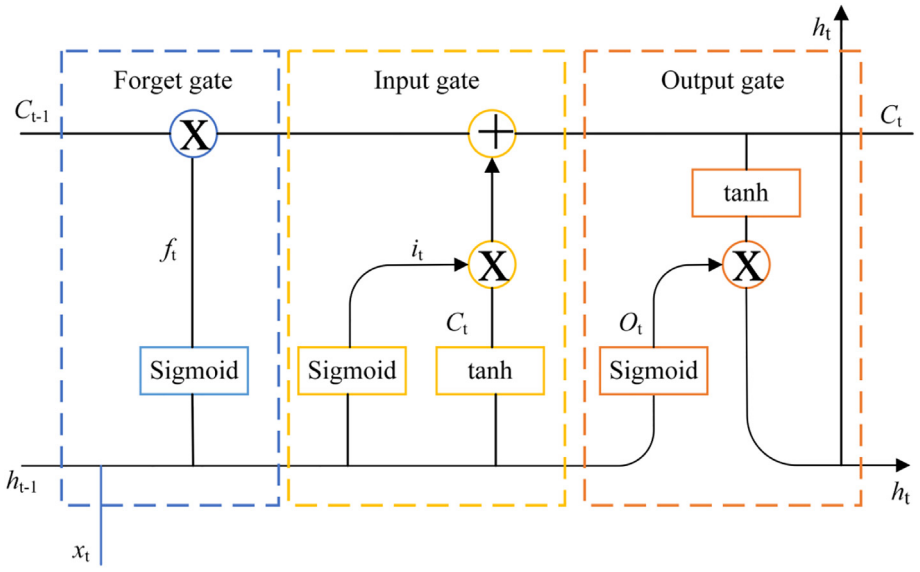

Fig. 12. The structure of LSTM. x_t, h_t and C_t is the input, output, and cell memory of the current time step. h_{t-1} and C_{t-1} is the output and cell memory of the previous time step. Sigmoid is the activation function.

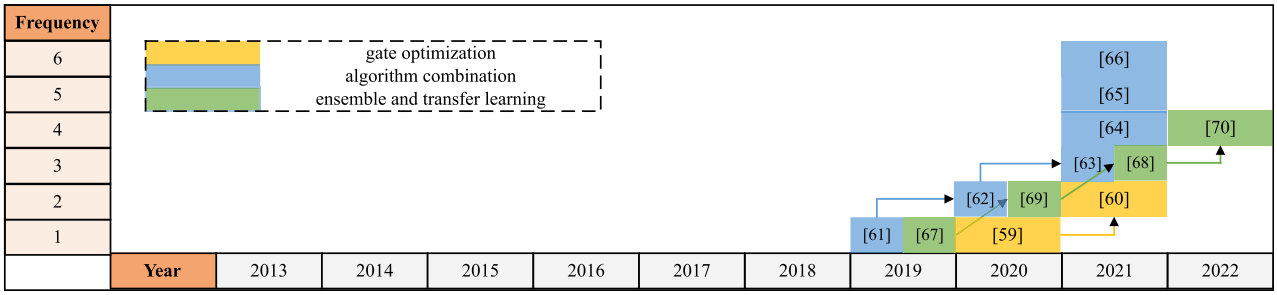

Fig. 13. The development of the research core of LSTM.

reduce the error [59]. Lin et al. connected the input gate and output gate to better control the output information of the memory cell. They also combined the forget and input gates into a single update gate, which can achieve the goal of forgetting some information and adding inputs simultaneously [60].

LSTM algorithm combination mainly includes the prediction of decomposed signal through different signal decomposition methods like EMD [61], more details have been discussed in the above 'Signal pre-processing' section, as the measurement variable of PF or KF and combining with CNN through different connection way. UKF or KF shows advantages in processing non-linear problems. However, the problem is that the measurement values are not easily and precisely obtained from the battery electrochemical model. For this, Cui et al. incorporated LSTM and RBF NN in UKF, where the LSTM predicted capacity is used as the measurement vector and RBF NN is for approximating the traditional exponential model as state transition [62]. Zhang et al. used PF to update the exponential model of LIBs, where LSTM was adopted to learn the degradation model and every particle was propagated using the state transition learned by LSTM [63]. Hu et al. also used the LSTM prediction mean values as the future measurement value of PF. The mean was obtained by the Monte Carlo-dropout method by processing the LSTM model iterating 100 times each step [64]. LSTM is often combined with CNN to extract spatial and temporal information from the data. More details are discussed in the sub-section 2.3.2.4 'Convolutional Neural Network' part. The combined way includes series [65] and parallel connections [66].

Ensemble learning can give a more reliable prediction result. Liu et al. proposed to use Bayesian model averaging (BMA) to ensemble LSTM sub-models obtaining higher accuracy and uncertainty representation. The LSTM sub-models adopted different sub-datasets derived from the degradation of training data [67]. Song et al.'s team [68] and Wang's team [69] both presented the ensemble model based on stacked LSTM to predict the RUL, where prediction values from individual LSTM were used as input to the ensemble model. Pan et al. proposed LSTM based on transfer learning (TL-LSTM) to predict lithium-ion battery capacity and RUL under different working conditions, where the LSTM layers are frozen in the training process of the target domain. The predicted value from TL-LSTM was also used as the measurement for the PF updation [70].

Gated recurrent unit:

GRU is similar to LSTM but has two gates combining the forget and input gates of LSTM to an update gate used to control the influence of the previous information on the current information and using a reset gate to control the extent of ignoring previous information. Profiting from a simpler structure than LSTM, GRU is faster to train while still being able to mitigate both exploding and vanishing gradients.

The research on GRU in battery RUL is majorly focused on the basic application, deep information extraction, and transfer learning.

Song et al. utilized the GRU to establish the battery RUL prediction method, where their number of hidden layers was four and the maximum error of the RUL prediction on the NASA B0006 battery was 11 cycles [71]. Ardeshiri and Ma et al. adopted the ADAM algorithm to optimize the network, which can reduce the training time and the influence of the learning rate. They also proposed an early stopping technique to deal with overfitting [72]. Wei et al. used GRU to predict RUL with Monte Carlo Dropout (MC_dropout), where MC was mainly used to generate the probability distribution and 95% confidence interval of the RUL prediction point. The dropout method applied to the weights of the hidden layers is to avoid over-fitting [73].

In order to extract more information on battery degradation, Wang et al. proposed an adaptive sliding window GRU. Characterized by the memory units and gate mechanism, the GRU can capture the long-term dependencies of their feature. The adaptive sliding window can learn the local regenerations and fluctuations, as the size of the sliding window updates according to the variation of the feature [74]. Tang et al. used IRes2Net-BiGRU-FC to predict different frequency data through CEEMDAN. In IRes2Net-BiGRU-FC, Bidirectional Gated Recurrent Unit (BiGRU) can capture battery capacity information in the two directions of the past and the future simultaneously, so it increases the diversity of characteristic information [75].

Transfer learning can reduce training data and training time for online applications. A combination of transfer learning and GRU was proposed to predict the online RUL of batteries based on having similar degradation trends [76]. The similarity of degradation trend for different batteries is compared by Euclidean distance.

2.3.1.2. Support vector regression. SVR is suitable for RUL prediction because it can map the non-linear relationship between input and output data based on structural risk minimization [77]. The key is using the kernel-trick to transform the complex nonlinear problem into a simple linear problem. The general form of the SVR model can be expressed as:

$$f(x) = w^T \phi(x) + b \tag{4}$$

where x is the input value. $f(x)$ is the target value. w is the weight vector. $\phi(x)$ is the kernel function matrix. b represents the bias term.

Zhao et al. utilized the SVR to predict RUL integrated with feature vector selection (FVS). The FVS-SVR algorithm can narrow down the search space of support vectors by removing redundant data. The time interval of an equal charging voltage difference (TIECVD), and the time interval of an equal discharging voltage difference (TIEDVD) are the extracted features in their research [78].

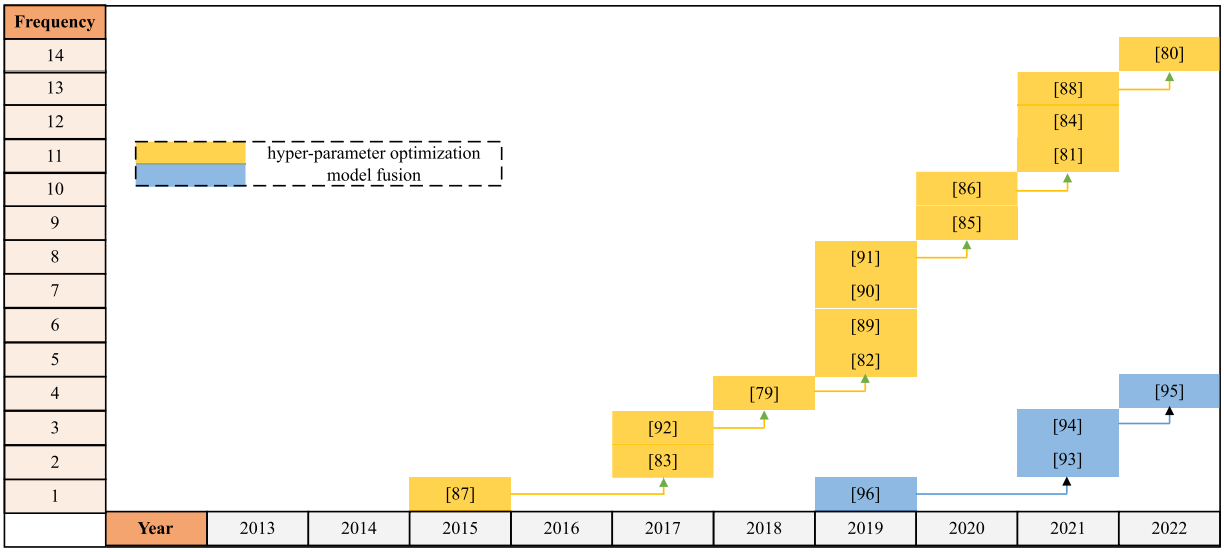

Fig. 14. The development of the research core of SVR.

The research about SVR in RUL prediction is mainly focused on hyper-parameter optimization and model fusion as shown in Fig. 14.

The most common hyper-parameters include the penalty factor, the band-size (controlled by epsilon), and any kernel function parameters. A variety of optimization methods have been applied to optimize the hyper-parameters: differential evolution [79], particle swarm optimization [80-83], whale optimization (WOA) [84], genetic algorithms [85-87], gray wolf optimization (GWO) [88], improved ant lion optimization (ALO) [89], artificial bee colony (ABC) [90], improved bird swarm (IBS) [91], and grid search [92].

The SVR is often used as the measurement equation in particles [93] and Kalman filters [94] to realize the step-ahead prediction. Furthermore, it has been used as an auxiliary algorithm to improve the prediction performance of a BiLSTM-AM model, where the output of the SVR is a prediction of the average temperature [95]. SVR has also been fused with other methods like multi-layer perceptron (MLP) based on the weighting principle [96].

The combination of multiple SVR with different hyper-parameter for different regions of one training data [97] or SVR with mixed kernel function [98] may be the possible future direction for SVR. The SVR algorithm implementation in the vehicle's central controller considering load conditions also should be researched further [99].

2.3.1.3. Auto-regression. AR is a time series model to predict the near-future values in the series. It has the advantage of long-term trend prediction for time series analysis [100]. For example, Vilsen et al used AR to capture the long-term behavior change of the internal resistance of the battery [101]. Its basic function is as follows:

$$X_t - \varphi_1 X_{t-1} - \varphi_2 X_{t-2} - \cdots \varphi_p X_{t-p} = \varepsilon_t \tag{5}$$

where p is the order of the AR model, $\varphi_i (i = 1, 2, \cdots, p)$ are parameters of the AR model, and ε_t is the stationary white noise with mean zero and variance. The determination of the order of the AR model is important to the prediction accuracy, and common determination methods include the Box-Jenkins method, Akaike information criterion (AIC), and Bayesian information criterion (BIC). Long et al. proposed the RMSE order determination method using the PSO method to search the optimal order of the AR model, and the order can adaptively vary with the updated data [102]. The parameter identification of the AR model is relatively simple, and methods include least square estimation, maximum likelihood estimation, Yule–Wallker method (autocorrelation method), and the Burg method.

The main research core of AR is nonlinear degradation implementation and algorithm combination, as shown in Fig. 15.

The degradation process of lithium-ion batteries is nonlinear because the degradation rate usually accelerates when the cycle number increases. For improving the nonlinear ability of the AR model, the nonlinear accelerated degradation factor is introduced which is related to the degradation cycle and time-varied. In practice, the factor is defined as the function of the prediction step. Therefore, it is called the nonlinear degradation AR (ND-AR) model [103]. Guo et al. modified the factor after analyzing the ND-AR model and proposed a nonlinear scale degradation parameter-based AR

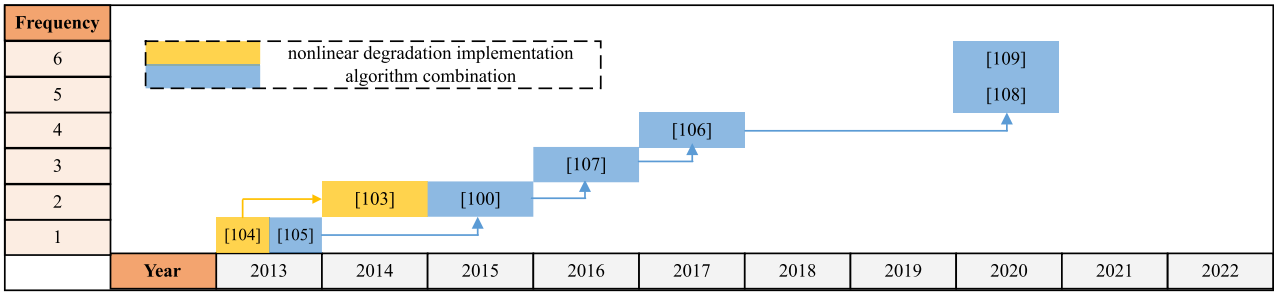

Fig. 15. The development of the research core of AR.

(NSDP-AR) model, where the factor is related to the current percentage of life-cycle length and the related parameters in the formula to calculate the factor are obtained using an EKF [104].

AR models are usually combined with different PF, where the output of the AR model is used as the observed or measured value in PF [105] to improve the long-term prediction and iterative prediction of PF. For this, in addition to the ND-AR model, the iterative nonlinear degradation autoregressive (IND-AR) model was proposed [106]. Besides the basic PF, there are also regularized PF (RPF) [107] and PF based on the deformed double exponential empirical degradation model [108]. Further, AR models have been combined with RVM, where the uncertainty quantification is based on the mixture distribution estimation [100]. Another proposing combination method was proposed by Lin et al [109]. In their research, the degradation trajectory is regarded as the multiple-change-point linear model rather than the fixed shape. The AR model with covariates is used to obtain the slope change between segments.

2.3.1.4. Convolutional neural network. CNN generally consists of convolutional layer, pooling layer, and fully connected layer, as shown in Fig. 16. It uses a filter to perform a convolution operation to extract spatial information from the previous layer. The extracted neurons are referred to as the receptive field, and the convolution is performed by:

$$F(i,j) = \sum_m \sum_n Z(i-m, i-n) * K(m,n) \quad (6)$$

where F is the obtained feature map, Z is the input matrix, and K is the filter. Then the pooling layers are used to extract local features, and the fully connected layer can be used for regression.

The main purpose of using CNN is to extract the spatial information or interrelations of the inputs and reduce it to a lower dimensional space [110]. Hsu et al. used two CNNs to reduce the dimension of the input data and combined these new features with human-constructed features. The MAPE by their deep neuron network (DNN) to predict battery RUL is 6.46% with only one cycle as testing [111]. Rui et al. proposed one semi-supervised method of capacity degradation estimation. Raw EIS data were directly fed into the CNN for feature extraction and then capacity was estimated without corresponding capacity labels [112].

The research core of CNN mainly includes hyperparameter optimization, combining CNN with LSTM, dilated CNN ensemble learning, and transfer learning, as shown in Fig. 17.

The hyperparameter is set before the training process from different datasets, which can influence the performance of CNN. The optimization methods used for hyperparameters include Bayesian optimization [113] and SSO [114].

Considering that LSTM can learn the temporal correlations between the past and future RUL, CNN is often integrated with LSTM to learn the spatial and temporal information of input data at the same time [113]. Zraibi et al. used capacity data at each cycle to train the proposed CNN-LSTM-DNN model to improve the accuracy of battery RUL prediction compared with single LSTM and CNN-LSTM models, the RMSE was 0.0204 [65]. However, it is worth mentioning that the CNN, LSTM, and integrated model required a lot of data to achieve this higher accuracy. For this, Ren et al. integrated an autoencoder with the CNN-LSTM model, where the autoencoder was used to increase the dimension of the data fed into the CNN. An autoencoder learns the time-domain characteristics between adjacent charging and discharging cycles from their analysis [66]. Zhang et al proposed the usage of CNN-LSTM in generating realistic time-series with convolutional recurrent GAN. In their model, the generative model took full use of the advantage of CNN extracting spatial information and the advantage of LSTM extracting the temporal information, enriching the data and reducing the prediction error of the model [115]. Yang et al. further combined CNN with Bi-LSTM to improve the learning effect and the generalization ability [34].

Compared with regular CNN which has a linearly growing number of receptive fields as the number of layers increases, the dilated CNN has an exponentially growing number of receptive fields, as

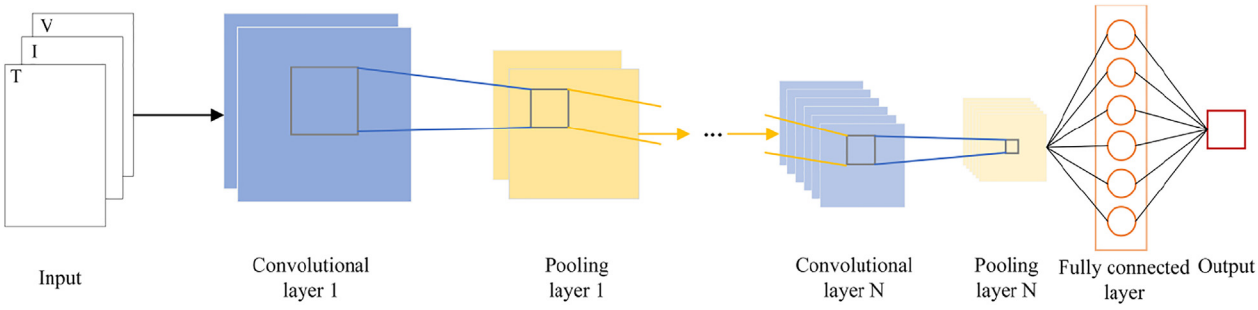

Fig. 16. The basic structure of CNN.

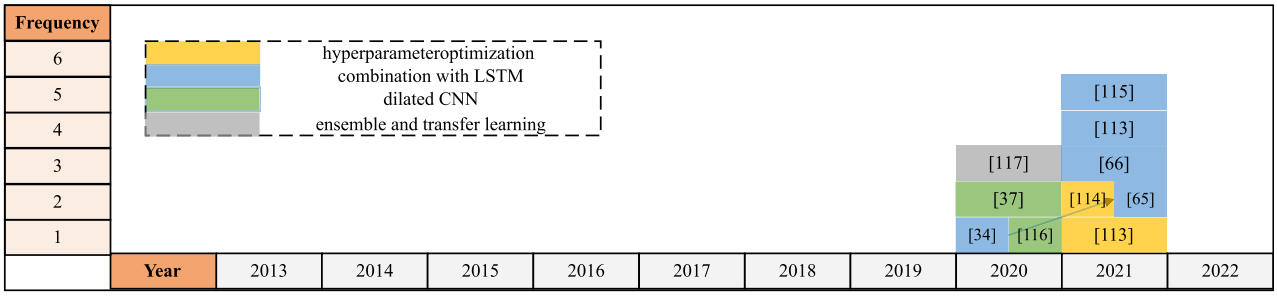

Fig. 17. The development of the research core of CNN.

shown in Fig. 18. Hong et al. proposed a framework using 5 dilated CNN layers which can improve the RUL prediction performance using less than four operation cycles [116]. Zhou proposed a temporal convolutional network (TCN) for battery SOH monitoring, where causal convolution and dilated convolution techniques were used in the model to improve the ability to capture local capacity regeneration, improving the overall prediction accuracy of the model [37].

Transfer learning is the way to make up for a small dataset and improve the training efficiency for CNN. In the research of Shen et al., a 10-year daily cycling data of eight implantable Li-ion batteries was used as the source dataset to pre-train eight deep convolutional neural network (DCNN) models. Then the obtained parameters of the pre-trained DCNN models were transferred to the target domain, which was used to generate eight DCNN with transfer learning (DCNN-TL) models. These DCNN-TL models were finally integrated into an ensemble model called the DCNN with ensemble learning and transfer learning (DCNN-ETL). The proposed DCNN-ETL method has higher accuracy and robustness than GPR, RF, and DCNN [117].

In the future, how to use limited datasets to achieve a fast and stable RUL prediction is still worth keeping an eye on.

2.3.1.5. Extreme learning machine. An extreme learning machine (ELM) is a kind of feedforward neural network, which contains a single hidden layer between an input layer and an output layer. The weights and biases between the input and hidden layer are generated randomly and, therefore, do not need to be trained. That is, the output weights are the only parameters that need to be trained during the learning process. The training of the output weights is usually performed using the generalized Moore-Penrose matrix inverse, so it has the advantages of fewer training parameters and fast learning speed. For online and fast RUL prediction, ELM could be considered.

There are many applications of ELM in the field of battery RUL prediction. Roozbeh et al. introduced an integrated ELM-based framework for RUL prediction in situations of missing observations or features [118]. The degradation path of the lithium battery can be approximated as having two different degradation rates (phases), Chen et al. adopted the two-phase Wiener process to derive the mathematical formula of RUL and an ELM to construct the feature of RUL and detect the change point of the model by the feature adaptively [119].

The research focus of ELM mainly includes the initial optimization of the input parameter, online sequential ELM (OS-ELM), and quasi-RNN. It is summarized in Fig. 19.

As mentioned before, the input weights and the bias of ELM are given. In order to improve the accuracy, PF [120], improved PSO with considering mutation factors [121], Genetic algorithm ant algorithm (GAAA) [122], and Hybrid Grey Wolf Optimization (HGWO) (which combines differential evolution (DE) and grey wolf optimization (GWO) [38]) are used to optimize initial weights and bias.

If the ELM model parameters can be updated based on the new data, rather than the retraining model, the prediction time of RUL will be reduced. In order to reduce the required training data to train ELM, an online sequential-extreme learning machine (OS-ELM) is proposed. It consists of 2 phases: the initialization phase and the sequential learning phase. The function of the sequential learning phase is to use the new data samples to update the output weights obtained in the initial phase. Tang et al. proposed a faster online sequential pooling extreme learning machine (OS-PELM) where the full connection between the input layer and hidden layer is replaced by convolution and pooling based on the

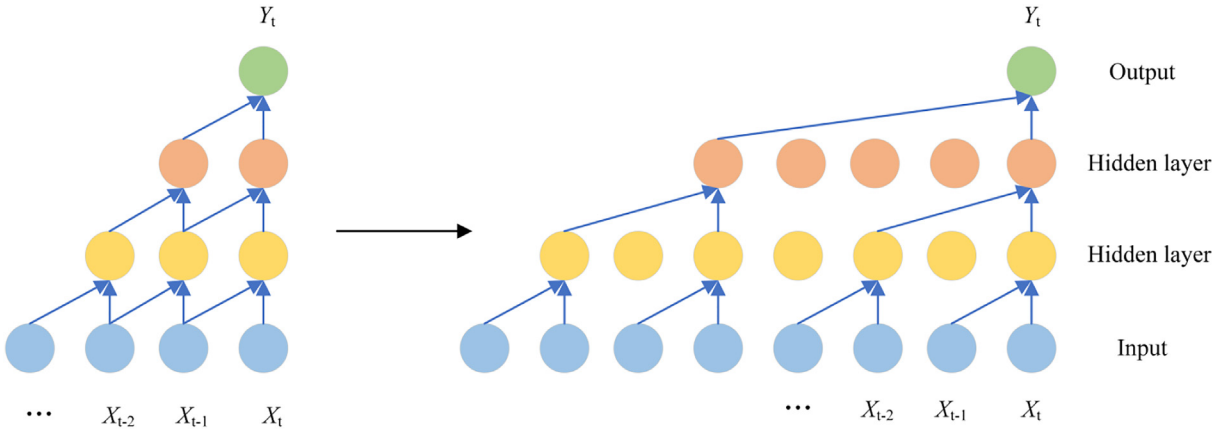

Fig. 18. From regular convolution to dilated convolution. X_t is the input and Y_t is the output.

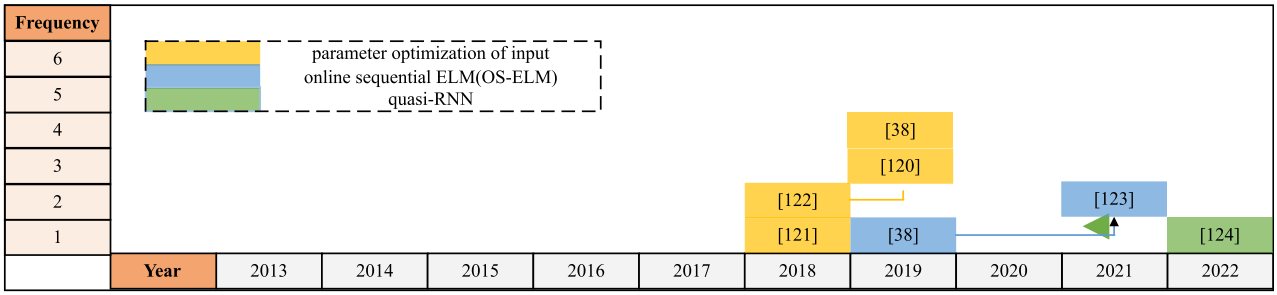

Fig. 19. The development of the research core of ELM.

OS-ELM [123]. Fan et al. further proposed Forgetting Online Sequential Extreme Learning Machine (FOS-ELM). Besides the dynamic incremental update function applied when new data is received, it also includes a forgetting mechanism that will gradually ignore any outdated data in the learning process [38].

Zhang et al. proposed a deep adaptive continuous time-varying cascade network based on extreme learning machines (CTC-ELM) under the condition of a small amount of data, as shown in Fig. 20. A CTC-ELM is similar to the RNN in terms of the time-varying nature because the input of each sub-network is composed of the output of the previous network and the original time series data. But it retains the advantage of ELM of being fast [124].

Although the ELM has many advantages, its shallow structure may restrict its ability to capture the features of high-dimensional data and not meet the requirement of increasing-size battery datasets of increasing size. Therefore, the broad learning extreme learning machine (BL-ELM) is proposed by Ma. The BL-ELM does not increase the number of layers but expands the input layer by widening the number of nodes in this layer. The ability of ELM to capture effective feature information in big data is greatly improved [125].

2.3.2. Probabilistic methods

2.3.2.1. Relevance vector machine. RVM is a sparse probability model based on Bayesian inference. The general form of its model is similar to that of SVR and can be expressed as Equation (4). Different from SVM, RVM is not restricted by Mercer's theorem. RVM owns the advantages of uncertainty expression and can establish

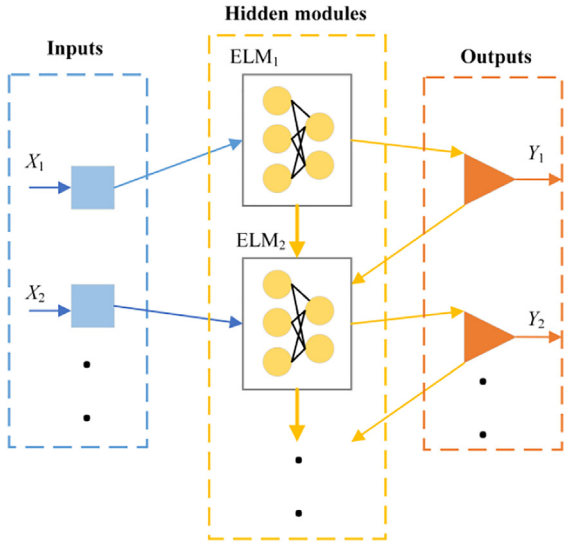

Fig. 20. The structure of CTC-ELM.

multiple kernel functions. In addition, RVM reduces computation because of the smaller number of hyper-parameters and the sparsity in the number of relevance vectors [126]. In online prediction applications, relevance vectors (RVs) are the capacity degradation data of lithium-ion batteries in specific cycles corresponding to non-zero weights containing the core information of the data [127]. From this point of view, it can be regarded as a dimension-reduction method due to the smaller number of sample points at the end. Zhang et al. used RVM to extract the features as relevance vectors which were then used to fit the aging exponential model [128].

The RVM used in RUL prediction is to build the relationship between features and capacity, where the common choices for features include the sample entropies of current and voltage [129], charging current change rate in CV stage (CCCR_CV) [130], constant current charge time (CCCT) and ambient temperature [131], the duration of equal discharging voltage difference (DEDVD) and the duration of equal charging voltage difference (DECVD) [132]. RVM is usually integrated with the Kalman filter (KF) or unscented Kalman filter (UKF) [133]. Zheng et al. used RVM to predict the future evolution of the UKF residuals [134]. RVM was also used to predict the new error series to correct the prediction result by UKF [33]. For giving the posterior probability distributions of the output, RVM is always used to fuse the outputs of different methods without probability distributions or different prediction starting points to improve the overall prediction performance [32].

As for the evolution of the research core of RVM, it includes 3 parts by statistics in this review as shown in Fig. 21, namely the kernel parameter optimization, kernel function combination, and long prediction performance improvement.

The kernel parameters of an RVM are normally selected based on empirical knowledge. The kernel function of optimized RVM [135] is the Gaussian kernel function with 5 kernel parameters. In order to improve prediction accuracy, Cai et al. introduced the artificial fish swarm algorithm (AFSA) to find the optimal kernel parameter [136]. Zhou et al. used the PSO method to optimize the kernel function bandwidth σ of RVM, and the average error of their model was less than 2.18% [137]. An optimized PSO method is utilized to search for the optimal parameter of the RBF kernel function in Ref. [138].

Multiple kernel learning, which combines the advantages of multiple different kernel functions, is another way to improve the kernel function of RVM. The key to this method is the weights determination of the selected kernel function. Zhang et al. used the PSO algorithm to achieve multiple kernel RVM [139], while Sun et al. used the grid search method to optimize the parameters of their multiple kernel RVM [36]. Some researchers tried to achieve the kernel parameter optimization and kernel function combination simultaneously. The ensemble weight coefficient of the different kernel functions and parameters was optimized by using a discrete PSO algorithm in [140], which realized the choice of some effective kernel and optimization of kernel parameters of RVM. The

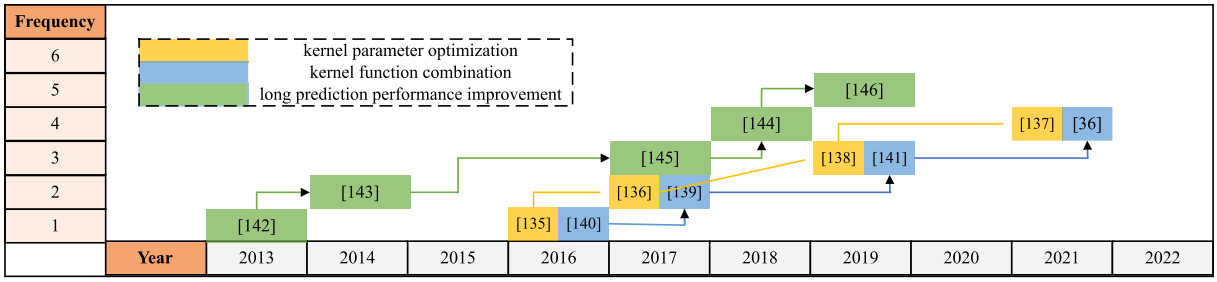

Fig. 21. The development of the research core of RVM.

prediction accuracy of this discrete PSO method was better than the single kernel RVM and RVM model ensembling all kernel functions. Chen et al. adopted the Cuckoo Search (CS) algorithm to find all kernel parameters and the ensemble weights of each kernel function. It was found that CS was more effective and accurate than the discrete PSO algorithm [141].

Poor performance of long-term prediction is one significant limitation of RVM. The main reason is that the relevance vectors and coefficients matrix of RVM cannot update at the same time in the single-step prediction. An incremental learning strategy is introduced to improve the precision of multi-step prediction of RVM, the key of which is that the input data of RVM contains the new online data sample in addition to the relevance vector [142,143]. A KF can be used to optimize the RVM result for better long-term prediction accuracy, by using each step result to update the training dataset and retrain the RVM model based on the new and online training dataset [144]. Zhao et al. used Deep Belief Network (DBN) to extract the features as the inputs of RVM to improve the stability of longer prediction [145]. The RVM and gray model (GM) are alternately adopted for long-term prediction considering the capacity regeneration in [146]. More details about GM and the algorithm structure for RUL prediction can be found in [146].

In the future, one ideal way is using RVM is used to generate RVs which are then input into other algorithms with better performance in long-term prediction. Also, the hardware implementation of RVM is important such as the architecture proposed in [147].

2.3.2.2. Gaussian process regression. The GPR model is a nonparametric and probabilistic method. It can be regarded as a collection of a limited number of random variables following a joint Gaussian distribution described by its mean and covariance. The Gauss process can be expressed as follows:

$$f(x) \backsim GP(m(x), k_f(x, x')) \tag{7}$$

where $f(x)$ is the target output, $m(x)$ is the mean function and $k_f(x, x')$ is the covariance function.

The general use of GPR when modeling RUL is to extract relevant features as input, which is then fed to a GPR to obtain the RUL. The most common features are the peaks, regional area, slope, and other geometric features from incremental capacity analysis (ICA) curves as described in [148-151]. Other features include voltage-dependent features from partial voltages [152] and thermal-dependent features [153], error series through CEEMD [154], and EIS spectrum [155].

Researchers working with GPR are mainly focused on kernel function enhancement, model combination, and pseudo-transfer learning. Their development over the last few years is shown in Fig. 22.

It can be quite effective to enhance kernel functions by considering electrochemical, or empirical elements, as was demonstrated by Liu et al. [156], who coupled the Arrhenius law and a polynomial equation with the covariance function in the kernel function. The improved kernel function can reflect the temperature, DOD, and capacity, and achieve an MAE and RMSE less than 0.07 Ah (0.3%), and 0.09 Ah (0.4%). They also modified the basic squared exponential (SE) function serving as a covariance function with the automatic relevance determination (ARD) to remove the irrelevant inputs of the GPR. Richardson et al. used an exponential model which can reflect battery degradation as a mean function rather than 0, and improved the predictive performance of basic GPR [157]. Another method to enhance kernel functions is to combine different kernel functions. In the research of Liu et al. [158], the square exponential covariance function and periodic covariance function were added for better ability, because the former can describe the capacity degradation and the latter can describe the regeneration phenomenon. Peng et al. also proposed a similar hybrid GPR to improve the accuracy of RUL prediction [159]. In [157], 10 different compound kernel functions are evaluated to choose a suitable kernel function with the best performance. Neural Gaussian Process (NGP) is proposed to solve the difficulty of selecting the kernel function of the Gaussian process through experience based on the advantages of the neural network function approximation [160].

The model combination means that the final prediction result is made up of multiple GPR results adopting different kernel functions based on different inputs. In [161], the input of the GPR was the residual sequences of a VMD and the sub-GPR models utilized the squared exponential covariance functions to describe the monotonicity and periodic covariance functions to describe recurrent sequences respectively. In Qiao et al.'s research, the GPR with the input of time index was used to capture the global trend and that with the lag vector was used to reflect the local regeneration and fluctuations [162]. Li et al. used multiple exponential and linear models as the trend function of a GP model to reflect the capacity degradation of Li-ion batteries under different degradation stages [163]. The combination or weighting strategies of multiple GPR models include the weighting strategy based on prediction uncertainty [164], and induced ordered weighted averaging (IOWA) operators [165].

In this review, the concept of pseudo-transfer learning is refined first, the key point of which is to research the correlation or covariance of the capacity between different batteries and to use the prior knowledge of similar batteries. Through sharing similar relationships between different batteries or different cycles, the model generalities and prediction speed can be improved. When the batteries have strong similarities, their prediction result will also be similar. In [166], the co-kriging surrogate is adopted to employ the low- and high-fidelity data to forecast the degradation of the primary cell, where the high-fidelity data is corresponding to the capacity degradation profile of the primary battery under cycling load and the low-fidelity data is corresponding to the capacity degradation profile of a related cell. In addition, multiple-output GPs models can be regarded as pseudo-transfer learning, because they can effectively exploit correlations among data from different

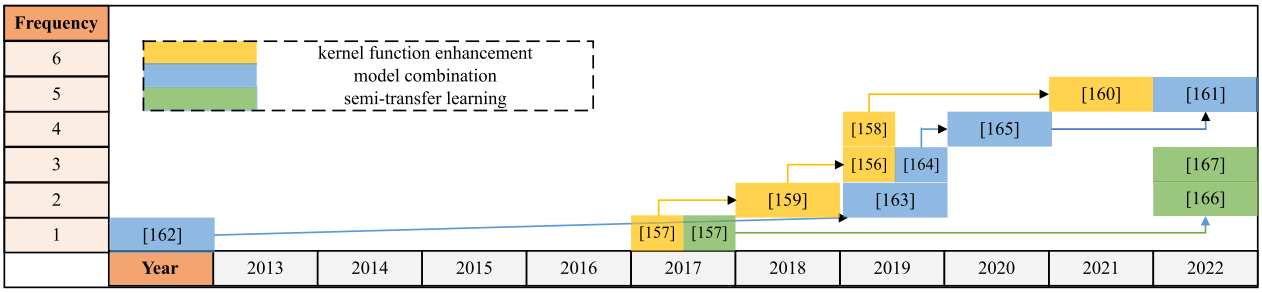

Fig. 22. The development of the research core of GPR.

cells and cycles by modifying the covariance in kernel function to reflect the similarity between multiple batteries. Herein, the covariance is the product of a label covariance and a standard covariance.[157]. Chehade et al. proposed a multi-output convolved gaussian process (MCGP) model, which tried to use the convolution process to learn the correlations of capacity between different batteries under similar conditions [167].

In the future, kernel functions that can reflect physical information and the covariance between different sources and different cycles should be further explored.

3. Comparison of RUL prediction methods

The ML algorithms are evaluated from the perspective of accuracy, and algorithm characteristics including online updating performance (time-varying characteristic), uncertainty quantification, generalization, and ease of implementation. The accuracy and algorithm characteristic results are summarized in Table 1. It is found that all ML algorithms show high accuracy.

The algorithm characteristics of different ML methods are further presented in the form of a radar map in Fig. 23. It can be concluded that RVM and GPR can be used to quantify the uncertainty with higher accuracy but relatively poor generalization performance. RNN and CNN present good performance and information extraction ability with high accuracy. SVR and ELM exhibit a good online updating ability and fast prediction. And AR is found to be the simplest algorithm with acceptable accuracy which is also easy to implement.

However, almost every algorithm has limitations for RUL prediction of batteries. The SVM method is not suitable for long-term dependency prediction, and always has weak accuracy [72]. It shows the overfitting [168] or non-convergence trend sometimes [169], especially with insufficient full-life data. SVM also cannot capture the capacity regeneration phenomenon of batteries [170],

Table 1
Performance overview of different ML methods for RUL prediction.

Year	ML method	Ref.	Feature	Battery	Accuracy	Characteristics
2021	RVM	[126]	rate of discharge surface temperature	lithium cobalt oxide batteries (LCO)	4.37e-06 RMSE	• high accuracy good robust performance
2021	BLS-RVM	[32]	capacity by EMD	LCO/graphite (NASA 5#)	0.0105 RMSE	• high prediction stability long-term prediction ability uncertainty expression
2017	RVM	[33]	capacity	LCO/graphite (CALCLE 0.9Ah)	5cycles	• higher accuracy than GPR narrow confidence intervals
2021	GPR	[153]	slope and intercept of dv and T	LFP/graphite (MIT124#battery)	93.27 (5.17%) cycles	• more accurate and convenient than existing feature extraction methods
2019	Multidimensional GPR	[158]	charge time interval of voltage varying from 3.9 V to 4.2 V, the charge voltage varying from 3.9 V to the voltage after 500 seconds, the CV charge current drop between 1.5 A (the CC charge current) and the current after 1000 s.	LCO/graphite (NASA 5#)	1 cycle	• high accuracy reliability output being probabilistic better in tracking the variation curve of actual value than the basic GPR model.
2021	GPR	[151]	optimized peak height, position, and area	CALCE CS2-35#	5 cycles	• high accuracy and robustness in a different type of battery
2021	RNN	[43]	capacity	LCO/graphite (NASA 5#)	0.0030 RMSE	• good generalization performance
2021	SADE-MESN	[45]	capacity	LCO/graphite (NASA 5#)	3 cycles	• high prediction accuracy stable output
2021	B-LSTM	[54]	RES of SOH	CALCE CS2-34#	0.0017 RMSE (1 cycle)	• small accumulated error high accuracy in different batteries and different test conditions
2021	GRU	[72]	capacity	LCO/graphite (NASA 5#)	0.0156 RMSE	• high accuracy and efficiency early stop technology
2022	SVR-PSO	[80]	the integral of the voltage over time (IVT)	LCO/graphite (NASA 5#)	0.0133 RMSE	• online method high accuracy
2013	AR	[102]	capacity	CALCE	26 cycles	• high accuracy, easy to implement
2022	CNN	[111]	human-picked feature; data-driven features	LFP/graphite (MIT)	6.46% MAPE	• high accuracy based on one cycle only
2020	hybrid parallel residual CNNTCN	[37]	capacity	LFP/graphite (MIT) LCO/graphite (NASA 5#)	0.009RMSE	• high accuracy early-stage prediction local fluctuation capture
2021	OS-PELM	[123]	singular values of the battery-measured parameters	LCO-NCO/graphite (Oxford) NMC /graphite (NASA) LFP/graphite (MIT)	0.2972 RMSE 0.0038 RMSE 0.0027 RMSE	• online sequential learning updation of network parameters
2022	CTC-ELM	[124]	capacity	LCO/graphite (NASA) LCO/graphite (Oxford)	0.000036 MSE 0.000001 MSE	• good time-varying characteristics high prediction accuracy
2020	BL-ELM	[125]	capacity	NASA CALCE	0.0145RMSE 0.0161RMSE	• more nodes of the input layer accurate, fast, and robust

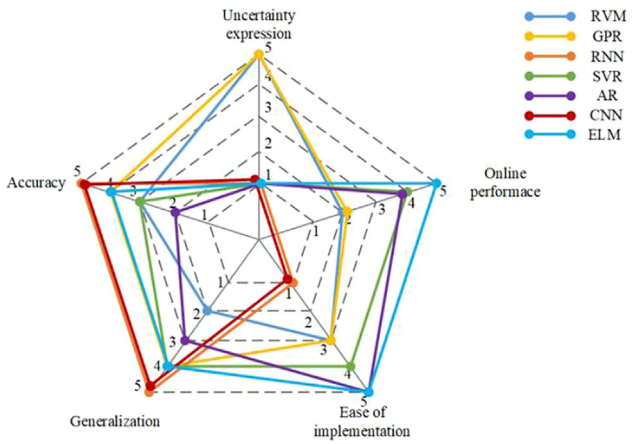

Fig. 23. Comparison of the different ML methods from different perspectives (where a higher number indicates a better performance).

and the prediction error fluctuation is large when used in different batteries [73]. The GPR method also cannot capture the capacity regeneration phenomenon of batteries [170] and its prediction curves sometimes show little similarity with real data. For example, the prediction curve is almost a straight line with higher RMSE in [126]. The prediction range of applicability of RVM is narrow [33], and it may lose accuracy in RUL prediction with a short length of training data [71]. In fact, the setting of the kernel function of SVM, RVM, and GPR has a significant influence on the RUL prediction result. If they are not set properly, the prediction result will be far from the real data. Researchers always like to set the mean of the kernel function as zero, which needs to be improved in the future. The AR lacks long-term memory capacity or SOH [171].

For the deep learning method, ELM has larger predictive errors [78]. Using CNN alone makes the prediction result far away from the real measurement data in Ref. [75]. The execution speed of LSTM is not fast because it needs more training iterations [171 72]. It predicts the future degradation trend based on historical data, so it omits short-term fluctuations. However, the short-term fluctuation will affect the network parameters of LSTM and thus reduce the accuracy of RUL prediction [172]. Similar limits occur in GRU which just generates a steady decline prediction for the overall trends [171]. The prediction error of GRU is higher than that of LSTM because GRU has only two gates. For deep learning methods, a large number of data points are necessary. If the training data points are limited, the predicted curve for capacity will be almost a horizontal straight line and the RUL prediction result will not be present.

4. Challenges and outlook

4.1. Improvements to the method

Plenty of research about RUL prediction based on the ML algorithm has been done in the last 10 years, mainly including ML algorithm parameter optimization, ML structure optimization, and algorithm fusion. Through the review of the typical algorithm, it is believed the following five aspects should be improved.

Early prediction: Early prediction means using fewer initial cycle data to predict RUL. Accurate early RUL prediction can detect failures earlier and reduce the consumption of batteries. However, it is still a challenging task because the effective degradation information available is limited in the early operation of the battery, and the electrochemical characteristics of the battery are greatly influenced by the following operating conditions. The related research about early prediction mainly exists in the LSTM [51] and CNN [111] algorithms as mentioned above. The prerequisite for early prediction is the effective extraction of the feature. One possible approach for realizing early RUL prediction is using the ML method to extract features directly rather than picking features manually. Another approach is inputting more features each time like adopting the broad learning method mentioned in the ELM part. Early broad learning can be extended to other ML methods in the future.

Local capacity regeneration modeling: The local capacity regeneration phenomenon exists in almost all battery aging tests. The capacity sometimes suddenly increases after a complete charging-discharging cycle or a long idling time. Guo et al. tried to explore this phenomenon from the full cell, electrodes, and materials levels. The results showed that large DODs and high temperatures can accelerate capacity regeneration [173]. If the ML algorithm can learn this phenomenon, the accuracy of RUL prediction especially for that based on the SOH estimation will be improved. As summarized above, signal decomposition methods are usually adopted to obtain a high-frequency signal to reflect capacity regeneration. ML algorithms then used the decomposed signal to predict future regeneration, which is the most common modeling method for local capacity regeneration. The essence of other methods including using sliding windows, dilated convolution, and many-to-one RNN structures is to obtain more information about local regeneration. The ideal situation is to keep the signal undistorted. In the future, identifying the regeneration interval from the whole degradation process followed by using a suitable ML method in this interval can be considered. In that case, different ML algorithms are responsible for different battery intervals.

Physical information fusion: Integrating physics information with ML algorithm has many advantages like higher accuracy, stronger long-term prediction performance, better generalization ability, etc. Now physics informed NN has been introduced [174,175], which still deserves further study. In fact, all ML algorithms have the possibility of physical information fusion. For example, modification of kernel function format can be taken into consideration when applying ML algorithm with kernel functions according to the Arrhenius law, polynomial equation with the covariance function, and periodic covariance function to reflect the temperature, DOD, and local regeneration respectively. More fusion formats need to be explored in the future.

Generalized transfer learning: It is proposed in this review that generalized transfer learning contains transfer learning and incremental learning. This approach is to maintain most of the RUL prediction model parameters when the distribution of data is similar rather than to retrain the model, where the similarity of the data can be measured by the correlation coefficient. Transfer learning can fine-tune model parameters in the target domain based on the source domain results. Incremental learning uses the new online data sample and previous data as the input data without rebuilding the model. This idea can improve the learning efficiency of ML algorithms and is realistic for online applications in the future.

Hardware implementation: Although great progress has been made in algorithmic research, the research about hardware implementation of the RUL ML algorithm is still not mature. We only found the relevant software and hardware architecture in [147]. The hardware conditions of EVs or other practical applications will be different from the experimental lab. The industry acceptance of computationally demanding RUL prediction algorithms is dependent on the cost and availability of hardware platforms. So, the ML algorithm implementation deserves further study from a hardware perspective.

4.2. RUL prediction for lifetime extension

Among all methods to extend the battery lifetime, charging profile optimization is the most realistic from the end-user perspective. Its essence is to minimize the irreversible capacity loss caused by side reactions for the entire charging process without reducing the amount of charging capacity. Huang et al. researched the influence of low-frequency positive pulsed current (PPC) charging on the lifetime extension of NMC-based batteries experimentally. Compared with the constant current (CC), the PPC charging at 0.05 Hz can extend the battery lifetime by 81.6% by reducing the LLI, LAM, and kinetic hindrance at different stages [10]. Lee presented a four-stage constant current (4SCC) charging strategy based on an aging-level, where the charge current was decided by the SOH, and the lifetime was increased by around 40 cycles [11]. Li et al. studied the multi-step fast charging protocol and the aging mechanism behind it. Compared to the standard constant current-constant voltage (CC-CV) profile, one of their proposed multi-step fast charging protocols can increase 200 full equivalent cycles and the charging time is reduced by 20% [176]. Maia optimized the conventional CC-CV charging profile for Li-ion batteries based on the electrochemical model that can calculate the irreversible capacity loss. The optimized charging profile is shown by the blue line in Fig. 24, which displayed that the lifetime can be increased by around 27% [12]. Most of the methods mentioned considered SOH in the model.

RUL can be regarded as an index to evaluate the lifetime extension. Peter et al. used the RUL prediction result and Bayesian optimization method to find the optimal fast charging profile with long cycle life from 224 charging profiles with low

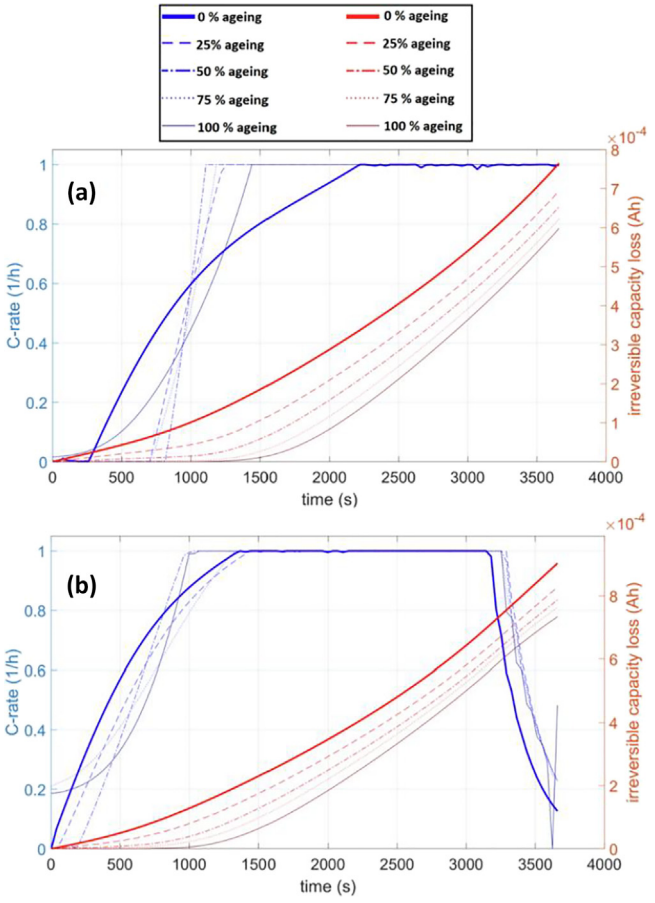

Fig. 24. The proposed charging profile in Ref. [12].

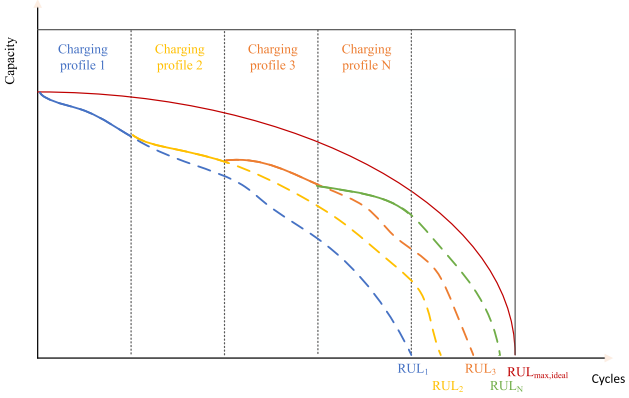

Fig. 25. Strategy of extending battery lifetime based on RUL prediction through optimizing charging profile.

experiment cost [177]. A possible method for extending the lifetime of the battery is optimizing the charging profile serval times according to the accurate RUL prediction results online in the future. The strategy is shown in Fig. 25. For EVs, different users will generate different discharging patterns and different RULs. For example, the new optimized charging strategy should be proposed when the RUL is reduced by 5% every time. The electrochemistry model reflecting the charging current profile and irreversible loss like SEI growth will be built. The optimized charging profile will be obtained using optimization algorithms based on the electrochemical model. Then the optimized charging profile is operated with a few cycles and RUL prediction results are used to judge whether the battery life can be extended. If the predicted RUL is increased, the optimal charging profile can be executed until the next optimization. The RUL upper bound needs to be determined.

5. Conclusions

Utilizing ML algorithms to predict the RUL of lithium-ion has made great progress in the last decade. This review systematically summarizes the research core and its development trend of common ML methods for battery RUL prediction, briefly overview the methods to extend the lifetime of a battery, and analyzes the possibility of extending the lifetime of a battery based on RUL prediction.

The ten most used RUL ML algorithms from 2012 to 2022 and their occurrence frequency in publications over years are analyzed. The RNN is the most frequently used algorithm with 61 occurrences in papers from the indexed database. The characteristics of the four most used signal processing methods and their application with ML methods are introduced in detail. The research cores changing with the publication year of the top-used ML algorithms are provided in unified figures to help the readers to understand the ML method development fast and clearly. Generally, the research of all algorithms is focused on parameter optimization, algorithm fusion, structure design, and transfer learning. Based on the analysis of the mentioned algorithms, the specific improvement direction and the possibility of RUL prediction are outlined: early prediction, local regeneration modeling, physical information fusion, generalized transfer learning, and hardware implementation. End-users can extend battery lifetime by optimizing the charging profile based on the online RUL prediction results. Through this paper, the authors hope to help researchers to have a clear understanding of the research core development of different ML algorithms in RUL prediction and to propose more useful algorithms in the future. We also hope the lifetime extension part of

this review article can give some inspiration in using RUL results to extend battery lifetime.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been funded by China Scholarship Council. The fund numbers are 202108320111 and 202208320055.

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jechem.2023.03.026.

References

[1] X. Sui, S. He, S.B. Vilsen, J. Meng, R. Teodorescu, D.-I. Stroe, Appl. Energy 300 (2021).
[2] Y. Che, X. Hu, X. Lin, J. Guo, R. Teodorescu, Energy Environ. Sci. 16 (2023) 338–371.
[3] Y. Li, K. Liu, A.M. Foley, A. Zülke, M. Berecibar, E. Nanini-Maury, J. Van Mierlo, H.E. Hoster, Renew. Sustain. Energy Rev. 113 (2019).
[4] H. Meng, Y.-F. Li, Renew. Sustain. Energy Rev. 116 (2019).
[5] Q. Zhang, L. Yang, W. Guo, J. Qiang, C. Peng, Q. Li, Z. Deng, Energy 241 (2022).
[6] K. Luo, X. Chen, H. Zheng, Z. Shi, J. Energy Chem. 74 (2022) 159–173.
[7] M.S.H. Lipu, M.A. Hannan, A. Hussain, M.M. Hoque, P.J. Ker, M.H.M. Saad, A. Ayob, J. Clean. Prod. 205 (2018) 115–133.
[8] S. Jin, X. Sui, X. Huang, S. Wang, R. Teodorescu, D.-I. Stroe, Electronics 10 (2021) 3126.
[9] H. Rauf, M. Khalid, N. Arshad, Renew. Sustain. Energy Rev. 156 (2022).
[10] X. Huang, W. Liu, J. Meng, Y. Li, S. Jin, R. Teodorescu, D.-I. Stroe, IEEE J. Emerg. Sel. Top. Power Electron. 11 (2023) 57–66.
[11] C.-H. Lee, Z.-Y. Wu, S.-H. Hsu, J.-A. Jiang, IEEE Trans. Energy Convers. 35 (2020) 1475–1484.
[12] L.K.K. Maia, L. Drünert, F. La Mantia, E. Zondervan, J. Clean. Prod. 225 (2019) 928–938.
[13] A. Motaqi, M.R. Mosavi, Eng. Sci. Technol. Int. J. 23 (2020) 544–554.
[14] D. Karimi, S. Khaleghi, H. Behi, H. Beheshti, M. Hosen, M. Akbarzadeh, J. Van Mierlo, M. Berecibar, Energies 14 (2021) 2907.
[15] F. Widmer, A. Ritter, P. Duhr, C.H. Onder, ETransportation 14 (2022).
[16] Y. Wu, Z. Huang, H. Liao, B. Chen, X. Zhang, Y. Zhou, Y. Liu, H. Li, J. Peng, Appl. Energy 257 (2020).
[17] K. Vatanparvar, S. Faezi, I. Burago, M. Levorato, M.A. Al Faruque, IEEE Trans. Smart Grid 10 (2019) 2959–2968.
[18] Y. Wu, Z. Huang, H. Hofmann, Y. Liu, J. Huang, X. Hu, J. Peng, Z. Song, Energy 251 (2022).
[19] K. Uddin, T. Jackson, W.D. Widanage, G. Chouchelamane, P.A. Jennings, J. Marco, Energy 133 (2017) 710–722.
[20] H. Min, Z. Zhang, W. Sun, Z. Min, Y. Yu, B. Wang, Appl. Therm. Eng. 181 (2020).
[21] J.J. Eckert, T.P. Barbosa, S.F. da Silva, F.L. Silva, L.C.A. Silva, F.G. Dedini, Energy Convers. Manag. 252 (2022).
[22] Y. Li, D.-I. Stroe, Y. Cheng, H. Sheng, X. Sui, R. Teodorescu, J. Energy Storage 33 (2021).
[23] B. Jiang, J. Zhu, X. Wang, X. Wei, W. Shang, H. Dai, Appl. Energy 322 (2022).
[24] J. Obregon, Y.R. Han, C.W. Ho, D. Mouraliraman, C.W. Lee, J.Y. Jung, J. Energy Storage 60 (2023).
[25] P. Gasper, A. Schiek, K. Smith, Y. Shimonishi, S. Yoshida, Cell Reports Phys. Sci. 3 (2022).
[26] M. Faraji-Niri, M. Rashid, J. Sansom, M. Sheikh, D. Widanage, J. Marco, J. Energy Storage 58 (2023).
[27] K. Mc Carthy, H. Gullapalli, K.M. Ryan, T. Kennedy, J. Energy Storage. 50 (2022) 104608.
[28] K.A. Severson, P.M. Attia, N. Jin, N. Perkins, B. Jiang, Z. Yang, M.H. Chen, M. Aykol, P.K. Herring, D. Fraggedakis, M.Z. Bazant, S.J. Harris, W.C. Chueh, R.D. Braatz, Nat. Energy 4 (2019) 383–391.
[29] Y. Zhou, M. Huang, Microelectron. Reliab. 65 (2016) 265–273.
[30] K. Liu, Y. Shang, Q. Ouyang, W.D. Widanage, IEEE Trans. Ind. Electron. 68 (2021) 3170–3180.
[31] M. Huotari, S. Arora, A. Malhi, K. Främling, Appl. Soft Comput. 111 (2021).
[32] Z. Chen, N. Shi, Y. Ji, M. Niu, Y. Wang, Energy 234 (2021).
[33] Y. Chang, H. Fang, Y. Zhang, Appl. Energy 206 (2017) 1564–1578.
[34] H. Yang, P. Wang, Y. An, C. Shi, X. Sun, K. Wang, X. Zhang, T. Wei, Y. Ma, ETransportation 5 (2020).
[35] Z. Yun, W. Qin, W. Shi, P. Ping, Energies 13 (2020) 4858.
[36] X. Sun, K. Zhong, M. Han, Measurement 170 (2021).
[37] D. Zhou, Z. Li, J. Zhu, H. Zhang, L. Hou, IEEE Access 8 (2020) 53307–53320.
[38] J. Fan, J. Fan, F. Liu, J. Qu, R. Li, IEEE Access 7 (2019) 160043–160061.
[39] Y. Wang, R. Pan, D. Yang, X. Tang, Z. Chen, Energy Procedia 105 (2017) 2053–2058.
[40] X. Han, Z. Wang, Z. Wei, Appl. Energy 302 (2021).
[41] H. Wei, J. Chen, H. Wang, J. An, L. Chen, Diangong Jishu Xuebao/Trans. China Electrotech. Soc. 35 (2020) 1181–1188.
[42] S.-J. Kwon, D. Han, J.H. Choi, J.-H. Lim, S.-E. Lee, J. Kim, J. Electroanal. Chem. 858 (2020).
[43] S. Ansari, A. Ayob, M.S. Hossain Lipu, A. Hussain, M.H.M. Saad, Sustainability. 13 (2021) 13333.
[44] J.C. Chen, T.-L. Chen, W.-J. Liu, C.C. Cheng, M.-G. Li, Adv. Eng. Informatics 50 (2021).
[45] Y. Ji, Z. Chen, Y. Shen, K. Yang, Y. Wang, J. Cui, Appl. Soft Comput. 104 (2021).
[46] M. Catelani, L. Ciani, R. Fantacci, G. Patrizi, B. Picano, IEEE Trans. Instrum. Meas. 70 (2021) 1–11.
[47] D. Liu, W. Xie, H. Liao, Y. Peng, IEEE Trans. Instrum. Meas. 64 (2015) 660–670.
[48] D. Liu, H. Wang, Y. Peng, W. Xie, H. Liao, Energies 6 (2013) 3654–3668.
[49] M. Lukoševičius, in: Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2012, pp. 659–686
[50] Y. Zhang, R. Xiong, H. He, Z. Liu, in: 2017 Progn. Syst. Heal. Manag. Conf., IEEE, 2017, pp. 1–4
[51] Y. Zhang, R. Xiong, H. He, M.G. Pecht, IEEE Trans. Veh. Technol. 67 (2018) 5695–5705.
[52] Z. Tong, J. Miao, S. Tong, Y. Lu, J. Clean. Prod. 317 (2021).
[53] C. Wang, N. Lu, S. Wang, Y. Cheng, B. Jiang, Appl. Sci. 8 (2018) 2078.
[54] G. Cheng, X. Wang, Y. He, Energy 232 (2021).
[55] K. Park, Y. Choi, W.J. Choi, H.-Y. Ryu, H. Kim, IEEE Access 8 (2020) 20786–20798.
[56] M. Zhang, L. Wu, Z. Peng, in: 2021 IEEE 16th Conf. Ind. Electron. Appl., IEEE, 2021, pp. 1364–1371
[57] Z. Wang, N. Liu, Y. Guo, Neurocomputin. 466 (2021) 178–189.
[58] Y. Huang, Y. Tang, J. VanZwieten, IEEE Trans. Ind. Electron. 69 (2022) 856–867.
[59] P. Li, Z. Zhang, Q. Xiong, B. Ding, J. Hou, D. Luo, Y. Rong, S. Li, J. Power Sources 459 (2020).
[60] M. Lin, D. Wu, G. Zheng, J. Wu, Trans. Inst. Meas. Control. (2021) 014233122110409.
[61] X. Li, L. Zhang, Z. Wang, P. Dong, J. Energy Storage 21 (2019) 510–518.
[62] X. Cui, T. Hu, IEEE Access 8 (2020) 207298–207307.
[63] Y. Zhang, L. Chen, Y. Li, X. Zheng, J. Chen, J. Jin, J. Energy Storage 44 (2021).
[64] X. Hu, X. Yang, F. Feng, K. Liu, X. Lin, J. Dyn. Syst. Meas. Control. 143 (2021) 1–13.
[65] B. Zraibi, C. Okar, H. Chaoui, M. Mansouri, IEEE Trans. Veh. Technol. 70 (2021) 4252–4261.
[66] L. Ren, J. Dong, X. Wang, Z. Meng, L. Zhao, M.J. Deen, IEEE Trans. Ind. Informatics 17 (2021) 3478–3487.
[67] Y. Liu, G. Zhao, X. Peng, IEEE Access 7 (2019) 155130–155142.
[68] C. Song, S. Lee, in: 2021 15th Int. Conf. Ubiquitous Inf. Manag. Commun., IEEE, 2021, pp. 1–5
[69] F.-K. Wang, C.-Y. Huang, T. Mamo, Appl. Sci. 10 (2020) 3549.
[70] D. Pan, H. Li, S. Wang, IEEE Trans. Instrum. Meas. 71 (2022) 1–10.
[71] Y. Song, L. Li, Y. Peng, D. Liu, in: 2018 12th Int. Conf. Reliab. Maint. Saf., IEEE, 2018, pp. 317–322
[72] R. Rouhi Ardeshiri, C. Ma, Int. J. Energy Res. 45 (2021) 16633–16648.
[73] M. Wei, H. Gu, M. Ye, Q. Wang, X. Xu, C. Wu, Energy Reports 7 (2021) 2862–2871.
[74] Z. Wang, Q. Ma, Y. Guo, Actuators 10 (2021) 234.
[75] T. Tang, H. Yuan, Reliab. Eng. Syst. Saf. 217 (2022).
[76] Y. Che, Z. Deng, X. Lin, L. Hu, X. Hu, IEEE Trans. Veh. Technol. 70 (2021) 1269–1277.
[77] J. Yu, J. Yang, Y. Wu, D. Tang, J. Dai, Int. J. Energy Res. 44 (2020) 11345–11360.
[78] Q. Zhao, X. Qin, H. Zhao, W. Feng, Microelectron. Reliab. 85 (2018) 99–108.
[79] F.-K. Wang, T. Mamo, J. Power Sources 401 (2018) 49–54.
[80] L. Zou, B. Wen, Y. Wei, Y. Zhang, J. Yang, H. Zhang, Energies 15 (2022) 2237.
[81] J. Xu, A. Zhen, Z. Cai, P. Wang, K. Gao, D. Jiang, IEEE Access 9 (2021) 85431–85441.
[82] Z. Wang, S. Zeng, J. Guo, T. Qin, Energy 167 (2019) 661–669.
[83] D. Gao, M. Huang, J. Power Electron. 17 (2017) 1288–1297.
[84] S. Li, H. Fang, B. Shi, Reliab. Eng. Syst. Saf. 210 (2021).
[85] L. Chen, Y. Zhang, Y. Zheng, X. Li, X. Zheng, Neurocomputing 414 (2020) 245–254.
[86] Z. Xue, Y. Zhang, C. Cheng, G. Ma, Neurocomputing 376 (2020) 95–102.
[87] X. Li, J. Miao, J. Ye, Adv. Mech. Eng. 7 (2015) 168781401562232.
[88] Z. Yang, Y. Wang, C. Kong, IEEE Trans. Instrum. Meas. 70 (2021) 1–11.
[89] Y. Wang, Y. Ni, N. Li, S. Lu, S. Zhang, Z. Feng, J. Wang, Energy Sci. Eng. 7 (2019) 2797–2813.
[90] Y. Wang, Y. Ni, S. Lu, J. Wang, X. Zhang, IEEE Trans. Veh. Technol. 68 (2019) 9543–9553.
[91] L.-L. Li, Z.-F. Liu, M.-L. Tseng, A.S.F. Chiu, Appl. Soft Comput. 74 (2019) 110–121.
[92] X. Li, X. Shu, J. Shen, R. Xiao, W. Yan, Z. Chen, Energies 10 (2017) 691.
[93] X. Li, Y. Ma, J. Zhu, Measurement 184 (2021).
[94] H. Dong, IEEE Access 9 (2021) 165490–165503.

[95] F. Wang, Z.E. Amogne, C. Tseng, J. Chou, Int. J. Energy Res. 46 (2022) 9080–9096.
[96] W. Liu, L. Yan, X. Zhang, D. Gao, B. Chen, Y. Yang, F. Jiang, Z. Huang, J. Peng, in: 2019 IEEE Energy Convers. Congr. Expo., IEEE, 2019, pp. 545–550
[97] S. Wang, L. Zhao, X. Su, P. Ma, in: 2014 Progn. Syst. Heal. Manag. Conf. (PHM-2014 Hunan), IEEE, 2014, pp. 317–322
[98] J. Zhou, D. Liu, Y. Peng, X. Peng, in: 2012 IEEE Int. Instrum. Meas. Technol. Conf. Proc., IEEE, 2012, pp. 2196–2199
[99] A. Nuhic, T. Terzimehic, T. Soczka-Guth, M. Buchholz, K. Dietmayer, J. Power Sources 239 (2013) 680–688.
[100] D. Liu, W. Xie, S. Lu, Y. Peng, in: 2015 Annu. Reliab. Maintainab. Symp., IEEE, 2015, pp. 1–6
[101] S.B. Vilsen, X. Sui, D.-I. Stroe, in: 2020 IEEE 9th Int. Power Electron. Motion Control Conf. (IPEMC2020-ECCE Asia), IEEE, 2020, pp. 1659–1666
[102] B. Long, W. Xian, L. Jiang, Z. Liu, Microelectron. Reliab. 53 (2013) 821–831.
[103] D. Liu, Y. Luo, J. Liu, Y. Peng, L. Guo, M. Pecht, Neural Comput. Appl. 25 (2014) 557–572.
[104] G. Limeng, P. Jingyue, L. Datong, P. Xiyuan, in: 2013 IEEE 11th Int. Conf. Electron. Meas. Instruments, IEEE, 2013, pp. 1014–1020
[105] D. Liu, Y. Luo, L. Guo, Y. Peng, in: 2013 IEEE Conf. Progn. Heal. Manag., IEEE, 2013, pp. 1–8
[106] Y. Song, D. Liu, C. Yang, Y. Peng, Microelectron. Reliab. 75 (2017) 142–153.
[107] Y. Song, C. Yang, T. Wang, D. Liu, Y. Peng, in: 2016 Progn. Syst. Heal. Manag. Conf., IEEE, 2016, pp. 1–6
[108] J. Lin, M. Wei, Int. J. Intell. Comput. Cybern. 14 (2021) 218–237.
[109] C.-P. Lin, J. Cabrera, F. Yang, M.-H. Ling, K.-L. Tsui, S.-J. Bae, Appl. Energy 275 (2020).
[110] B. Zhou, C. Cheng, G. Ma, Y. Zhang, I.O.P. Conf, Ser. Mater. Sci. Eng. 895 (2020).
[111] C.-W. Hsu, R. Xiong, N.-Y. Chen, J. Li, N.-T. Tsou, Appl. Energy 306 (2022).
[112] R. Xiong, J. Tian, W. Shen, J. Lu, F. Sun, J. Energy Chem. 76 (2023) 404–413.
[113] D. Kong, S. Wang, P. Ping, Int. J. Energy Res. 46 (2022) 6081–6098.
[114] P. Ding, X. Liu, H. Li, Z. Huang, K. Zhang, L. Shao, O. Abedinia, Renew. Sustain. Energy Rev. 148 (2021).
[115] X. Zhang, Y. Qin, C. Yuen, L. Jayasinghe, X. Liu, IEEE Trans. Ind. Informatics 17 (2021) 6820–6831.
[116] J. Hong, D. Lee, E.-R. Jeong, Y. Yi, Appl. Energy 278 (2020).
[117] S. Shen, M. Sadoughi, M. Li, Z. Wang, C. Hu, Appl. Energy 260 (2020).
[118] R. Razavi-Far, S. Chakrabarti, M. Saif, E. Zio, Expert Syst. Appl. 115 (2019) 709–723.
[119] X. Chen, Z. Liu, J. Wang, C. Yang, B. Long, X. Zhou, Electronics 10 (2021) 540.
[120] T. Sun, B. Xia, Y. Liu, Y. Lai, W. Zheng, H. Wang, W. Wang, M. Wang, Energies 12 (2019) 3678.
[121] J. Yang, Int. J. Electrochem. Sci. 13 (2018) 4991–5004.
[122] Y. Jiang, W. Zeng, L. Chen, Y. Xin, in: 2018 Int. Conf. Sensing,Diagnostics, Progn. Control, IEEE, 2018, pp. 415–420
[123] T. Tang, H. Yuan, J. Power Sources 514 (2021).
[124] M. Zhang, G. Kang, L. Wu, Y. Guan, Energy 238 (2022).
[125] Y. Ma, L. Wu, Y. Guan, Z. Peng, J. Power Sources 476 (2020).
[126] H. Feng, D. Song, J. Energy Storage 34 (2021).
[127] D. Wang, Q. Miao, M. Pecht, J. Power Sources 239 (2013) 253–264.
[128] Y. Zhang, R. Xiong, H. He, M. Pecht, J. Clean. Prod. 212 (2019) 240–249.
[129] S. Jia, B. Ma, W. Guo, Z.S. Li, J. Manuf. Syst. 61 (2021) 773–781.
[130] R. Wang, H. Feng, Qual. Reliab. Eng. Int. 37 (2021) 1232–1243.
[131] Y. Zhou, M. Huang, J. Electr. Eng. Technol. 13 (2018) 733–741.
[132] X. Qin, Q. Zhao, H. Zhao, W. Feng, X. Guan, in: 2017 IEEE Int. Conf. Progn. Heal. Manag., IEEE, 2017, pp. 1–6
[133] X. Zheng, H. Wu, Y. Chen, in: 2017 11th Asian Control Conf., IEEE, 2017, pp. 2698–2703
[134] X. Zheng, H. Fang, Reliab. Eng. Syst. Saf. 144 (2015) 74–82.
[135] Y. Zhou, M. Huang, Y. Chen, Y. Tao, J. Power Sources 321 (2016) 1–10.
[136] Y. Cai, L. Yang, Z. Deng, X. Zhao, H. Deng, in: 2017 2nd Int. Conf. Power Renew. Energy, IEEE, 2017, pp. 1–6
[137] Y. Zhou, H. Gu, T. Su, X. Han, L. Lu, Y. Zheng, J. Energy Storage 44 (2021).
[138] P. Guo, Z. Cheng, L. Yang, J. Power Sources 412 (2019) 442–450.
[139] C. Zhang, Y. He, L. Yuan, S. Xiang, IEEE Access 5 (2017) 12061–12070.
[140] W.-A. Yang, M. Xiao, W. Zhou, Y. Guo, W. Liao, Shock Vib. 2016 (2016) 1–15.
[141] Y. Chen, C. Zhang, N. Zhang, X. Guo, H. Wang, Y. Chen, in: 2019 Progn. Syst. Heal. Manag. Conf., IEEE, 2019, pp. 1–6
[142] J. Zhou, D. Liu, Y. Peng, X. Peng, in: 2013 IEEE Int. Instrum. Meas. Technol. Conf., IEEE, 2013, pp. 561–565
[143] D. Liu, J. Zhou, D. Pan, Y. Peng, X. Peng, Measurement 63 (2015) 143–151.
[144] Y. Song, D. Liu, Y. Hou, J. Yu, Y. Peng, Chinese J. Aeronaut. 31 (2018) 31–40.
[145] G. Zhao, G. Zhang, Y. Liu, B. Zhang, C. Hu, in: 2017 IEEE Int. Conf. Progn. Heal. Manag., IEEE, 2017, pp. 7–13
[146] L. Zhao, Y. Wang, J. Cheng, Appl. Sci. 9 (2019) 1890.
[147] S. Wang, D. Liu, J. Zhou, B. Zhang, Y. Peng, Energies 9 (2016) 572.
[148] G. Dong, Y. Xu, Z. Wei, IEEE Trans. Energy Convers. 37 (2022) 718–728.
[149] X. Pang, X. Liu, J. Jia, J. Wen, Y. Shi, J. Zeng, Z. Zhao, Microelectron. Reliab. 127 (2021).
[150] X. Li, C. Yuan, Z. Wang, J. Power Sources 467 (2020).
[151] W. Pan, X. Luo, M. Zhu, J. Ye, L. Gong, H. Qu, J. Energy Storage 42 (2021).
[152] J. Jia, J. Liang, Y. Shi, J. Wen, X. Pang, J. Zeng, Energies 13 (2020) 375.
[153] J. Kong, F. Yang, X. Zhang, E. Pan, Z. Peng, D. Wang, Energy 223 (2021).
[154] X. Cong, C. Zhang, J. Jiang, W. Zhang, Y. Jiang, IEEE Trans. Veh. Technol. 69 (2020) 12775–12785.
[155] Y. Zhang, Q. Tang, Y. Zhang, J. Wang, U. Stimming, A.A. Lee, Nat. Commun. 11 (2020) 1706.
[156] K. Liu, X. Hu, Z. Wei, Y. Li, Y. Jiang, IEEE Trans. Transp. Electrif. 5 (2019) 1225–1236.
[157] R.R. Richardson, M.A. Osborne, D.A. Howey, J. Power Sources 357 (2017) 209–219.
[158] J. Liu, Z. Chen, IEEE Access 7 (2019) 39474–39484.
[159] Y. Peng, Y. Hou, Y. Song, J. Pang, D. Liu, Energies 11 (2018) 1420.
[160] A. Yin, Z. Tan, J. Tan, Sensors 21 (2021) 1087.
[161] C. Zhang, S. Zhao, Y. He, IEEE Trans. Veh. Technol. 71 (2022) 2601–2613.
[162] Y.-J. He, J.-N. Shen, J.-F. Shen, Z.-F. Ma, AIChE J. 61 (2015) 1589–1600.
[163] M. Li, M. Sadoughi, S. Shen, C. Hu, in: 2019 IEEE Int. Conf. Progn. Heal. Manag., IEEE, 2019, pp. 1–6
[164] X. Zheng, X. Deng, IEEE Access 7 (2019) 150383–150394.
[165] M. Bouzenita, L. Mouss, F. Melgani, T. Bentrcia, Qual. Reliab. Eng. Int. 36 (2020) 2146–2169.
[166] H. Valladares, T. Li, L. Zhu, H. El-Mounayri, A.M. Hashem, A.E. Abdel-Ghany, A. Tovar, J. Power Sources 528 (2022).
[167] A.A. Chehade, A.A. Hussein, IEEE Trans. Power Electron. 37 (2022) 896–909.
[168] S.S. Mansouri, P. Karvelis, G. Georgoulas, G. Nikolakopoulos, IFAC-PapersOnLine 50 (2017) 4727–4732.
[169] R. Xiong, J. Lu, IFAC-PapersOnLine 51 (2018) 268–273.
[170] J. Lin, Z. Liu, Y. Yang, I.O.P. Conf, Ser. Earth Environ. Sci. 585 (2020).
[171] Y. Yang, J. Wen, Y. Shi, J. Jia, M. Li, J. Zeng, in: 2021 China Autom. Congr., IEEE, 2021, pp. 270–275
[172] H. Pan, C. Chen, M. Gu, Energies 15 (2022) 2498.
[173] J. Guo, Y. Li, J. Meng, K. Pedersen, L. Gurevich, D.-I. Stroe, J. Energy Chem. 74 (2022) 34–44.
[174] R.G. Nascimento, M. Corbetta, C.S. Kulkarni, F.A.C. Viana, J. Power Sources 513 (2021).
[175] W. Guo, Z. Sun, S.B. Vilsen, J. Meng, D.I. Stroe, J. Energy Storage 56 (2022).
[176] Y. Li, J. Guo, K. Pedersen, L. Gurevich, D.-I. Stroe, J. Energy Chem. 80 (2023) 237–246.
[177] P.M. Attia, A. Grover, N. Jin, K.A. Severson, T.M. Markov, Y.H. Liao, M.H. Chen, B. Cheong, N. Perkins, Z. Yang, P.K. Herring, M. Aykol, S.J. Harris, R.D. Braatz, S. Ermon, W.C. Chueh, Nature 578 (2020) 397–402.