

Accepted Manuscript

A novel support vector regression method for online reliability prediction under multi-state varying operating conditions

Tao Tao , Enrico Zio , Wei Zhao

PII: S0951-8320(17)30475-1
DOI: [10.1016/j.ress.2018.04.027](https://doi.org/10.1016/j.ress.2018.04.027)
Reference: RESS 6146



To appear in: *Reliability Engineering and System Safety*

Received date: 22 April 2017
Revised date: 20 April 2018
Accepted date: 27 April 2018

Please cite this article as: Tao Tao , Enrico Zio , Wei Zhao , A novel support vector regression method for online reliability prediction under multi-state varying operating conditions, *Reliability Engineering and System Safety* (2018), doi: [10.1016/j.ress.2018.04.027](https://doi.org/10.1016/j.ress.2018.04.027)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Highlights

- A data-driven method is proposed for prognostics in multi-state operating condition.
- Features are developed to assess different reliability evolution trends.
- A sequential Monte Carlo framework is adopted for the online prediction.
- The proposed method is applied with several electrical and mechanical applications.
- The prediction accuracy, robustness and speed are evaluated.

ACCEPTED MANUSCRIPT

A novel support vector regression method for online reliability prediction under multi-state varying operating conditions

Tao Tao^a, Enrico Zio^{b,c}, Wei Zhao^{a*}

^a*School of Electronic and Information Engineering, Group 203, Beihang University, Beijing, 100191, China*

^b*Department of Energy, Polytechnic of Milan, Via Ponzio 34/3, 20133 Milan, Italy*

^c*Ecole Central Paris et Supelec, France*

**Corresponding Author Email: { zhaowei203@buaa.edu.cn }*

Abstract—Modeling the evolution of system reliability in the presence of Condition Monitoring (CM) signals is an important issue for improved reliability assessment and system lifetime prediction. In practice, during its lifetime, a system usually works under varying operating conditions due to internal or external factors such as the ambient environments, operational profiles or workloads. In this context, the system reliability can show varying evolution behaviors (follow changing underlying trajectories), which presents new challenges to describe precisely the dynamics of system reliability. Thus, this paper proposes a novel data-driven approach to address the problems including the identification of varying operating conditions, the construction and dynamical updating of evolution model, and finally the online prediction of system reliability, focusing on systems under one common and typical case of varying operating conditions, the multi-state operating condition. Experiments based on artificial data and some widely studied real reliability cases reveal that the proposed method has superior performance compared with some existing benchmark approaches, in the case under consideration. This improved reliability prediction provides fundamental basis for advanced prognostics such as the Remaining Useful Life (RUL) estimation.

Key words—reliability prediction, multi-state varying conditions, support vector regression, particle filter, possibilistic clustering classification

1. INTRODUCTION

Generally, reliability assessment focuses on predicting the future system reliability or State of Health (SOH) based on Condition Monitoring (CM) signals (observable indicators used to infer the unobservable underlying SOH, e.g., the capacity of a battery or the bearing vibration of a gear-box) [1]. It provides fundamental analysis for failure prognostics methods such as Remaining Useful Life (RUL) estimation or other methodologies aiming at avoiding system sudden shutdowns, increasing system availability and safety, and reducing the cost of accident and maintenance [2].

Traditional reliability assessment methods regard the degradation process of system reliability or SOH as determined and seek to construct the underlying degradation model from a large number of historical data of similar equipments, without taking account the dynamics of operating conditions or specificity for a individual equipment[3-8]. Actually as noted by Bian, by now the majority of reliability prediction models are based on the assumption that the prevailing operating conditions are regarded as temporarily constant or irrelevant to the evolution process [9]. However, the engineering equipments in practical industrial systems, especially in the modern complex systems, usually work under varying operating conditions caused by not only the uncontrollable external environment such as ambient temperature or other circumstance factors, but also the controllable operating profiles or workloads. In this situation, it is indispensable to consider the effect of varying conditions for advanced reliability assessment.

Recently, reliability assessment and SOH prediction for equipment under varying conditions have been investigated [10] and a dynamic multi-state condition is proposed as a typical model to depict general varying conditions [11]. Under the dynamic multi-state condition:

- I) The system is regarded as operating at one of several discrete candidate states.
- II) The concerned time series consisting of CM signals evolve following different underlying degradation models under different state.
- III) The transition between states is randomly happened and can be regarded as a hopping process but not a gradual change.

This modeling framework is mostly appropriate when the operating conditions can be clearly distinguished and their effects on the reliability evolution processes are significant. For instance, consider the workload induced from an aircraft engine in different flight conditions as: takeoff, maximum climb, maximum cruise, loiter, flight idle, taxi, ground idle, and cutoff [9]. The evolution of workload in these conditions will be totally different and the switching of conditions is sudden and can be regarded as random, thus distinct dynamic multi-state conditions can be recognized in this case.

Reliability assessment and SOH prediction with reference to equipments operating under this dynamic multi-state condition have been investigated from the perspective of statistical modelling and priori inference. The works of [9, 12-14] are some examples representing the state of art on this branch of methodologies.

The inherent drawbacks of the statistical modelling based methodologies derives from its two strong premises: 1) the degradation process of system state should follow a certain statistical model, such as the continuous-time Markov chain, the hidden Markov model, the hidden semi-Markov model or the Wiener process, .etc.; 2) the statistical property of the degradation model, for example the transition probability matrix for Markov-based models, should be apriori known or estimated. However, for practical instances, theoretical statistical models such as Markov chain are very hard to be verified and estimating its transition probability matrix is often time consuming or even inaccessible. Thus the applicability of such methods is limited in engineering practitions.

Another trajectory to address the reliability assessment and SOH prediction resorts to posterior estimation methodologies, e.g. machine learning, which asserts system state and system reliability or SOH through a “black box” constructed upon massive historical CM data and current measurement [15-17]. Nevertheless, this kind of methods have not yet been explored in depth for online SOH prediction under the dynamic multi-state condition, because of the difficulties lying in three aspects: 1) how to identify different system state by CM signals, in other words, how to effectively select the feature from CM signals; 2) how to efficiently classify the selected features into classes and 3) how to dynamically adapt the “black box”-like prediction model to meet the realtime demand for online tasks.

In this paper, we develop a novel Multi-state dynamic SVR approach to deal with the online reliability assessment and SOH prediction problems under the dynamic multi-state conditions. To the authors’ knowledge, this is the first time that such type of problems is solved with an online machine learning structure. To begin with, premises of this paper are listed in following:

- 1) The whole historical training reliability data are assumed known.
- 2) The measured CM signals get updated at each new time step.
- 3) Only the effects of different states are concerned. The effects posed by the state transition are regarded beyond the discussion of this paper.

As Fig. 1 shows, the proposed Multi-state dynamic SVR is a framework to posteriorly estimate the system state from recent measurements and recursively update the SOH prediction model according to the state estimation through a sequential Monte Carlo (SMC) paradigm.

As we shall show, novelties and contributes of the present work exhibit in the following aspects:

- 1) The effect of operating condition on system reliability evolution is analyzed. Especially, the dynamic multi-state operating condition is modeled and investigated.
- 2) An online machine learning framework is proposed to deal with the realtime reliability assessment and SOH prediction problems under the dynamic multi-state operating condition. It improves the existing statistical modelling based methods in two points: 1) the priori information of system states and state transition is not required, so it is more universal for practical applications

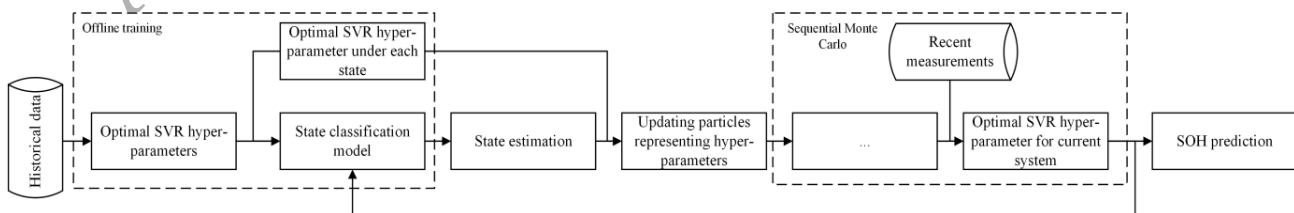


Fig. 1 Brief framework of the proposed Multi-state dynamic SVR modelling

and 2) it defines different operating states directly from the posterior degradation model of measurements but not from the preset operational profiles (though, of course, the alteration of degradation model of measurements is often caused by the change of operational profiles). This state classification and identification result shall be more efficient to improve the prediction model.

3) A novel feature of “optimal SVR hyper-parameters” with superior representing capacity and implementing efficiency is proposed to classify reliability evolution trajectories under different state. The feasibility of this feature derives from a notable fact that since the hyper-parameters of SVR are critically decisive to its prediction performance, the optimal SVR hyper-parameters filtered by SMC for training degradation trajectories under different state will distribute to different zone in the parameters space. In another word, the distribution of the particles (distributed candidate solutions in SMC paradigm) can be actually regarded as a statistical description or a feature about the system state. Compared with features obtained from traditional feature selection methods such as K-means or Principle Component Analysis (PCA), this “optimal SVR hyper-parameters” feature possesses enough representability of depicting the degradation models under different state, but cost fewer computation.

4) On the basis of “optimal SVR hyper-parameters” feature, an online prediction framework involving recursive SMC and a novel Replacement Operation is proposed to dynamically update the prediction model based on the state estimation. This implementing framework significantly decrease the computational burden on the online stage.

5) Improved prediction performance under the multi-state operating condition is achieved by the proposed method. On one hand, when the concerned system is recognized as working under a stable state, the PF-SVR will maintain convergence to the corresponding clustering center, which leads to a more accurate and stable prediction results in this case. On the other hand, when the concerned system is recognized as experiencing a transition between multi-state, the system state estimation can effectively help to capture the changing evolution trend and adapt to the new trend much quicker, so finally avoids losing tracking and results in better prediction performance.

6) It is noteworthy that all the online procedures including the system state estimation and prediction model updating are performed recursively, meaning that at each time step only the latest measurement are requested to be manipulated but not the total historical data. That is to say, the computational cost of the proposed method could be considerable.

To illustrate the aforementioned strengths, the proposed approach is applied to one artificial case concerning time series with a common multi-state condition consisting of four states (increasing, steady, vibration and decreasing) and real reliability cases including two cases regarding typical CM signals of Li-ion battery: I) the inner temperature; II) the capacity of full charged, and more cases based on standard databases from NASA PCoE (Prognostics Center of Excellence) [18]. Through these case studies, the performance of the proposed approach is evaluated with respect to the metric as Root Mean Square Error (RMSE) and is compared with the original PF-SVR method and another benchmark approach, the FGAPSO-SVR, from literature [19].

The remainder of the paper is organized as follows. Section 2 introduces some reviewing background knowledge about PF-SVR and the PCC method. The proposed novel model is presented in Section 3. Section 4 illustrates the case studies. Section 5 provides some conclusions on the findings of the research.

2. THEORETICAL BACKGROUND

Reliability prediction based on measured CM data can amount to a time series prediction problem that estimates the future values based on the known current and past data. Moreover, considering the multi-state operating condition, the objective problem this paper try to address can be mathematically stated as following.

Given that:

- 1) A collection of time series extracted from historical reliability trajectories under different states:

$$A = \{ts_1^1, \dots, ts_1^{L_1}, \dots, ts_j^1, \dots, ts_j^{L_j}, \dots, ts_c^1, \dots, ts_c^{L_c}\} \quad (1)$$

where ts_j^i represents a time series that j represents its true state, L_j represents the total number of time series under this state, and c counts the total number of states.

- 2) A discrete time series Y of variable y composed of past measurements until current time step:

$$Y = (y_k, y_{k-1}, \dots, y_1) \quad (2)$$

where k corresponds to the counter of current time step.

Pursue:

The value of time series in next time step:

$$\hat{y}_{k+1} = f_k(s_k) = f_k(y_k, y_{k-1}, \dots, y_{k-p+1}) \quad (3)$$

where the input vector s_k represents the p -lagged previous values of \mathbf{Y} and $f_k(\cdot)$ represent the prediction model at current time step.

Solution:

The problem described above can be divided into three sub-problems performed at, respectively, the off-line stage and online stage.

Off-line stage:

1) Build the classification model by denoting the collection of probable system states as $\Gamma = \{1, \dots, c\}$, where c corresponds to the total number of states:

$$\Delta: A \rightarrow \Gamma \quad (4)$$

which partitions A into some dissimilar classes, such that time series belonging to the same class are characterized as working under the same state.

Online stage:

2) Estimate the current system state according to the classification model Δ and the latest measurements s_k :

$$\mathbf{U}_k = \ell(s_k | \Delta) \quad (6)$$

where $\mathbf{U}_k = \{u_k^i, i \in \Gamma\}$ represents the memberships (probabilities) of current system belonging to each state.

3) Update the prediction model for prediction task of Eqs. (3), based on state estimation and priori trained optimal prediction models of each state:

$$f_k(\cdot) = h(f_{k-1}(\cdot) | \mathbf{U}_k) \quad (7)$$

In this paper, the sub-problems (1) are solved by the PCC method and the sub-problems (2) and (3) are addressed within our proposed approach based on an SVR framework. Due to the limits of paper length, only a brief overview about the basic theoretic background of SVR and PCC method are presented in the following and more detailed description about the proposed approach is given in Section 3.

2.1. Overviews about the SVR method and its parameters tuning through PF

For a concerned time series data set $D = \{(s_i, y_i)\}_{i=1}^n$, the regression function (3) can be framed as following:

$$f(s) = \mathbf{w}^T \Phi(s) + b \quad (8)$$

where \mathbf{w} represents the weight vector and b represents the intercept, respectively, of the regression model, and $\Phi: R^p \rightarrow F$ represents a nonlinear function which maps the low-dimensional input vectors to the feature space F . Thus, the pervasive form (8) is valid not only for the linear case but also the nonlinear case.

Then, the ε -insensitive loss function l is introduced to evaluate the error of regression:

$$l = |y - f(s)|_{\varepsilon} = \begin{cases} 0, & |y - f(s)| \leq \varepsilon \\ |y - f(s)| - \varepsilon, & \text{otherwise} \end{cases} \quad (9)$$

By minimizing both the regression error, which is denoted as slack variables ξ_i, ξ_i^* , and the Euclidean norm of the weight vector \mathbf{w} , i.e., $\|\mathbf{w}\|$, which depict the complexity of the regression model and therefore is closely related to the generalization ability, a

compromised quadric optimization problem to identify the regression model arises as follows:

$$\begin{aligned} & \text{minimize } J(\mathbf{w}, \xi_i, \xi_i^*) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \\ & \text{s.t. } \begin{cases} y_i - \mathbf{w}^T \Phi(\mathbf{s}) - b \leq \varepsilon + \xi_i \\ \mathbf{w}^T \Phi(\mathbf{s}) + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \quad i = 1, \dots, n \end{aligned} \quad (10)$$

where C is the penalty coefficient denotes the trade-off between empirical and generalization errors. Through the Lagrangian Dual method, the solution of this quadratic optimization problem can be obtained through the KKT condition and the output value is:

$$\begin{aligned} f(\mathbf{s}) &= \langle \mathbf{w} * \Phi(\mathbf{s}) \rangle + b = \sum_{i=1}^n (\alpha_i - \alpha_i^*) K(\mathbf{s}, \mathbf{s}_i) + b \\ K(\mathbf{s}_i, \mathbf{s}_j) &= \Phi(\mathbf{s}_i)^T \Phi(\mathbf{s}_j) \end{aligned} \quad (11)$$

where $K(\mathbf{s}_i, \mathbf{s}_j)$ is a kernel function satisfying the Mercer condition. More detailed derivation about the Dual Theorem and the KKT condition can refer to [20].

Generally, for better learning and prediction performance, the SVR parameters that need to be properly tuned include the kernel width σ which control the mapping function (refers Radial Basis Function without specific instructions) from the low dimensional non-linear data into the high dimensional feature spaces, the regularization coefficient C specifying the trade-off cost between minimizing the empirical risk and the model complexity and the tube size ε of the ε -insensitive loss function. Stacked into a vector as $\mathbf{x} = [\sigma, C, \varepsilon]$, the optimal hyper-parameters vector $\mathbf{x}_{optimal}$ is treated as the true state of a hypothetical system and the real reliability values are treated as the measurements of this system, thus the PF method can be adopted to estimate the true state, namely the optimal SVR hyper-parameters.

The state transition equation and measurement equation of this hypothetical system can be written as:

$$\begin{aligned} \mathbf{x}_{k+1} &= h_k(\mathbf{x}_k) + \mathbf{u}_k \\ y_k &= g_k(\mathbf{x}_k, \mathbf{s}_k) + \mathbf{v}_k \end{aligned} \quad (12)$$

where $\mathbf{x}_k \in R^m$ is the system state (optimal hyper-parameters) at time k , h_k is the state transition equation, $\mathbf{s}_k \in R^p$ is the input vector at time k , g_k is the measurement function and y_k is the measurement at time k . The vectors $\mathbf{u}_k, \mathbf{v}_k$ represent nonlinear noises with zero means and variances Q, R , respectively. In this PF-SVR model, the state transition equation h_k actually represents the particles drift operation, which manipulates the old particles to generate a set of new particles through the application of a given rule. The measurement equation g_k represents the expanded regression model built by SVR with the parameters of the optimal particle at time k . The input vector \mathbf{s}_k represents the training samples, in any case consisting of the historical reliability values.

On the basis of Eq. (12), the main procedures of SVR tuned by PF shows as follow:

- (1) Initialization. Set the initial SVR parameter vector \mathbf{x}_0 and the number of particles N_s (e.g. in the range [300, 500]). The particles values and their weights are initialized as $\{\mathbf{x}_0^i = \mathbf{x}_0, \omega_0^i = 1/N_s\}_{i=1}^{N_s}$.

For $k = 1, 2, \dots$:

- (2) Perform PF process:

a) By utilizing the particles set $\{\hat{\mathbf{x}}_{k-1}^i, 1/N_s\}_{i=1}^{N_s}$ at the previous time $k-1$, predict the current state (at time step k) through the state transition equation:

$$\hat{\mathbf{x}}_{k|k-1}^i = h(\hat{\mathbf{x}}_{k-1}^i) + \mathbf{u}_{k-1}^i, \omega_k^i = \omega_{k-1}^i \quad (13)$$

b) Predict the SVR regression outputs $\{\hat{y}_{k|k-1}^i\}_{i=1}^{N_s}$ through the measurement equation:

$$\hat{y}_{k|k-1}^i = g_{k-1}(\hat{\mathbf{x}}_{k|k-1}^i, \mathbf{s}_{k-1}) + \mathbf{v}_k \quad (14)$$

This is actually performed by training the SVR with the parameter vector $\hat{\mathbf{x}}_{k|k-1}^i$ and historical reliability values \mathbf{s}_{k-1} .

c) As the new measurement y_k arrives, update each particle weight by substituting the difference between the predictions $\{\hat{y}_{k|k-1}^i\}_{i=1}^{N_s}$ and the measurement y_k into a Gaussian likelihood equation:

$$\omega_k^i \propto \omega_{k-1}^i p(y_k | \hat{\mathbf{x}}_{k|k-1}^i) \quad (15)$$

$$p(y_k | \hat{\mathbf{x}}_{k|k-1}^i) = \frac{1}{\sqrt{2\pi R}} \cdot \exp\left\{-\frac{(y_k - \hat{y}_{k|k-1}^i)^2}{2}\right\} \quad (16)$$

d) Normalize the particles weights:

$$\omega_k^i = \omega_k^i / \sum_{j=1}^{N_s} \omega_k^j \quad (17)$$

e) Perform resampling to eliminate the low-weighted particles and reproduce high-weighted particles [21], generating a new random particles set $(\hat{\mathbf{x}}_k^i, 1/N_s)$.

(3) Output the parameter estimation at time k

$$\hat{\mathbf{x}}_k = \sum_{i=1}^{N_s} \hat{\mathbf{x}}_k^i / N_s \quad (18)$$

(4) Train the SVR model with the updated parameter vector $\hat{\mathbf{x}}_k$, to get the prediction \hat{y}_{k+1} of the future time step $k+1$.

In practical applications, one issue to be considered is how to obtain the initial SVR parameters. According to whether the historical data is sufficient or not, two kinds of approaches can be considered here: the exhaustive searching approaches such as the Grid-Searching method with historical data and the expert based approaches such as the AS method [22], when there is no sufficient historical data.

2.2. Overview about the PCC method

PCC is an approach derived from Bezdek's Fuzzy C -Means (FCM) algorithm [23]. It reformulates the fuzzy clustering partition [24, 25], which limits that one pattern can only belong to one class, and avoids the trivial solution that all memberships are equal to zero [26]. Instead, an elastic possibilistic partition is proposed, so that it can be used to generate memberships that have a typicality interpretation as a confidence measure.

Let \mathbf{U} denote a partition matrix generated by the PCC algorithm. The elements u_{ij} of \mathbf{U} are subject to the following conditions [27]:

$$\begin{aligned} u_{ij} &\in [0, 1], \text{ for all } i \text{ and } j, \\ 0 &< \sum_{j=1}^N u_{ij} \leq N \text{ for all } i, \text{ and} \\ \max u_{ij} &> 0 \text{ for all } j. \end{aligned} \quad (19)$$

Here, u_{ij} are the membership values of pattern \mathbf{x}_j to cluster i ; N is the size of the data set. Then, the purpose of possibilistic clustering is to assign high membership values to patterns which are good representatives of the clusters and low membership values to non-representative patterns. The objective function which meets this purpose can be formulated as follows [28]:

$$J_m(\mathbf{U}, \mathbf{v}) = \sum_{i=1}^C \sum_{j=1}^N (\mu_{ij})^{r_m} s_{ij} + \sum_{i=1}^C \eta_i \sum_{j=1}^N (1 - \mu_{ij})^{r_m} \quad (20)$$

where S_{ij} is the distance of pattern \mathbf{x}_j from the prototype \mathbf{v}_i (the center of i -th cluster), η_i are suitable positive numbers and r_m determines the fuzziness of the final possibilistic partition and the shape of the possibility distribution (if $r_m \rightarrow 1$, the membership functions are crisp, if $r_m \rightarrow \infty$, they are maximally possibilistic). Here, the first term demands that the distances S_{ij} be as low as possible, whereas the second term requires that u_{ij} be as large as possible, thus avoiding the trivial solution. The selection of η_i will be discussed later.

The minimization of the objective function in Eqs. (20) leads to optimal possibilistic membership functions of the form:

$$\mu_{ij}^* = \frac{1}{1 + \left(\frac{S_{ij}}{\eta_i}\right)^{\frac{1}{r_m-1}}} \quad (21)$$

It is obvious that the u_{ij} obtained from Eqs. (21) lies in the range that Eqs. (19) imposes as constraint.

The value of η_i determines at which distance the membership of a pattern to the cluster i can be assigned a value of 0.5 (like “the 3 dB point” in confidence analysis) and, thus, it influences the shape of the cluster. In this sense, it needs to be chosen according to the desired “bandwidth” of the possibility (membership) distribution for each cluster. On the other hand, η_i determines the relative importance of the second term compared with the first term. In practice, η_i is taken in proportion to the average fuzzy distance of all patterns to cluster i [28]:

$$\eta_i = K \frac{\sum_{j=1}^N \mu_{ij}^{r_m} S_{ij}}{\sum_{j=1}^N \mu_{ij}^{r_m}} \quad i = 1, 2, \dots, c \quad (22)$$

where K is typically chosen unitary and the membership u_{ij} is usually obtained by the standard Fuzzy C-Means (FCM) algorithm [25, 27, 29].

The updating of the prototypes relies on the chosen distance measure, since different distance measures lead to different algorithms. For the most common case, the distance is described by an inner product induced norm metric, as in the FCM algorithm, i.e.

$$s_{ij}^2 = (\mathbf{x}_j - \mathbf{v}_i)^T \mathbf{M}_i (\mathbf{x}_j - \mathbf{v}_i) \quad (23)$$

The prototype is, then, updated as:

$$\mathbf{v}_i^* = \frac{\sum_{j=1}^N (\mu_{ij}^*)^{r_m} \cdot \mathbf{x}_j}{\sum_{j=1}^N (\mu_{ij}^*)^{r_m}} \quad i = 1, 2, \dots, c \quad (24)$$

From Eqs. (21), (22), (24), the procedures for the standard possibilistic clustering algorithm are:

1. Set the number of clusters c ;
2. Set the fuzziness index r_m ;
3. Perform an initial partition of the patterns into c fuzzy clusters, e.g. by using the standard Fuzzy C-Means (FCM) algorithm [25].

4. Approximate η_i using Eqs. (22).
5. With the given $\mathbf{U}^* = \{u_{ij}^*\}$, compute the optimal prototypes \mathbf{v}_i^* , $i = 1, 2, \dots, c$ from Eqs. (19);
6. With the optimal $\mathbf{v}^* = \{\mathbf{v}_i^*\}$, compute the optimal membership values u_{ij}^* , $i=1, 2, \dots, c, j=1, 2, \dots, N$ from Eqs. (21).
7. Repeat steps 5 and 6, until no change of $J_m(\mathbf{U}, \mathbf{v})$ is detected from one iteration t to the next or until the maximum change of the membership functions from one iteration t to the next is below a predefined threshold ε , i.e.

$$\max |(\mu_{ij}^*)^{t+1} - (\mu_{ij}^*)^t| \leq \varepsilon \quad (25)$$

3. THE NOVEL MULTI-STATE DYNAMIC SVR METHOD

In this section, a detailed description about our proposed Multi-state Dynamic SVR is provided. Like all the dynamic prediction methods, this proposed method is implemented in two stages: I) the off-line stage training the prior knowledge, i.e., the historical time series of reliability evolution trajectories, to construct the state classification model, II) and the online stage dynamically predicting the system reliability based on the posterior measurement. Firstly, a flowchart of the proposed approach is given as Fig. 2 for clarity and more specific details are demonstrated in the following.

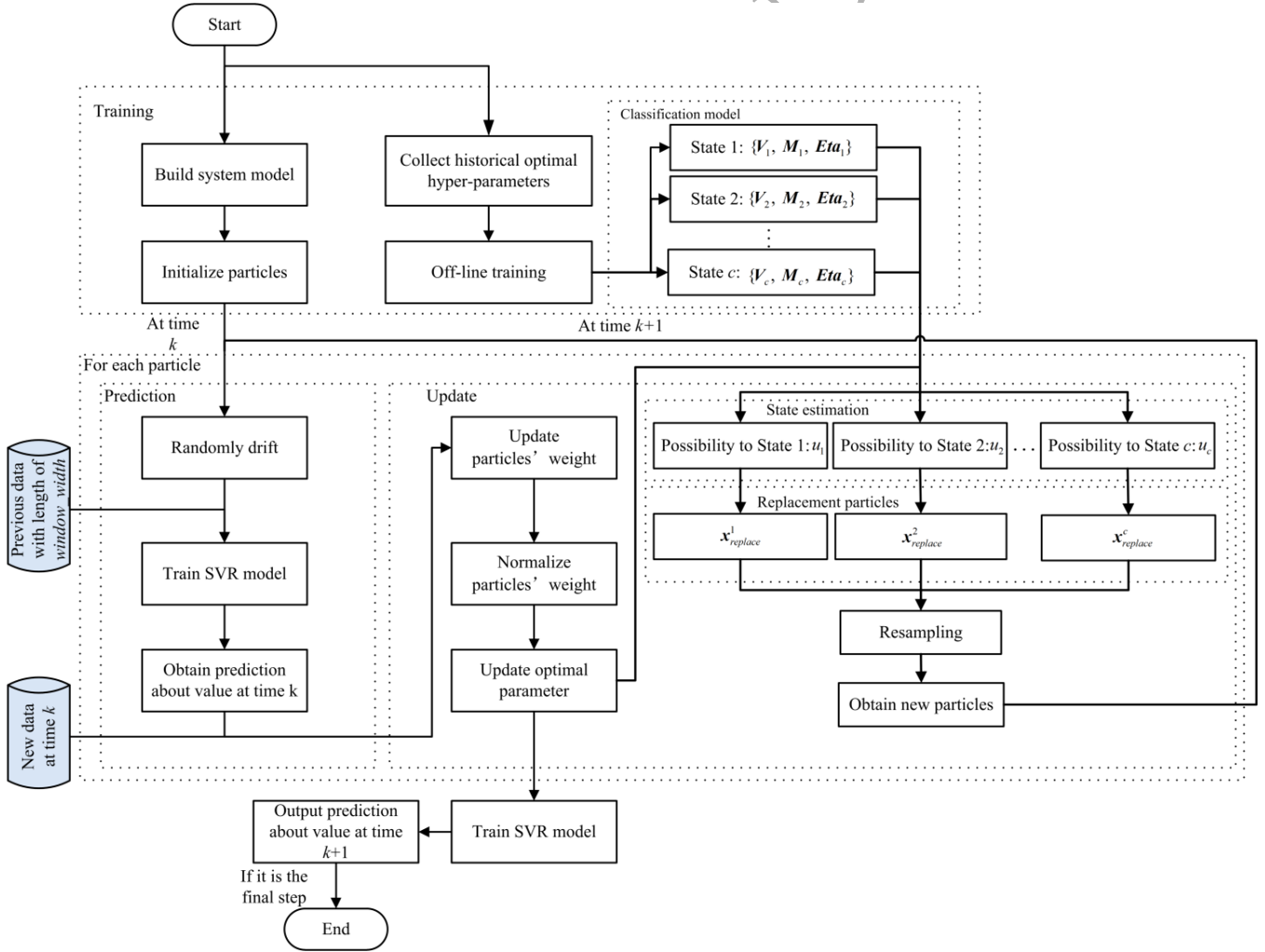


Fig.2 Flowchart of the proposed Multi-state Dynamic SVR method

3.1. Off-line training stage:

As stated in previous sections, in this work, the historical reliability trajectories and their true state are assumed known. Then the prior information remained to be trained in the off-line stage can be noted as Eqs. (1) and the objective in this stage is to build a classification model $\Delta: \mathcal{A} \rightarrow \Gamma$.

For this, it is requested to first select features \mathbf{F} from \mathcal{A} , and then learn the relationship between the extracted features \mathbf{F} and their labeled state, thus the construction of classification model can be rewritten as:

$$\Delta: \mathbf{F} \rightarrow \Gamma \quad (26)$$

which actually represents a supervised learning process.

Step 1: Feature selection

It is necessary, prior to any training, to select proper features that competently describe the difference between samples with different label. In this paper, the selected feature to characterize time series trends resorts to hyper-parameters, noted as $\mathbf{x} = [\sigma, C, \varepsilon]$, of SVR model that optimally fit the concerned time series. The feasibility of this “optimal hyper-parameters” feature derives from a notable fact that since the hyper-parameters are critically important to the prediction performance of SVR model, the hyper-parameters of SVR model optimally fitting time series with different trend will distribute in different area within the parameters space. In another words, the distribution of the optimal hyper-parameters can actually stands for a statistical description identifying the relationship between time series and their state. Within the PF-SVR paradigm, this “optimal hyper-parameters” feature can achieve satisfactory performance, including the capability of characterizing and availability of implementation.

For the time series $ts_j^i \in \mathcal{A}$, their features \mathbf{x}_j^i are firstly collected. Since the computational cost is not that important in the off-line stage, the precise but computation consuming high-dense mesh-refinement Grid Search method through the Cross-Validation scheme are considered. Then, a collection of optimal hyper-parameters $\mathbf{F} = \{\mathbf{x}_1^1, \dots, \mathbf{x}_1^{L_1}, \dots, \mathbf{x}_j^i, \dots, \mathbf{x}_j^{L_j}, \dots, \mathbf{x}_c^1, \dots, \mathbf{x}_c^{L_c}\}$ can be extracted from \mathcal{A} and is regarded as the labeled features.

For ease of clarity, Fig. 3 shows an example of collecting the optimal hyper-parameters from time series with two states of conditions.

Step 2: Classification model construction.

With the selected features \mathbf{F} , this step aims to construct the classification model.

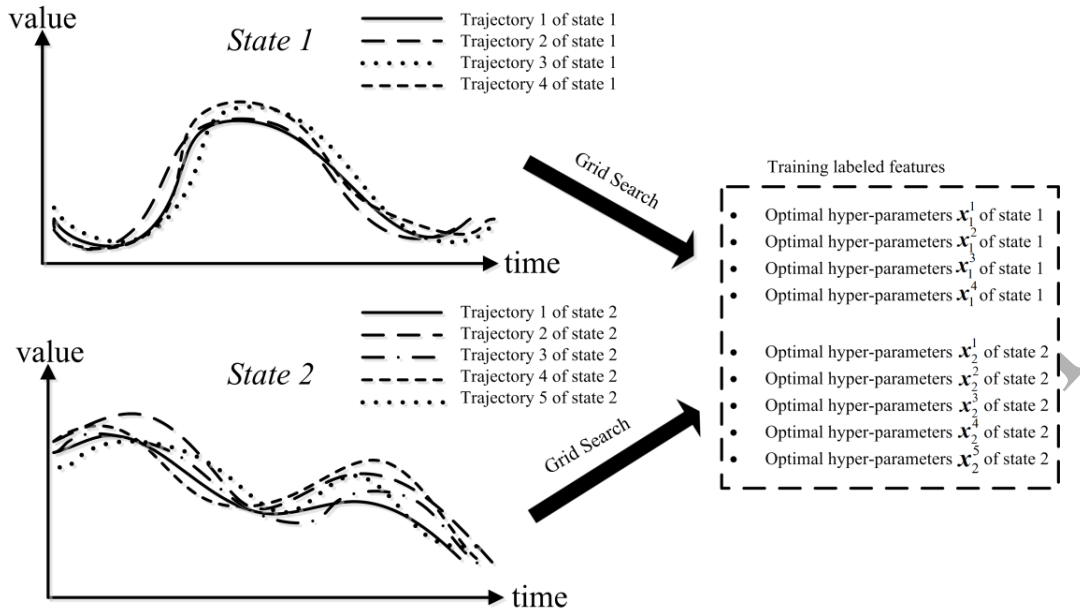


Fig. 3 An example of extracting the “optimal hyper-parameters” features for time series with two potential states

Through the PCC method, the classification model $\Delta: F \rightarrow F$ is realized by assigning each feature x a membership $U = \{u_1, \dots, u_c\}$ where u_i represents the probability it belongs to class i . Denoting the calculation of the membership as:

$$U = \Phi(x, B)$$

$$B = \{B_1, \dots, B_c\} = \{(v_1, M_1, \eta_1), \dots, (v_c, M_c, \eta_c)\} \quad (27)$$

where $B_i = (v_i, M_i, \eta_i)$ is the statistical description used by PCC about class i that v_i represents the prototype (class center) of each class, M_i represent the distance measure covariance, and η_i represents a tuned positive number controlling the degree of concentration. Then, for a given test feature x , its membership U can be calculated through $\Phi(\cdot)$, which corresponds to the procedures of Eqs (21), (22), (23).

As Fig. 4 shows, the class description $B = \{(v_1, M_1, \eta_1), \dots, (v_c, M_c, \eta_c)\}$ can be intuitively represented as ellipses in the features space, where the ellipse centers represent the prototype v_i of each state and the dots on the ellipse edge are assigned a membership value of 0.5 to the corresponding prototype which actually represent the distance measurement matrix M_i and constant η_i of each state. The details of how to acquire v , M and η on the basis of F can refer to Section 2.2.

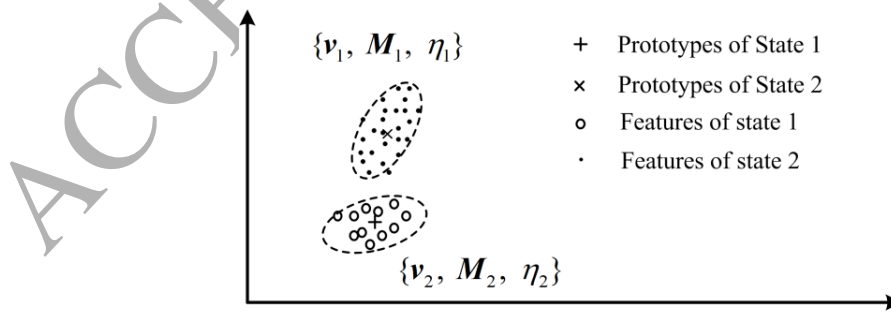


Fig. 4 An illustration of PCC classification model

3.2. Online reliability prediction stage:

As stated in Section 2, prediction problem considered in this paper can be defined as:

$$\hat{y}_{k+1} = f_k(\mathbf{s}_k) = f_k(y_k, y_{k-1}, \dots, y_{k-p+1}) \quad (28)$$

where the output \hat{y}_k is the prediction of time series value at time step k , the input vector \mathbf{s}_k is composed of the p -lagged reliability values in p previous time steps and $f_k(\cdot)$ represents the prediction model in time step k .

In the online stage, it is assumed that at each new time step the measurement of the last time step becomes available and the underlying evolution trend of system reliability is changed which means the prediction model shall be adjusted correspondingly. Thus, the objective in this stage is to, first, update the system state estimation on the basis of new measurement and, second, update the prediction model according to the state estimation.

As time step $k+1$ arrives, with the newly available measurement value y_k , the state estimation \mathbf{U} at time step k can be calculated. Denote the membership calculation process as:

$$\mathbf{U}_k = \ell(y_k | \mathbf{B}) \quad (29)$$

Then the prediction model is also recursively updated according to the updated membership and optimal prediction models of each state:

$$f_{k+1}(\cdot) = \hat{h}(f_k(\cdot) | \mathbf{U}_k) \quad (30)$$

Therefore, the aforementioned objectives in this stage can be mathematically stated as equivalent to the construction of ℓ and \hat{h} .

In this paper, the prediction model is constructed within the SVR paradigm, thus it can be actually described by its hyper-parameters for given training samples (for more details refer to Section 2.1). Denote the prediction model built by SVR with the hyper-parameters \mathbf{x} and the training dataset \mathbf{Y} as $f_{\mathbf{x}, \mathbf{Y}}(\cdot)$, then the prediction model can be rewritten as:

$$\begin{aligned} f_{k+1}(\cdot) &\triangleq f_{\mathbf{x}_{k+1}, \mathbf{Y}}(\cdot) \\ f_k(\cdot) &\triangleq f_{\mathbf{x}_k, \mathbf{Y}}(\cdot) \end{aligned} \quad (31)$$

Hence, the problem of updating the prediction models stated as Eqs (30) turns to a problem of updating the hyper-parameters:

$$\mathbf{x}_{k+1} = \hat{h}(\mathbf{x}_k | \mathbf{U}_k) \quad (32)$$

Besides, the optimal SVR hyper-parameters $\mathbf{x}_{optimal}$ is also regarded as the state feature in this paper, so the problem of calculating the memberships stated as Eqs (29) also turns to a problem of calculating the hyper-parameters:

$$\begin{aligned} \mathbf{x}_{optimal}^k &= \ell'(y_k, \mathbf{x}_k) \\ \mathbf{U}_k &= \Phi(\mathbf{x}_{optimal}^k, \mathbf{B}) \end{aligned} \quad (33)$$

Therefore, a link between the measured reliability data and the state estimation is built through the SVR hyper-parameters, as Fig. 5 illustrates. Then, the constructions of ℓ' and \hat{h} are presented in the following parts.

Step 3: System state estimation: construction of ℓ' .

In this paper, the prediction model is constructed within the PF-SVR paradigm, thus the hyper-parameters can be represented by a set of particles (for more details refer to Section 2.1). Denoting the particle set at time step k as:

$$Particles_k \triangleq \left\{ \mathbf{X}_k : (\mathbf{x}_k^1, \dots, \mathbf{x}_k^N), \mathbf{W}_k : (w_k^1, \dots, w_k^N) \right\} \quad (34)$$

Then the predictions of y_k before it becomes available by each particle can be represented as:

$$\hat{y}_k^i = f_{\mathbf{x}_k^i, \mathbf{Y}}(y_{k-1}, y_{k-2}, \dots, y_{k-p}), \mathbf{Y} = \{y_{k-\text{window_size}+1}, \dots, y_{k-1}\} \quad (35)$$

It is notable that only a properly selected length, denoted as *window_size*, of previous data is utilized as the training data to generate the prediction model. Otherwise, a too short training window is not sufficient to depict the characteristics of current system state; and a too long training window will significantly increase the computational cost and, more importantly, the state estimation about the current system could be obscured because of the overlap effect of other previous states.

Once the time step updates to time step $k+1$, the true measured data at time step k , y_k , is available, so the weight vector can be updated as $\mathbf{W}_k \rightarrow \mathbf{W}_k$, by substituting the difference between the output predictions $\{\hat{y}_1^k, \dots, \hat{y}_N^k\}$ and the true measurement y_k into Eqs. (15), (16), (17). Thus the optimal hyper-parameters for time step k is obtained as:

$$\mathbf{x}_{\text{optimal}}^k = \sum w_k^i \mathbf{x}_k^i \quad (36)$$

By the Eqs (36), the construction of \mathcal{L}' is fulfilled. Given this updated optimal SVR parameter $\mathbf{x}_{\text{optimal}}^k$, the membership indicating the possibility that the current system belonging to each state can be represented as:

$$\begin{aligned} \mathbf{U}_k &= \{u_1^k, u_2^k, \dots, u_c^k\} \\ u_i^k &= \frac{1}{1 + \left(\frac{s_i}{\eta_i}\right)^{\frac{1}{m-1}}} \\ s_i &= (\mathbf{x}_{\text{optimal}}^k - \mathbf{v}_i)^T \mathbf{M}_i^{-1} (\mathbf{x}_{\text{optimal}}^k - \mathbf{v}_i) \end{aligned} \quad (37)$$

Step 4: Prediction model updating: construction of $\hat{\mathbf{h}}$.

Once the system state estimation at time step k is characterized by the membership \mathbf{U}_k , the prediction model is requested to be updated according to \mathbf{U}_k .

As we have stated, within the PF-SVR framework, the updating of prediction model means the updating of SVR hyper-parameters, which resorts to a sequential evolution process of a set of particles, thus:

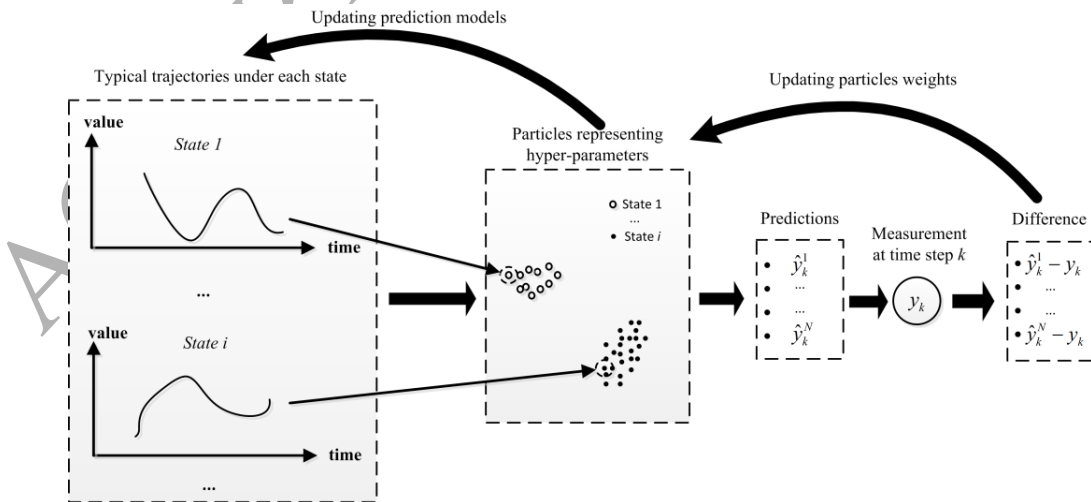


Fig. 5 An illustration of how the SVR hyper-parameters connect the measurements and prediction model

$$\mathbf{x}_{k+1} = \hat{h}(\mathbf{x}_k | \mathbf{U}_k) \rightarrow \text{Particles}_{k+1} = \hat{h}(\text{Particles}_k | \mathbf{U}_k) \quad (38)$$

In order to use the state estimation result \mathbf{U}_k to improve the evolution process of particles, a novel Replacement Operation is proposed and is performed with the following procedures:

A) Generating the replacement particles. At a general time step k , a collection of replacement particles, noted as $\mathbf{X}_{replace}^k = \{\mathbf{x}_{replace}^{k-1}, \dots, \mathbf{x}_{replace}^{k-c}\}$ where $\mathbf{x}_{replace}^{k-i}$ denotes the particles generated from state i , is obtained based on the membership \mathbf{U}_k and classification model $\{\mathbf{V}, \mathbf{M}, \mathbf{Eta}\}$. The total size of $\mathbf{X}_{replace}^k$ is set as:

$$N_{replace} = N \cdot \text{replace_rate} \quad (39)$$

where N is the particles number and replace_rate is a preset coefficient controlling the total amount of replacement particles, and the size of $\mathbf{x}_{replace}^{k-i}$, i.e., number of particles generated from state i , which is proportional to the corresponding membership value u_i , is quantified as:

$$N_{replace}^{k-i} = N_{replace} \cdot u_i^i, \quad i = 1, 2, \dots, c \quad (40)$$

Then, $\mathbf{x}_{replace}^{k-i}$ is generated by:

$$\mathbf{x}_{replace}^{k-i} = \text{Gaussian}\{\mathbf{v}_i, \mathbf{M}_i, N_{replace}^{k-i}\}, \quad i = 1, 2, \dots, c \quad (41)$$

where $\text{Gaussian}\{\mathbf{v}, \mathbf{M}, N\}$ is a Gaussian generating function that draws N Gaussian distributed samples with center on \mathbf{v} and covariance of \mathbf{M} .

B) Replacing old particles. In the resampling step, only $(N - N_{replace})$ particles are resampled from the old particles. Denoting the collection of resampled particles as $\mathbf{X}_{resample}^k$, the updated collection of particles are then composed of the $(N - N_{replace})$ resampled particles and $N_{replace}$ replacement particles:

$$\mathbf{X}_{k+1} = [\mathbf{X}_{replace}^k, \mathbf{X}_{resample}^k] \quad (42)$$

This Replacement Operation plays an important role in improving the particles evolution process. As illustrated in Fig. 6, in the case that the current system is characterized with a high membership value to a particular state, which is intuitively reflected by that the current $\mathbf{x}_{optimal}^k$ is “close” to a specific cluster center (see Fig. 6a), then large number of replacement particles will be generated from this cluster and replace the old particles, so that the particles will converge quickly to this cluster center and gain the resistance

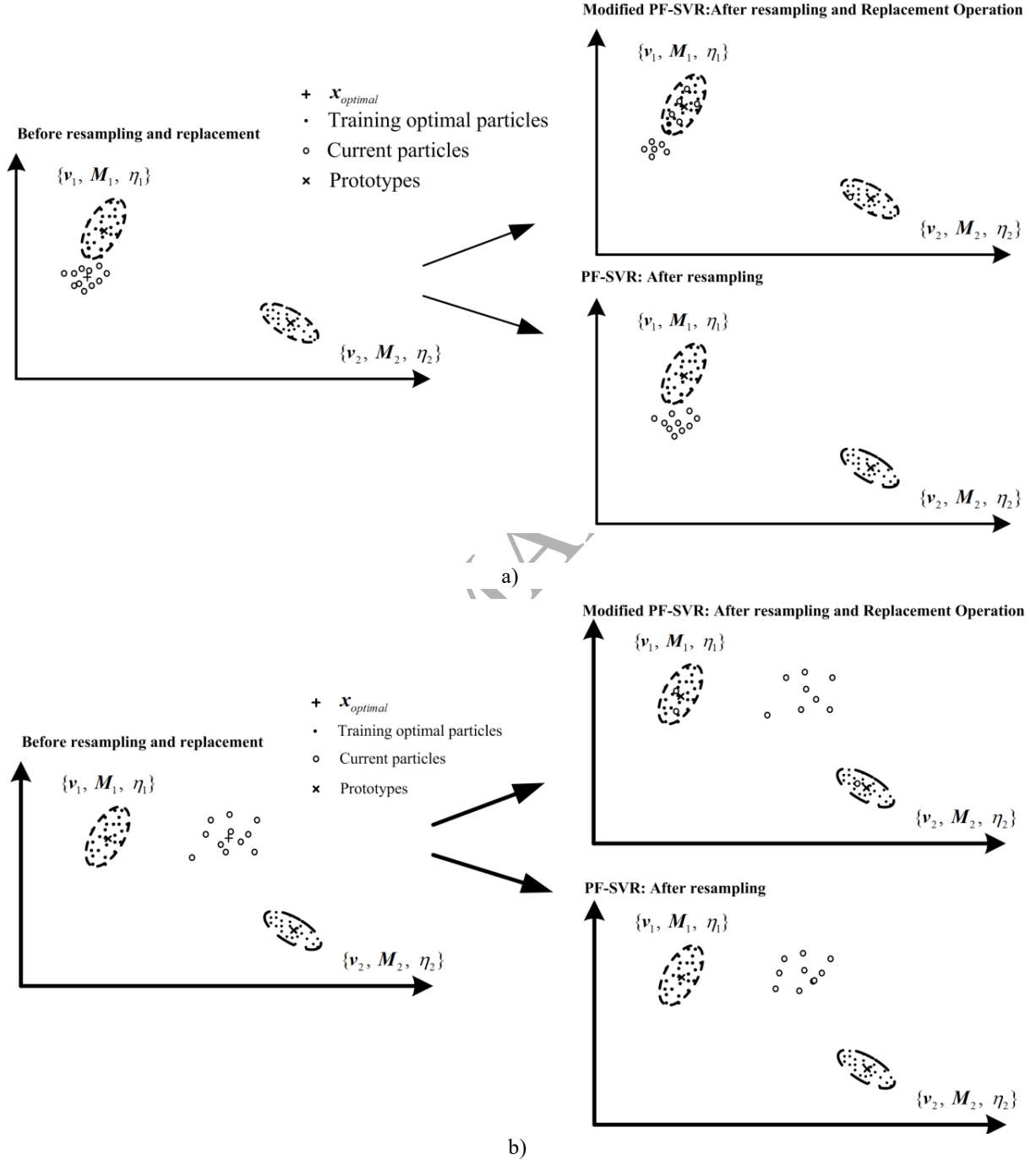


Fig. 6 Particles evolution with/without Replacement Operation in two situations: a) with clear state estimation, b) with ambiguous state estimation.

to other local optima. So, in this case, the evolution process of particles is significantly improved by the replacement particles, namely, the state estimation knowledge. On the contrary, the origin PF-SVR takes no advantage of the state estimation knowledge, thus the evolution of particles mainly relies on the random drift, which tends to show a comparatively slower convergence and the

vulnerability to local optima. On the other hand, in another case that the current system is characterized with membership all low or nearly homogeneous to each state (see Fig. 6b), which means there is no clear state assignment can be provided, then few replacement particles are generated, thus the proposed method degrades to the original PF-SVR method. In this case, the proposed method inherits the searching ability of the original PF-SVR.

At last, it is worth noting that since the Modified PF-SVR introduces additional particles with satisfactory quality and variability into the particles set, to an extent it can be also regarded as a considerable remedy for an inherent drawback of origin PF method: the Sample Impoverishment. The improvement of the proposed method on another drawback of PF method, the Divergence, will be demonstrated later through some experiments.

3.3. Illustrative Pseudo-code and flowchart

For better completeness, the pseudo-code of the proposed method is given in the following.

Pseudo-code: The implementation of the Multi-state Dynamic SVR

Off-line training:

Step 1, Obtain the $\{V, M, Eta\}$ by PCC method.

Step 2, Initialize the particles of SVR hyper-parameters: $\{X, W\}$;

Online iteration:

Time Step= k : Denotes the particles of SVR hyper-parameters as $\{X_k, W_k\}$;

Time Step= $k+1$: Measurement at Time Step k , y_k , is available;

Step 3, Randomly drift the particles $X_k \rightarrow X_k$;

Step 4, Update the weights of new particle set via:

Step 4.1, Build SVR regression models with every particle from X_k , based on a data fragment with length of "window_width";

Step 4.2, Obtain the predictions of every particle at time k : $\{\hat{y}_{k|k-1}^i\}_{i=1}^{N_k}$;

Step 4.3, Update $W_k \rightarrow W_k$ by eqs (15), (16), (17);

Step 5, Obtain the updated optimal parameters vector $x_{optimal}^{k+1}$ as the weighted sum of X_k with weights W_k ;

Step 6, Predict \hat{y}_{k+1} with $x_{optimal}^{k+1}$;

Step 7, Calculate the memberships $U = \{u_1, u_2, \dots, u_k\}$ of $x_{optimal}^{k+1}$ to each state;

Step 8, Perform the resampling and replacement operation, to obtain the new particles set $\{X_{k+1}, W_{k+1}\}$;

Step 9, Back to Step 3.

Here, the parameter *window_width* determines the size of back forwards data used as training data to build the prediction model.

4. CASE STUDIES

In this section, the potentiality and strengths of the proposed approach is demonstrated by experiments carried on some illustrative case studies. All the experiments in this paper adopt the single-step-ahead time series prediction, which regards each sample s_k as the output of an underlying mapping function and the sample in the previous time step s_{k-1} . Then, the relationship between the prediction at the next time step and the input at the current time step is described by the following equation:

$$\hat{y}_{k+1} = f_k(y_k) \quad (43)$$

4.1. Case study I: Lithium-ion battery test datasets under dynamical multi-state operating profiles

In this section, the time series consisting of Lithium-ion battery monitoring data are firstly concerned. Due to the advantages of high energy density, long cycle life, low self-discharge rate, low weight and no memory effect, etc. [30], Lithium-ion (Li-ion) batteries have become the hot-topic rechargeable power supply for various kind of electronic devices. Especially, the Li-ion battery is getting most focused in recent years as the hybrid electric vehicles (HEVs) and plug-in hybrid electric vehicles (PHEVs) are drawing more and more attentions from consumers and manufacturers since the growing awareness of environment friendliness,

demand of exploiting alternative energy and interest in “intelligent drive” [31]. Like all the other electronic components, the Li-ion batteries degrade when utilized for various reasons. Thus, it is important to reasonably manage the health of batteries [32].

Two typical datasets with Li-ion battery operated under multi-state conditions are considered in this case study:

1) Dataset 1: the inner temperature of Li-ion batteries in different charge-discharge profile. Keeping monitoring on the battery inner temperature is very important for battery health management because of two reasons. First, the inner temperature will significantly impact the degradation process of battery performance. Then, a comparatively high inner temperature will pose extra risk of some heavy accident such as explosion. Thus, for a battery management system (BMS) aiming at increasing battery life and predicting battery failure, the thermal management is indispensable, especially for high-temperature batteries [33]. So, Dataset 1 firstly considers the online prediction problem of Li-ion batteries inner temperature in different charge-discharge profiles.

2) Dataset 2: the SOH data of Li-ion batteries under different state of charge range. Dataset 2 considers another crucial CM signal, the SOH of battery, which is directly related to battery performance and usually used to predict the battery RUL. The SOH of battery represents the ratio of the maximum stored energy a battery can provide during a full-charge-full-discharge process in a specific moment compared with the nominal capacity [34]. For batteries operated in charge-discharge cycles, the cycling operation reduces its SOH through a variety of concurrent failure mechanisms [35-37]. If the SOH is lower than the preset threshold, the battery is identified as failed to continue providing reliable performance in the designated applications. Thus, the capacity of a battery can be regarded as a direct indicator to evaluate its reliability. In recent years, the prediction of battery SOH has become an important standalone research field on prognostics and health management of battery. The degradation of battery SOH is highly dependent on the charging-discharging conditions such as the range of State of Charge (SOC), where the SOC usually quantifies the ratio between the current available capacity and the rated capacity. In most practical applications, batteries undergo charge-discharge cycle for only partial State of Charge (SOC) ranges, but not full SOC (0%-100%), and in different SOC ranges the batteries will show different degradation behavior. Saxena, et.al., have thoroughly studied this phenomenon, and obtained sufficient experiment results demonstrating the effects of different SOC ranges on battery capacity degradation [38]. In this dataset, these experimental results are introduced as the training dataset to build the classification model, and capacity values of a battery working under a multi-state condition consisting of different SOC ranges is considered as the test dataset

4.1.1. Description of data

1) Dataset 1.

Dataset 1 comes from a set of tests on Li-ion batteries from the NASA PCoE (Prognostics Center of Excellence) database [18]. For each test, the battery is run undergoing a charge-discharge cycle through 3 different operational profiles: 1) constant current (CC) charge at current of 1.5A until the battery voltage reaches 4.2V, 2) constant voltage (CV) charge at battery voltage of 4.2V until the charge current drops to 20mA, and 3) discharge carried out using a 0.05Hz square wave loading profile of 4A amplitude and 50% duty cycle until the battery voltage falls to 2.7V. The transition between each operational mode is instantaneous.

We take 36 independent charge-discharge cycles into consideration, with 35 cycles regarded as the training cycles on which the training of optimal particles (SVR hyper-parameters) for each operational mode is based, and the remaining one as the test cycle on which the Multi-state Dynamic SVR is performed. The total time steps and the number of optimal particles for each operational mode in each cycle are different, as listed in Table 2.

Table 2: Length of each operational state for Dataset 1

Cycles	Length of each states		
	CC Charge	CV Charge	Discharge
1	119	147	65
2	109	138	59
3	115	151	71
4	104	129	68
5	111	144	63

2) Dataset 2.

Dataset 2 comes from the experiments results of literature [38]. In this study, the commercial graphite/LiCoO₂ pouch cells with a nominal capacity of 1.5 Ah and a nominal voltage of 3.7 V were used. The end-of-charge voltage is specified as 4.2 V and the end-of-discharge voltage is specified as 2.75V. The charging-discharging cycling operation is performed through an Arbin BT2000 Battery Tester with 16 independent channels.

To understand the battery degradation behavior in different SOC ranges and build the classification model, the selected batteries were tested under different SOC ranges with different mean SOC and Δ SOC values, which are listed as Table 3. The experimental SOH values of batteries under charging-discharging cycles with these states of SOC ranges are presented in Fig. 7, which are used as the training dataset.

Table 3: States of SOC ranges for Dataset 2

States	SOC range	Mean SOC	Δ SOC
1	0%-100%	50%	100%
2	20%-80%	50%	60%
3	0%-60%	30%	60%
4	40%-100%	70%	60%

The test dataset considers SOH values of a battery operated under a multi-state condition which is specified by resorting to randomly transferred states of SOC range and the real state of SOC range is unknown to users.

4.1.2. Application of proposed method

In accordance with the implementation procedures presented in Section 3, the application of the proposed method is described here step by step.

Step 1: For every training time series, the high-dense mesh-refinement Grid-Search method is applied within the 5-fold Cross Validation scheme to obtain the optimal SVR hyper-parameters, which are demonstrated in Fig. 8 and Fig. 9 by the dots and serves as the features of related states. We can see clearly that the features, namely the optimal SVR hyper-parameters, related to time series under different state distribute in significantly different area of the parameters space, which provides fairly supports to the availability of this optimal hyper-parameter features

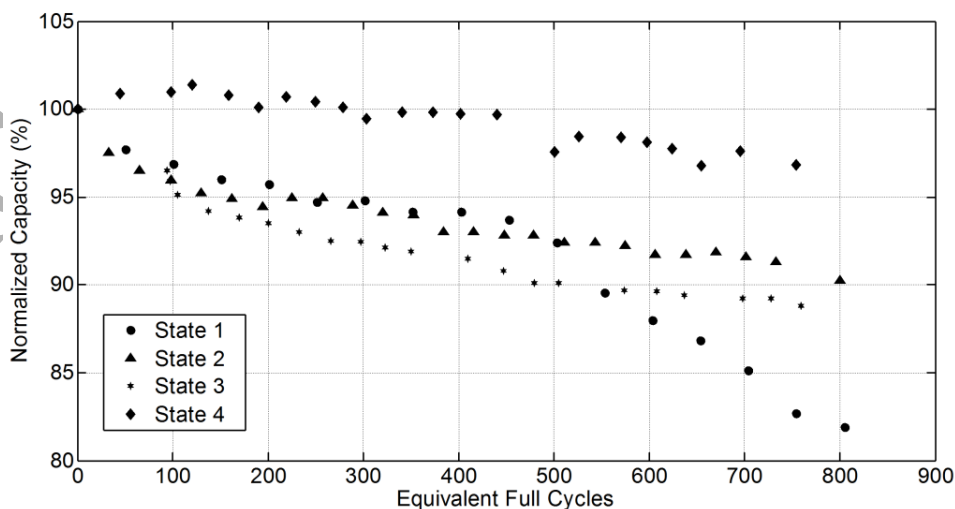


Fig. 7 Training trajectories of Dataset 2

Step 2: For the training cycles, the true operational profile is known. For both Dataset 1 and Dataset 2, we assign the particles a high membership values to the true operational profile (linearly vary from 0.5 to 0.95 as time steps increases, guaranteeing that the more sufficient trained particles have the higher membership values), and, on the contrary, a low membership values to the other wrong profiles (set to 0.05). Then, the possibilistic clustering classification can be performed with the clusters number C set equal to 4 and the fuzzy parameter r_m set equal to 3. The classification results, together with the optimal particles, are also shown in Figure 6 and Figure 7. As depicted in Section 3, the edges of ellipses represent the classification model $\{V, M, Eta\}$.

Step 3: Given the classification model obtained in Step 2, the proposed method is tested within the task of online prediction of dataset D . First, the current system state is estimated by substituting the current feature (the latest optimal SVR hyper-parameter) and the classification model into Eqs. (21), (22), (23). The probability of the system belongs to each state at every time steps for dataset 1 are calculated and shown in Fig. 10.

Step 4: With *replace_rate* set equal to 0.3 and N set equal to 100 from trial, the Modified PF-SVR proposed in this paper plays important roles in updating the prediction model. The updated prediction model makes use of both the information of system state estimation and the searching ability of PF-SVR, thus it is expected to achieve satisfactory prediction performance, which remains to be verified by the experiment results.

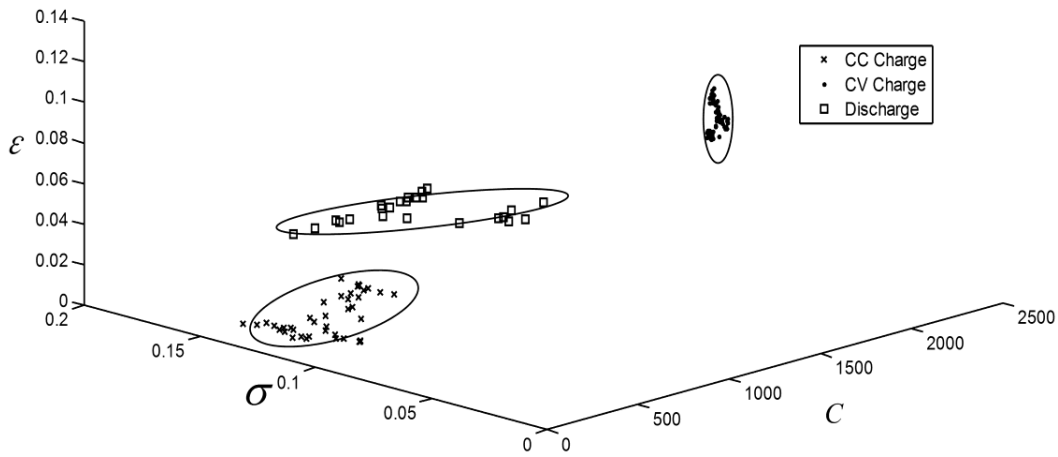


Fig. 8 Optimal SVR parameters and possibilistic clustering classification results for Dataset 1.

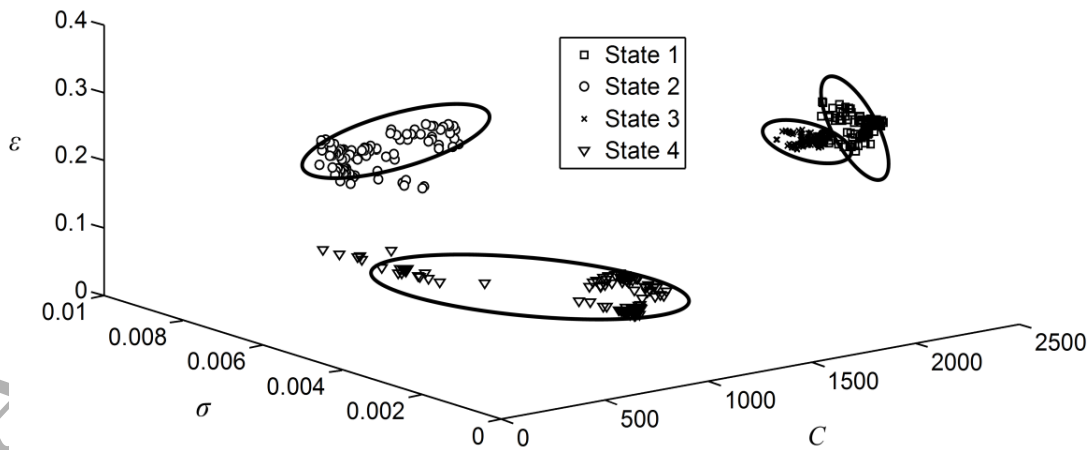


Fig. 9 Optimal SVR parameters and possibilistic clustering classification results for Dataset 2

4.1.3. Analysis of Experiment results

1) For Dataset 1:

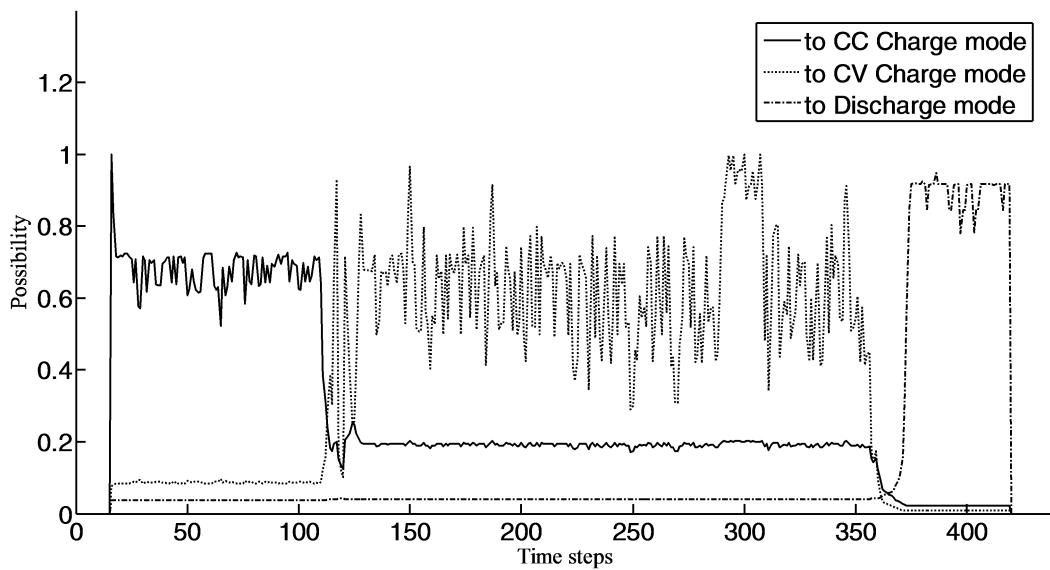


Fig.10 State estimation result for Dataset 1: possibility that the current system belongs to each state

Without loss of generality, Fig. 11a shows one trajectory of the prediction results with respect to PF-SVR, Multi-state Dynamic SVR and FGAPSO-SVR for the global test cycle. For further comparison of the performances after a state transition, the partial results of the time steps after 357 (transition from CV Charge state to Discharge state happens) is given as Fig.11b. In Fig. 11a, we can see clearly that the battery follows different thermal behaviors in different operational profiles. Fig.11b illustrates that the Multi-state Dynamic SVR method is more suitable for predicting this multi-state behavior, with more stable convergence to a normal state and quicker adaptation to changed state than the other two methods. Moreover, it is shown clearly that after the state transition at time step 357, both the PF-SVR and the FGAPSO-SVR experience a long “bad” period for more than 20 time steps, while the Multi-state Dynamic SVR adjusts to the “good” prediction after only 4 or 5 steps of “bad” predictions. Fig.13 gives the state estimation results and the *RMSEs* of predictions for 10 independent experiments are listed in Table 4 and Table 5, which hold the same conclusions as discussed in last case study.

Table 4: *RMSEs* for 10 runs in the real case for PF-SVR, Multi-state Dynamic SVR and FGAPSO-SVR (at normal states: Time Steps 21-357, 359-400)

Run Index	PF-SVR	Multi-state Dynamic SVR	FGAPSO-SVR
1	0.1195	0.1037	0.1064
2	0.1206	0.0928	0.1382
3	0.1342	0.0982	0.1304
4	0.1106	0.0977	0.1176
5	0.1582	0.1036	0.0984
6	0.1097	0.1120	0.1528
7	0.0861	0.0818	0.1321
8	0.0921	0.0919	0.1453
9	0.1409	0.1021	0.1189
10	0.1327	0.0997	0.1238

Mean	0.1205	0.0984	0.1264
Standard Deviation	0.0221	0.0082	0.0169

Table 5: *RMSEs* for 10 runs in the real case for PF-SVR, Multi-state Dynamic SVR and FGAPSO-SVR (at state transition: Time Steps 358-378)

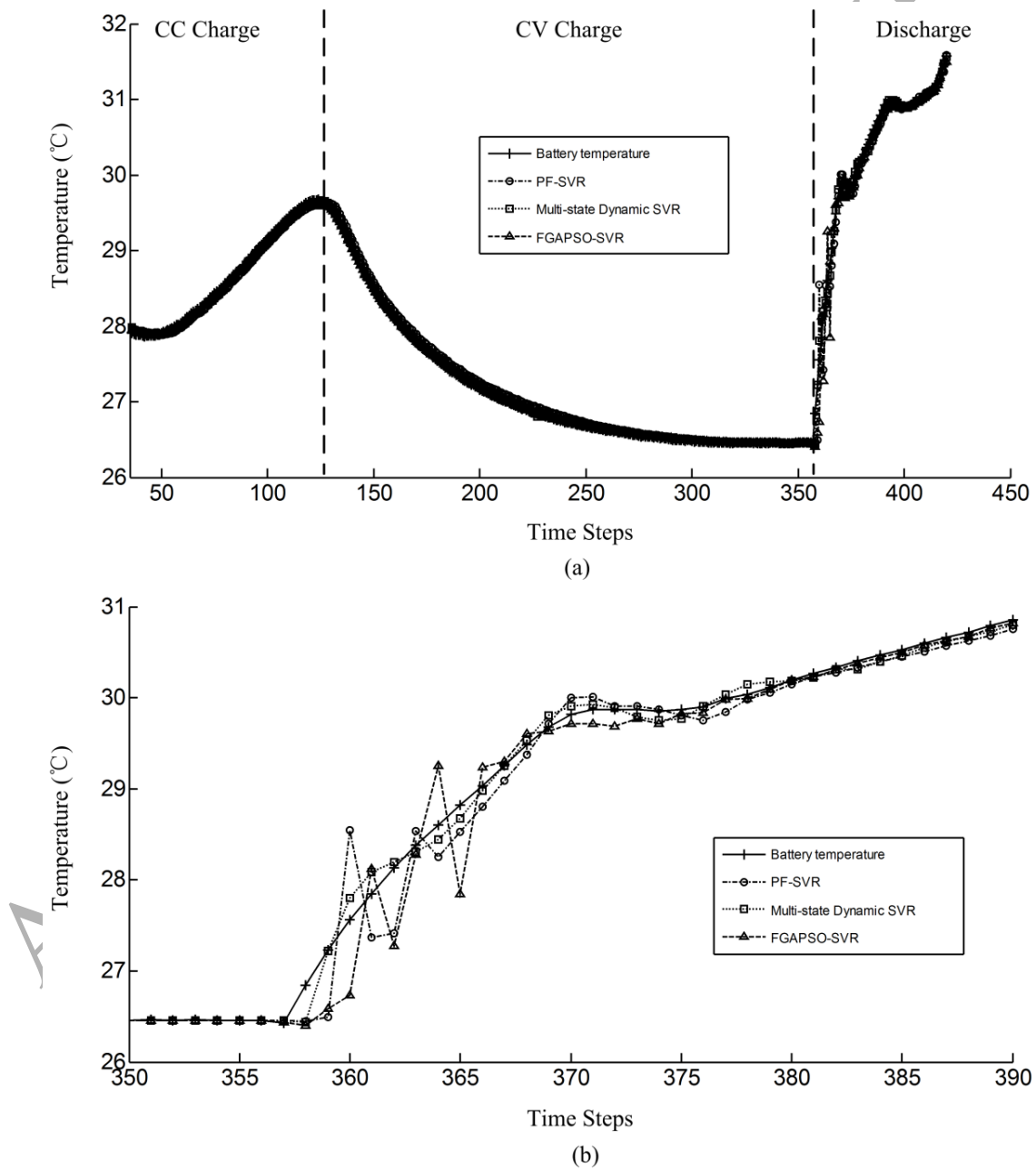


Fig. 11 Prediction results for Dataset 1: (a) global results, (b) partial results

Run Index	PF-SVR	Multi-state Dynamic SVR	FGAPSO-SVR
1	0.7149	0.1187	0.8021
2	0.8326	0.1320	0.9362
3	0.6938	0.0975	1.0134
4	0.7267	0.3124	0.8927
5	0.9518	0.1984	0.8825
6	1.0133	0.0871	0.7914
7	0.8620	0.2346	0.9217
8	0.9763	0.2289	0.8932
9	0.8982	0.1798	0.9275
10	1.0971	0.1545	1.1326
Mean	0.8767	0.1744	0.9193
Standard Deviation	0.1364	0.0708	0.0986

2) For Dataset 2:

Fig. 12 shows one trajectory of the predicted SOH by the proposed method, PF-SVR method and FGAPSO method, respectively, for the dataset 2. We can see that, though the predictions provided by these approaches are generally comparable in most time steps, the proposed method achieve obvious better performance after every state transition.

To further evaluate the performance of the proposed approach compared with the benchmark approaches, experiments are independently repeated for 10 times and the *RMSEs* of prediction results of the 10 tests are plotted in Fig. 13.

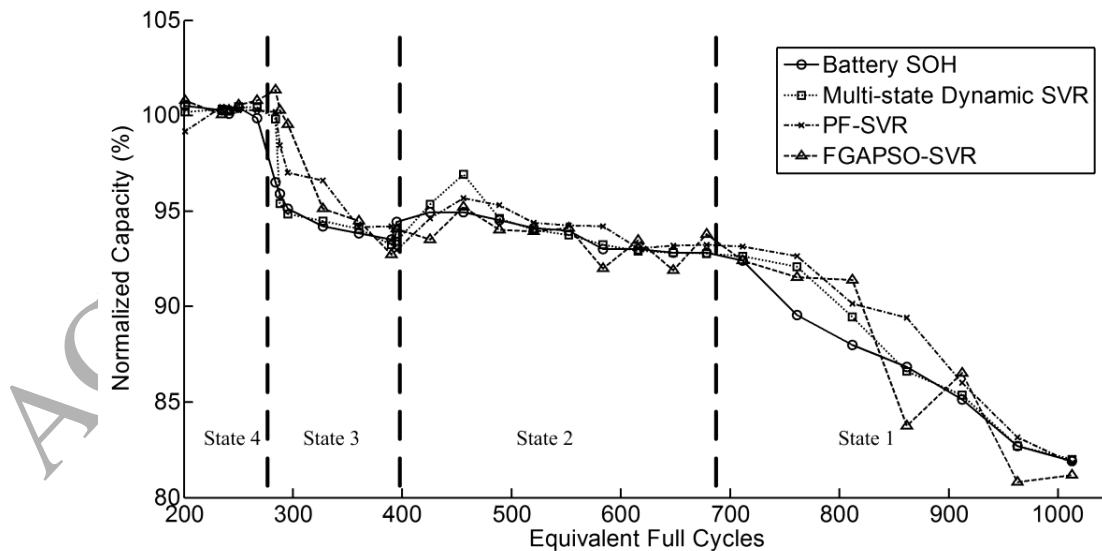


Fig. 12 The real and predicted battery SOH for one illustrative battery degradation trajectory.

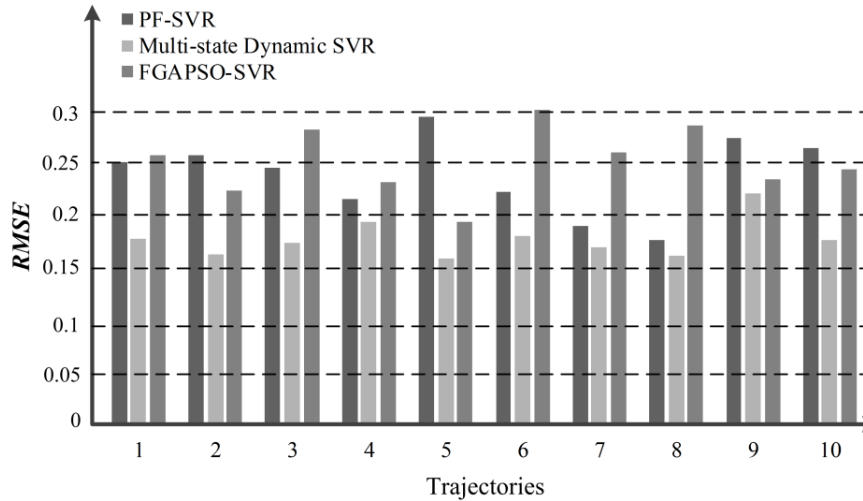


Fig. 13 Comparison of the $RMSE$ s for three approaches in 10 test trajectories for dataset 2

4.2. More reliability data concerning mechanical structures

More reliability data are considered in this section to illustrate the proposed method further. All the introduced reliability data are taken from the NASA PCoE dataset, as described in Table 6. For more details about these data, readers can refer to the website of NASA PCoE dataset [18]. Experiment results for this data are reported in Table 7, 8 (only the *Means* and the *Standard Deviations* of 10 $RMSE$ s are listed, at the normal time steps and the state transition time steps, respectively).

Table 6: More reliability data from NASA PCoE dataset

Name	Chosen numbers of states	CM indicators	Descriptions
Milling Data	4	Wear of the milling insert	Experiments on a milling machine for different speeds, feeds, and depth of cut.
Turbofan Engine Degradation Simulation Data	6	Wear degree	Engine degradation simulations on C-MAPSS for different combinations of operational conditions and fault modes.

Table 7: Means of 10 $RMSE$ s for more reliability datasets

Datasets	Normal State			State Transition		
	PF-SVR	Multi-state Dynamic SVR	FGAPSO-SVR	PF-SVR	Multi-state Dynamic SVR	FGAPSO-SVR
Milling Data	0.5764	0.5628	0.5521	2.3317	1.0349	1.5476
Turbofan Engine Degradation Simulation Data	0.1324	0.1241	0.1195	0.6097	0.3029	0.5781

Table 8: Standard Deviations of 10 $RMSE$ s for more reliability datasets

Datasets	Normal State			State Transition		
	PF-SVR	Multi-state Dynamic SVR	FGAPSO-SVR	PF-SVR	Multi-state Dynamic SVR	FGAPSO-SVR
Milling Data	0.0874	0.0574	0.1033	0.2173	0.0988	0.3284
Turbofan Engine Degradation Simulation Data	0.0288	0.0093	0.0217	0.1986	0.0974	0.1875

From both the real case experiments, it is shown that the advantages of the proposed Multi-state Dynamic SVR method lie in providing more stable predictions when system runs under a normal state (reflected by lower Standard Deviation values), and in quicker adaptation to new state when system experience a state transition (reflected by lower Mean values).

4.3. Analysis of computational complexity

In terms of the computational complexity of the proposed approach, it is worth noting that the PCC is an off-line procedure and the replacement operation brings little extra computational burden to the SVR modeling and predicting procedure. So the Multi-state Dynamic SVR method has an equivalent computational complexity with PF-SVR. The comparison of the total running time (with experiment platform of Microsoft Windows 7, Matlab 7.9.0, Intel 2.4 GHz) to predict the Li-ion Battery dataset and the artificial test dataset D by the three methods for 50 runs is reported in Table 9.

Table 9: Computing time (seconds) for 50 experiments for PF-SVR, Multi-state Dynamic SVR and FGAPSO-SVR

Dataset	Lengths	PF-SVR	Multi-state Dynamic SVR	FGAPSO-SVR
Dataset 1	350	4983.2	5172.8	6419.3
Dataset 2	420	5949.2	6217.9	7238.4

From above experiments, we can conclude that the proposed Multi-state dynamic SVR method can effectively estimate the current system state and obtain superior predictions. Specifically, the significances of our method that is illustrated by the experiments results lie in: 1) it novelly predict the system reliabilities under the multi-state operating condition, 2) a concise but effective way to estimate the system state is provided, 3) the prediction results is more stable when system runs under a normal state, and more precise when system experience a state transition, 4) the computational cost of the proposed approach is considerable.

5. CONCLUSION

In this paper, a data-driven approach that relies on time series expanded regression is proposed for solving online reliability prediction tasks. Unlike a large number of existing methods which have thoroughly investigated the classic static prognostics problems, this paper looks at systems/components under varying operating conditions, where the effect of operating conditions on the evolution of reliability shall be taken into consideration. Especially, a typical multi-state varying operating condition is emphasized. Thus, the proposed approach is designed with the capabilities of both recognizing the operating state and dynamically updating the regression model.

The proposed approach is implemented by a recursive sequential SVR framework with off-line stage and online stage. First, a novel “optimal hyper-parameters” feature is proposed to characterize the evolution trends of system reliability under different states. Then, the PCC method is chosen to classify the training features off-line and then recognize the test features online. Third, the recognition result of current feature, namely the state estimation of current system, is utilized to recursively update the regression/prediction model built by SVR through a novel Replacement Operation.

Experiments based on an artificial dataset and real case tests on Li-ion battery and more data sets show the superior prediction performance of the proposed method compared with the basic PF-SVR method and the FGAPSO-SVR method. With our proposed

approach, the operating condition states are characterized and estimated by a concise and effective way. The state estimation result is then used for constructing a dynamic prediction model, which consequently leads to a better prediction model. The superiority of the obtained prediction model is illustrated by the prediction results that: more stable when system runs under a normal state, and more precise when system experience a state transition. Besides, the total computing time experiments show that the computational complexity of the proposed method is almost equivalent to PF-SVR, and lower than FGAPSO-SVR.

Since the objective systems faced by reliability engineering community is getting more and more complex, more attentions shall be paid to prognostics under the varying operating condition in the future development. For this, two critical tasks remain to be settled are: 1) how to model the reliability evolution under the varying operating condition and, 2) how to predict the system fail with the modeled system reliability. Therefore, our ambitions in further researches are to, first, enhance the proposed multi-state modeling framework for better universal performance; second, investigate the failure threshold definition under the varying operating condition; and third, develop a possibilistic health management strategy under the varying operating condition.

ACKNOWLEDGEMENT

This research is supported by the Academic Excellence Foundation of BUAA for PhD Students.

REFERENCES

1. Son, J., et al., *Remaining useful life prediction based on noisy condition monitoring signals using constrained Kalman filter*. Reliability Engineering & System Safety, 2016. **152**: p. 38-50.
2. Bevilacqua, M. and M. Braglia, *The analytic hierarchy process applied to maintenance strategy selection*. Reliability Engineering & System Safety, 2000. **70**(1): p. 71-83.
3. Khashei, M. and M. Bijari, *An artificial neural network (p, d, q) model for timeseries forecasting*. Expert Systems with Applications, 2010. **37**(1): p. 479-489.
4. Lin, Y.H., Y.F. Li, and E. Zio, *Fuzzy Reliability Assessment of Systems With Multiple-Dependent Competing Degradation Processes*. IEEE Transactions on Fuzzy Systems, 2015. **23**(5): p. 1428-1438.
5. Moura, M.d.C., et al., *Failure and reliability prediction by support vector machines regression of time series data*. Reliability Engineering & System Safety, 2011. **96**(11): p. 1527-1534.
6. Sapankevych, N. and R. Sankar, *Time series prediction using support vector machines: a survey*. Computational Intelligence Magazine, IEEE, 2009. **4**(2): p. 24-38.
7. Shin, J., et al., *Development of a cyber security risk model using Bayesian networks*. Reliability Engineering & System Safety, 2015. **134**(134): p. 208-217.
8. Tian, L. and A. Noore, *Evolutionary neural network modeling for software cumulative failure time prediction*. Reliability Engineering & System Safety, 2005. **87**(1): p. 45-51.
9. Bian, L., N. Gebraeel, and J.P. Kharoufeh, *Degradation modeling for real-time estimation of residual lifetimes in dynamic environments*. Iie Transactions, 2015. **47**(5): p. 471-486.
10. Liu, Y., et al., *Dynamic Reliability Assessment for Multi-State Systems Utilizing System-Level Inspection Data*. IEEE Transactions on Reliability, 2015. **64**(4): p. 1287-1299.
11. Ding, Y. and A. Lisnianski, *Fuzzy universal generating functions for multi-state system reliability assessment*. Fuzzy Sets and Systems, 2008. **159**(3): p. 307-324.
12. Dong, M. and D. He, *A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology*. Mechanical Systems and Signal Processing, 2007. **21**(5): p. 2248-2266.
13. Ghasemi, A., S. Yacout, and M.S. Quall, *Evaluating the Reliability Function and the Mean Residual Life for Equipment With Unobservable States*. IEEE Transactions on Reliability, 2010. **59**(1): p. 45-54.
14. Hu, et al., *A degradation path-dependent approach for remaining useful life;estimation with an exact and closed-form solution*. European Journal of Operational Research, 2013. **226**(1): p. 53-66.
15. Costello, J.J., G.M. West, and S.D. McArthur, *Machine learning model for event-based prognostics in gas circulator condition monitoring*. IEEE Transactions on Reliability, 2017. **66**(4): p. 1048-1057.
16. Yang, C., et al., *Machine learning-based methods for TTF estimation with application to APU prognostics*. Applied Intelligence, 2017. **46**(1): p. 227-239.
17. Elforjani, M. and S. Shanbr, *Prognosis of Bearing Acoustic Emission Signals Using Supervised Machine Learning*. IEEE Transactions on Industrial Electronics, 2018. **65**(7): p. 5864-5871.
18. Carden, E.P. and P. Fanning, *Vibration Based Condition Monitoring: A Review*. Structural Health Monitoring, 2004. **3**(4): p. 355-377.
19. Valdez, F., P. Melin, and O. Castillo, *Evolutionary method combining particle swarm optimization and genetic algorithms using fuzzy logic for decision making*. in *Fuzzy Systems, 2009. FUZZ-IEEE 2009. IEEE International Conference on*. 2009.
20. Vapnik, V., *The Nature of Statistical Learning Theory*. 2013: Springer New York.
21. Arulampalam, S., et al., *A tutorial on particle filters for on-line non-linear/non-gaussian*. IEEE Trans. Signal Processing, 2001: p. 100-107.
22. Cherkassky, V. and Y. Ma, *Practical selection of SVM parameters and noise estimation for SVM regression*. Neural networks, 2004. **17**(1): p. 113-126.
23. Bo, Y., G.J. Klir, and J.F. Swan-Stone, *Evolutionary fuzzy c-means clustering algorithm*. in *Fuzzy Systems, 1995. International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium., Proceedings of 1995 IEEE Int*. 1995.
24. de Oliveira, J.V. and W. Pedrycz, *Advances in fuzzy clustering and its applications*. 2007: Wiley Online Library.
25. Yang, M.S., *A survey of fuzzy clustering*. Mathematical and Computer Modelling, 1993. **18**(11): p. 1-16.
26. Podofillini, L., et al., *Dynamic safety assessment: Scenario identification via a possibilistic clustering approach*. Reliability Engineering & System Safety, 2010. **95**(5): p. 534-549.
27. Bezdek, J.C., *Pattern recognition with fuzzy objective function algorithms*. 2013: Springer Science & Business Media.
28. Krishnapuram, R. and J.M. Keller, *A possibilistic approach to clustering*. Fuzzy Systems, IEEE Transactions on, 1993. **1**(2): p. 98-110.
29. Dunn, J.C., *A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters*. Journal of Cybernetics, 1973. **3**(3): p. 32-57.

30. Long, B., et al., *An improved autoregressive model by particle swarm optimization for prognostics of lithium-ion batteries*. Microelectronics Reliability, 2013. **53**(6): p. 821-831.
31. Xu, X., Z. Li, and N. Chen, *A Hierarchical Model for Lithium-Ion Battery Degradation Prediction*. IEEE Transactions on Reliability, 2016. **65**(1): p. 310-325.
32. Xu, X. and N. Chen, *A state-space-based prognostics model for lithium-ion battery degradation*. Reliability Engineering & System Safety, 2017. **159**: p. 47-57.
33. Garche, J. and A. Jossen. *Battery management systems (BMS) for increasing battery life time*. in *Telecommunications Energy Special Conference, 2000. TELESCON 2000. The Third International*. 2000.
34. Kaiser, R., *Optimized battery-management system to improve storage lifetime in renewable energy systems*. Journal of Power Sources, 2007. **168**(1): p. 58-65.
35. Christensen, J. and J. Newman, *Stress generation and fracture in lithium insertion materials*. Journal of Solid State Electrochemistry, 2006. **10**(5): p. 293-319.
36. Hendricks, C., et al., *A failure modes, mechanisms, and effects analysis (FMMEA) of lithium-ion batteries*. Journal of Power Sources, 2015. **297**: p. 113-120.
37. Vetter, J., et al., *Ageing mechanisms in lithium-ion batteries*. Journal of Power Sources, 2005. **147**(1-2): p. 269-281.
38. Saxena, S., C. Hendricks, and M. Pecht, *Cycle life testing and modeling of graphite/LiCoO₂ cells under different state of charge ranges*. Journal of Power Sources, 2016. **327**: p. 394-400.