Applied Soft Computing 106 (2021) 107314

Contents lists available at ScienceDirect

Applied Soft Computing

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

Functional deep echo state network improved by a bi-level optimization approach for multivariate time series classification

Zhaoke Huang [a,b], Chunhua Yang [a,*], Xiaofang Chen [a], Xiaojun Zhou [a,d], Guo Chen [a], Tingwen Huang [c], Weihua Gui [a]

[a] School of Automation, Central South University, Changsha 410083, China
[b] Peng Cheng Laboratory, Shenzhen 518000, China
[c] Texas A&M University at Qatar, Doha 23874, Qatar
[d] State Key Laboratory of Synthetical Automation, Shenyang 110819, China

ARTICLE INFO

Article history:
Received 14 August 2020
Received in revised form 20 December 2020
Accepted 8 March 2021
Available online 24 March 2021

Keywords:
Multivariate time series classification
Functional deep echo state network
Bi-level optimization
State transition algorithm
Aluminum electrolysis

ABSTRACT

The multivariate time series (MTS) classification is one of the major tasks of time series data mining. Many methods have been proposed to investigate the MTS classification. Among them, the method based on feature representation is the most popular and widely used one. However, there exist some shortcomings for this method, such as unsatisfactory accuracy, being sensitive to noise and not able to fully make use of time series data attributes. In order to overcome these disadvantages, we propose a new method called functional deep echo state network (FDESN) for MTS classification that utilizes two special operators: temporal aggregation and spatial aggregation. In general, the parameters of the FDESN are determined by random selection, human experience or trial and error. This may increase the complexity of the FDESN or reduce the accuracy of the FDESN. In this study, a novel bi-level optimization approach is proposed to optimize the parameters of the FDESN. The parameter selection problem in the FDESN is transformed into the bi-level optimization problem. The state transition algorithm (STA) is used to solve the bi-level optimization problem. Finally, the experimental results show that the proposed method is superior to other methods. In addition, the proposed method is successfully applied to anode condition identification in aluminum electrolysis. For the aluminum electrolysis datasets, the proposed method improved the average classification accuracy by about 3.5% compared with the other methods. For a specific aluminum electrolysis dataset ACS2504, the classification accuracy significantly increased from 77.92% to 82.69% by using the proposed method.

1. Introduction

Time series exist widely in all aspects of human activities, such as meteorological data recorded in different years and months, various economic indicators recorded in the economic field, and system operation data recorded in the industrial production process. The data sequence obtained from the continuous measurement of a certain variable of the system over a period of time is the time series. If you measure a single variable of the system, you will get a univariate time series (UTS). If you measure multiple variables of the system, you will get a multivariate time series (MTS). A time series contains the operation information of the system within a certain period of time. Obviously, compared to UTS, MTS contains more information, so this kind of data is more worthy of study. MTS classification is one of the major tasks of time series data mining and one of the hottest issues in recent years [1]. However, there are two main difficulties in dealing with multivariate time series classification: (1) the MTS sample contains complex dynamic features that are difficult to represent; (2) the relative importance of subsequences in the MTS sample are different. Hence, it is difficult to deal with it using traditional machine learning algorithms.

In the last two decades, a lot of research have been carried out and many methods have been proposed for MTS classification. Most of these methods can be described as the methods based on feature representation [2–4]. The main characteristic of these methods is that some important features extracted from the original MTS samples are used to train the classifier instead of the samples. These methods can handle curse of dimensionality well by reducing the dimensionality of the samples. Górecki et al. [5] proposed a new approach for MTS classification using a parametric derivative dynamic time warping distance. Baydogan et al. [6] provided a novel classifier based on a new symbolic representation for MTS with several important elements. Mei et al. [7] proposed a Mahalanobis distance-based dynamic time

* Corresponding author.
E-mail address: ychh@csu.edu.cn (C. Yang).

https://doi.org/10.1016/j.asoc.2021.107314
1568-4946/

warping measure for MTS classification. Wang et al. [8] proposed a new approach for MTS classification that utilizes recurrent neural network and adaptive differential evolution algorithm. However, there exist some shortcomings for these methods: (1) unsatisfactory accuracy; (2) time consuming; (3) sensitivity to noise; (4) insufficient use of time series data attributes.

In recent years, deep learning is a hot field in machine learning research and has been successfully applied to many fields, such as image recognition, document classification and speech recognition [9–11]. It gradually transforms the initial low-level feature representation into high-level feature representation through multi-level processing and complex learning tasks such as classification. The potential ability of deep learning models to classify time series especially MTS has gradually attracted the interest of the machine learning community [12]. Indeed, the ability of deep learning to capture the dynamic features and temporal structure of MTS is really strong. Many deep learning algorithms have been proposed for MTS classification, such as echo state network (ESN), convolutional neural network (CNN) and multi layer perceptron (MLP). Among them, the ESN is a relatively recent type of recurrent neural networks (RNNs). In general, an ESN consists of an input layer, a hidden layer (i.e. the reservoir), and an output layer. The core of the ESN is that it is a sparsely connected random RNN. That is, the hidden layer is initialized randomly and constitutes the reservoir. The ESN can mitigate the challenges of RNNs by eliminating the need to compute the gradient for the hidden layer which reduces the training time of these neural networks thus avoiding the vanishing gradient problem. Due to the faster training speed and stronger non-linear approximation capacity, the ESN is a promising method for MTS classification [13–18].

In order to enhance the ability of feature representation, the deep echo state network (DESN) was recently proposed in [19], which has multiple hidden layers. The performance of DESN has been demonstrated in several actual applications. Being composed of a stack of multiple non-linear reservoir layers, the DESN potentially allows to exploit the advantages of a hierarchical temporal feature representation at different levels of abstraction, at the same time preserving the training efficiency typical of the reservoir computing methodology. In other words, the ability of the DESN to represent dynamical features at multiple levels of abstraction allows to capture more naturally the temporal structure of the data whenever it is intrinsically characterized by a multiple time-scales organization. Specifically, the DESN has shown to outperform the shallow ESN for time series classification [20–23]. Gallicchio et al. [22] provided a novel approach to the architectural design of DESN using signal frequency analysis. Xu et al. [24] proposed a novel method based on the wavelet-denoising algorithm and DESN to improve the prediction accuracy of noisy multivariate time series. Sun et al. [25] proposed a deep belief echo state network (DBESN) for time series prediction. In addition, some DESN variants have been proposed in recent years. Long et al. [26] proposed evolving deep echo state networks for intelligent fault diagnosis. Li et al. [27] introduced an approach to pre-train a growing ESN with multiple sub-reservoirs by optimizing singular values, based on particle swarm optimization and singular value decomposition. Ma et al. [28] proposed a novel multiple projection-encoding hierarchical reservoir computing framework called deep projection-encoding echo state network. Although the DESN has a very good ability to capture the dynamic features of time series, it do not consider the relative importance of temporal data at different time steps.

In this paper, by introducing two special operators: temporal aggregation and spatial aggregation, a novel DESN method called functional deep echo state network (FDESN) is proposed for MTS classification, which is inspired by [29]. Temporal aggregation mainly accumulates the information from time-varying input signals in the time domain and characterizes the temporal information of input signals with dynamic weighting functions, while spatial aggregation aggregates the information from the inputs, that are time independent, and obtains joint information from multiple independent inputs with static weights. The combination of temporal aggregation and spatial aggregation can effectively accumulate temporal information in the time dimension that consider the relative importance of temporal data at different time steps. The main idea of the FDESN is to map the information (the neural states and the output weight matrix) of the last reservoir from the functional space to the real number space. This mapping makes the FDESN become a true classifier.

The challenge faced by the FDESN is usually related to the design of its architecture and weights. Specifically, extraction of compact, equal or extended new data features is related to both the DESN's complexity and accuracy. Sometimes, a compact representation may lead to loss of some details about the original data. Also, an extended representation may engender a complexity increase. Many other scenarios are also possible. In a word, the performance of the FDESN is limited by the settings of its architecture and weights. However, due to the lack of understanding the property of the hidden layer, the parameters of the FDESN are difficult to set. In general, the parameters are set manually by trial and error and there are no precise rules to choose these parameters. Hence, the parameters of the FDESN should be optimized through other techniques to improve its performance. Evolutionary algorithms, one of typical representative of metaheuristics, are inspired from biological evolution facts and has proven its ability in finding optimal solutions in complex optimization problems, especially in the parameter optimization problem. Hence, evolutionary algorithms can used to optimize the parameters of the FDESN.

At present, many evolutionary algorithms have been developed for complex optimization problems, which include genetic algorithm (GA), particle swarm optimization (PSO), differential evolution (DE) and so on. Recently, a novel nature-inspired method called state transition algorithm (STA) has emerged in evolutionary algorithms [30]. The powerful global search ability and flexibility of the STA have been demonstrated in many real-world applications [31–35]. The main advantages of the STA over other EAs are threefold. First, the STA has some control parameters and the range of candidate solutions it generates is manageable. In addition, the STA has a simple structure, which is effective, fast and capable of addressing complex optimization problems and also easily suitable for different numerical optimization problems. Finally, the STA's strategies including generating trial solutions, controlling the search direction and range, and the secondary transition mechanism give it very powerful exploration and exploitation capabilities. Thus, the STA could be an effective and efficient method for optimizing the parameters of the FDESN.

In order to handle the above problem, a novel bi-level optimization approach is proposed to optimize the parameters of the FDESN. First, the parameter selection problem in the FDESN is transformed into the bi-level optimization problem. In the bi-level optimization problem, the upper level is to optimize the architecture-related parameters, and the lower level is to optimize the weight-related parameters. Meanwhile, the objective of the upper level is to minimize the computational complexity while maintaining a better classification accuracy, and the objective of the lower level is to maximize the classification accuracy. Then the STA is introduced to find the optimal parameters, which has successfully solved a class of bi-level optimization problem [32]. Finally, in order to verify the validity of the proposed method, some comparative experiments are carried out. The experimental results show that the proposed method outperforms

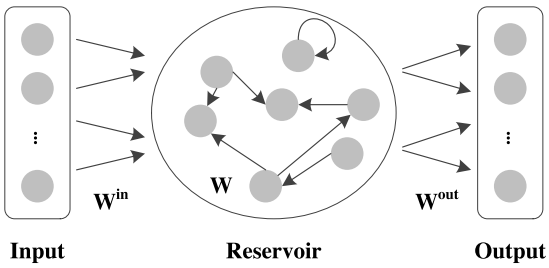

Fig. 1. Echo state network model.

other methods. In addition, the proposed method is successfully applied to anode condition identification in the aluminum electrolysis. The novelty and main contributions of this study are highlighted as follows:

- By introducing two special operators: temporal aggregation and spatial aggregation, a novel DESN method called functional deep echo state network (FDESN) is proposed for MTS classification, which can not only capture the dynamic features of the MTS, but also consider the relative importance of temporal data at different time steps.
- A novel bi-level optimization approach is proposed to optimize the parameters of the FDESN. In the upper level, the architecture-related parameters are optimized by providing a trade-off between the accuracy and complexity. In the lower level, the optimized architecture-related parameters are used to optimize the weight-related parameters by maximizing the classification accuracy.
- The STA is used to solve the bi-level optimization problem. The upper level optimization problem is handled by the multi-objective STA (MOSTA), and the lower level optimization problem is addressed by a single-objective STA.
- The proposed method is successfully applied to anode condition identification in the aluminum electrolysis.

The remainder of this paper is organized as follows. Section 2 introduces deep echo state network, temporal and spatial aggregation operators. Section 3 presents the proposed method. Section 4 provides the experimental results and analysis. Finally, conclusion is given in Section 5.

2. Background

2.1. Deep echo state network (DESN)

The echo state network (ESN) is a novel recurrent neural network (RNN), which uses a sparsely connected structure also called "reservoir" to form the hidden layer [13]. The ESN's architecture is shown in Fig. 1. The characteristics of the ESN are as follows: (1) It contains a relatively large number of neurons; (2) Connections between neurons are randomly generated; (3) The links between neurons are sparse. An ESN consists of an input layer, a hidden layer (i.e. the reservoir), and an output layer. The dynamics of the ESN are defined as follows:

$$\begin{aligned}\mathbf{x}(t+1) &= f(\mathbf{W}^{in}\mathbf{u}(t+1) + \mathbf{W}\mathbf{x}(t)) \\ \mathbf{y}(t+1) &= g(\mathbf{W}^{out}\mathbf{x}(t+1))\end{aligned} \tag{1}$$

where $\mathbf{u}$, $\mathbf{x}$ and $\mathbf{y}$ are the inputs, the internal states and the outputs, respectively; $\mathbf{W}^{in}$, $\mathbf{W}$ and $\mathbf{W}^{out}$ represent the input-to-reservoir weight matrix, the reservoir weight matrix and the reservoir-to-output weight matrix, respectively; f and g are the activation functions of the reservoir and output units, respectively.

In the training phase, the matrices $\mathbf{W}^{in}$ and $\mathbf{W}$ are initialized randomly. The inputs are projected into the high-dimensional state spaces in the reservoir, and the matrix $\mathbf{W}^{out}$ can be learned by linear regression. Thus, training an ESN is both simple and fast, and the ESN can avoid the vanishing gradient problem and reduce computational complexity for modeling time series data.

Recently, in order to improve the feature representation ability of ESN, multi-layered echo state network, also known as deep echo state network (DESN), was proposed by Malik et al. [19], which has multiple reservoirs. The reservoirs in the DESN are serially connected. Hence, each reservoir state relies mainly on its previous state and the output of its previous reservoir. The DESN's architecture is shown in Fig. 2. The dynamics of DESN are defined as follows:

$$\begin{aligned}&\mathbf{x}^1(t+1) = f(\mathbf{W}^{in}\mathbf{u}(t+1) + \mathbf{W}^1\mathbf{x}^1(t)) \\ &\mathbf{x}^2(t+1) = f(\mathbf{W}^{exter(1)}\mathbf{x}^1(t+1) + \mathbf{W}^2\mathbf{x}^2(t)) \\ &\cdots \\ &\mathbf{x}^M(t+1) = f(\mathbf{W}^{exter(M-1)}\mathbf{x}^{M-1}(t+1) \\ &\qquad + \mathbf{W}^M\mathbf{x}^M(t))\end{aligned} \tag{2}$$

The output of DESN can be computed as follows:

$$\mathbf{y}(t+1) = g(\mathbf{W}^{out}\mathbf{x}^M(t+1)) \tag{3}$$

where M is the number of the reservoirs; $\mathbf{u}$, $\mathbf{x}$, $\mathbf{y}$ denote the inputs, the internal states and the outputs, respectively; $\mathbf{W}^{in}$ and $\mathbf{W}^{out}$ represent the input weight matrix and the output weight matrix, respectively; $\mathbf{W}^i$ $(i = 1, \ldots, M)$ and $\mathbf{W}^{exter(j)}$ $(j = 1, \ldots, M-1)$ stand for the inner recurrent matrix of each reservoir and the external weight matrix for the jth reservoir and the $(j+1)$th reservoir, respectively; f and g are the activation functions of the reservoirs and the output, respectively. In this study, the *tanh* function is used as the activation functions of the reservoirs in the DESN.

2.2. Temporal and spatial aggregation operators

Temporal and spatial aggregation operators were first proposed by He and Xu [36], which basically mimic the operating mechanism of the biological brain. The brain can be seen as a large network of many neurons. In a brain, each neuron can receive and process biochemical signals according to the signal time. Between neurons, biochemical signals can be transmitted from multiple neurons to others. Temporal aggregation and spatial aggregation are a simulation of these two processes. The structures of these two operators are shown in Fig. 3. The temporal aggregation operator is to accumulate the dynamic weight information from an input signal and map it to a real number output. The output of temporal aggregation operator can be described as follows:

$$y = \int_{t=0}^{T} w(t)u(t)dt \tag{4}$$

where $w(t)$ is the weight function, $u(t)$ is the input signal and y is the output; T represents the length of the input signal. The spatial aggregation operator is to obtain the information from multiple input signals and output a real number. The output of spatial aggregation operator can be described as follows:

$$y = f(\sum_{i=1}^{N} w_i u_i - \theta) \tag{5}$$

where w_i is a static weight, N is the number of the input signals, θ is a threshold and f is the activation function.

In order to obtain the relative importance of temporal data at different time steps, the temporal aggregation and spatial aggregation need to be added to the DESN in this study. They can project the temporal signals into a real number and make the DESN become a true classifier.

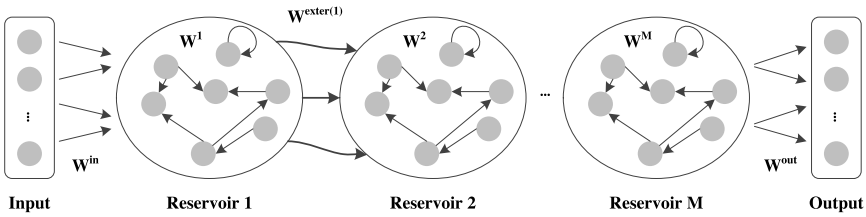

Fig. 2. Deep echo state network model.

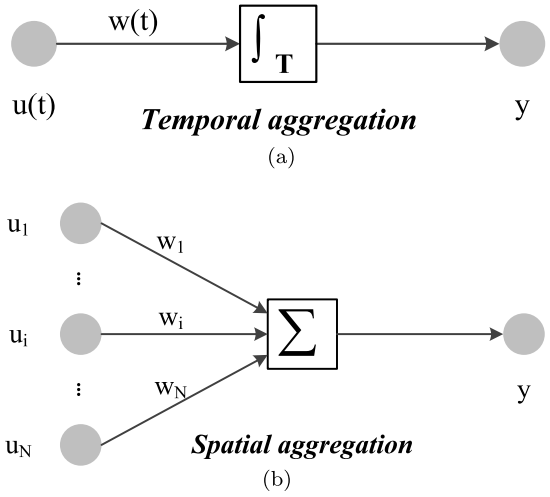

Fig. 3. (a) The temporal aggregation operator. (b) The spatial aggregation operator.

3. The proposed method

In this section, a novel DESN method called functional deep echo state network (FDESN) is proposed for MTS classification. Then the crucial parameters of FDESN are introduced. Finally, a novel bi-level optimization approach is proposed to optimize the parameters of the FDESN.

3.1. Functional deep echo state network (FDESN)

The FDESN has four layers: input layer, reservoir layer, spatio-temporal aggregation layer, and output layer. The structure of the FDESN is shown in Fig. 4. The difference between FDESN and DESN is that FDESN has the spatio-temporal aggregation layer. The basic idea behind FDESN is to map the information (the neural states and the output weight matrix) of the last reservoir from the functional space into the real number space using the temporal aggregation operator, and then forms the discriminating hyperplane using the spatial aggregation operator. These operations make the FDESN become a true classifier.

In the FDESN, the neural states are updated as the same as Eq. (2). We use the following equation to represent the updating rules:

$$\begin{aligned} &F : \mathbb{R}^{N_U} \times \underbrace{\mathbb{R}^{N_R} \times \cdots \times \mathbb{R}^{N_R}}_{M} \rightarrow \underbrace{\mathbb{R}^{N_R} \times \cdots \times \mathbb{R}^{N_R}}_{M} \\ &\mathbf{x}(t) = F(\mathbf{u}(t), \mathbf{x}(t-1)) \end{aligned} \tag{6}$$

where F is a function to update the global state of the FDESN. N_U is the size of the input. N_R is the size of the reservoir.

The details of the spatio-temporal aggregation layer are presented as follows. According to Eq. (4), the temporal aggregation result of the ith unit in the last reservoir is

$$\eta_{ji} = \int_{t=0}^{T} w_{ji}(t) x_i(t) dt \tag{7}$$

where $x_i(t)$ is the neural state of the ith unit in the last reservoir at time t and $w_{ji}(t)$ is the weight from the ith unit to the jth output. According to Eq. (5), the spatial aggregation of the jth output is shown as follows:

$$\Sigma_j = \sum_{i=1}^{N} \eta_{ji} \tag{8}$$

Thus, the jth output is

$$y_j = g(\Sigma_j) = g(\sum_{i=1}^{N} \int_{t=0}^{T} w_{ji}(t) x_i(t) dt) \tag{9}$$

Let N denote the number of neurons in the last reservoir, L denote the number of classes. The output weight matrix of the last reservoir at time t is

$$\mathbf{W}^{out}(t) = \begin{bmatrix} w_{11}(t) & w_{12}(t) & \cdots & w_{1N}(t) \\ w_{21}(t) & w_{22}(t) & \cdots & w_{2N}(t) \\ \vdots & \vdots & \ddots & \vdots \\ w_{L1}(t) & w_{L2}(t) & \cdots & w_{LN}(t) \end{bmatrix} \tag{10}$$

The neural states in the last reservoir at time t is denoted as

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) & \cdots & x_N(t) \end{bmatrix}^T \tag{11}$$

Thus, the output could be written as

$$\mathbf{y} = g(\int_{t=0}^{T} \mathbf{W}^{out}(t) \mathbf{x}(t) dt) \tag{12}$$

From Eq. (12), the output weight matrix of the last reservoir $\mathbf{W}^{out}(t)$ is a function of time, which represents the discriminating hyperplane. The goal of FDESN is to find a proper $\mathbf{W}^{out}(t)$ to classify the samples. The detailed method for learning the $\mathbf{W}^{out}(t)$ can be seen in [29].

However, the FDESN has many parameters and the performance of the FDESN is closely related to these parameters. Hence, the crucial parameters of FDESN are introduced in the following subsection.

3.2. The crucial parameters of FDESN

The parameters of the FDESN can be divided into two classes due to the different properties of the parameters. The first class is related to the architecture and the second class is related to the weights. The first class of parameters includes the number of the reservoirs M, the size of the ith reservoir S_{R_i}, input connectivity rate C_{in}, the ith reservoir's internal connectivity rate C_{R_i}, the ith reservoir's external connectivity rate $C_{exter(i)}$ and the ith reservoir's spectral radius ρ_i. The second class of parameters includes the input weight matrix $\mathbf{W}^{in}$, the inner recurrent matrix of the ith reservoir $\mathbf{W}^i$ and the external weight matrix for the ith reservoir and the $(i+1)$th reservoir $\mathbf{W}^{exter(i)}$. The function of the first class of parameters is to determine the architecture of the FDESN and the range of the second class of parameters. For example, if the size of the first reservoir S_{R_1} is 100, the inner recurrent matrix of the first reservoir W^1 is a 100-by-100 matrix. Hence, the first class

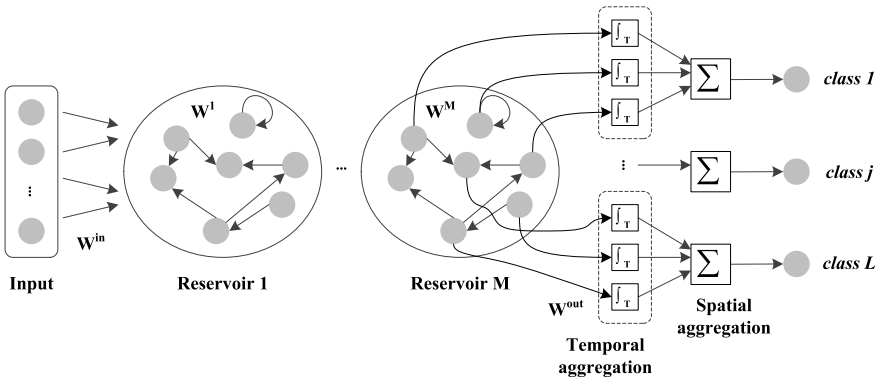

Fig. 4. Functional deep echo state network model.

of parameters has important and quantitative influence on the second class of parameters, and further affects the performance of the FDESN. The second class of parameters directly affects the performance of the FDESN. The details of these parameters are presented as follows.

The reservoir layer is an important part of the FDESN, which is used as the information processing medium to map the input signals from the low-dimensional input space to the high-dimensional state space. Then the nonlinear relationship between the input layer and the output layer can be transformed into a linear relationship between the reservoir layer and the output layer. M is the number of the reservoirs in the reservoir layer. S_{R_i} is the number of neurons contained in the ith reservoir. The bigger M can provide better expression for data description meanwhile needs more computation. The bigger S_{R_i} can describe the dynamic evolution mechanism of the system while too big one will cause the problem of over-fitting. Hence, M and S_{R_i} are the most important parameters affecting the performance of the FDESN.

The connectivity rate represents the sparse degree of the connections among the neurons. In FDESN, there are three connectivity rates: C_{in} is the input connectivity rate, C_{R_i} is the ith reservoir's internal connectivity rate and $C_{exter(i)}$ is the ith reservoir's external connectivity rate. The more connectivity rate, The more connections among the neurons. As Jaeger said, the dynamic characteristics of ESN can be satisfied with about 10% connections. But this is not mandatory and can be tailored to specific issues. In this study, the connectivity rate is among the range of 5–10%.

The spectral radius ρ_i refers to the maximum absolute value of the eigenvalue of the internal connection matrix of the ith reservoir. Since the reservoir is a recursive neural network, it is inevitable to consider the stability problem. Jaeger pointed out that the stable operation of ESN can be guaranteed when the spectral radius is between [0, 1]. But it depends on the specific problems. In this study, ρ_i is also among the range of [0, 1].

Besides, $\mathbf{W}^{in}$, $\mathbf{W}^{i}$ and $\mathbf{W}^{exter(i)}$ in the original FDESN are randomly generated and unchanged until the end of the training process. In order to improve the accuracy of the FDESN for MTS classification, these three weight matrices will be optimized as the lower level parameters.

3.3. A bi-level optimization approach applied to FDESN

In general, the first class of parameters of the FDESN are determined by human experience or trial and error. Although it can achieve good results, it is difficult to choose the best parameters for complex tasks. On the other hand, $\mathbf{W}^{in}$, $\mathbf{W}^{i}$ and $\mathbf{W}^{exter(i)}$ in the original FDESN are randomly generated and unchanged until the end of the training process. This will have an impact on the training results, causing the performance of FDESN to degrade in many applications. Hence, a novel bi-level optimization approach is proposed to optimize the parameters of the FDESN. The general scheme of it is shown in Fig. 5. The details of this approach are presented as follows.

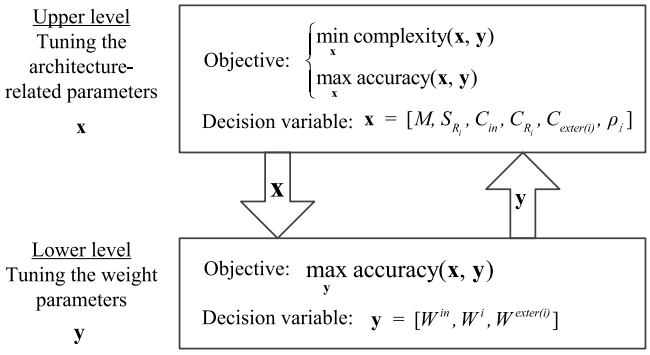

Fig. 5. The general scheme of the proposed bi-level optimization approach.

3.3.1. Upper level: architecture optimization

In the upper level, the main purpose is to tune the architecture-related parameters in the FDESN. So the decision variables are the architecture-related parameters, which are denoted by a vector $\mathbf{x}$. Based on the analysis in the previous subsection, these parameters affect both computational efficiency and computational performance. Hence, different from the traditional parameter selection, a multi-objective approach is adopted to consider the contradiction between computational efficiency and computational performance. Two objectives are considered, including the efficiency objective to minimize the computational complexity and the performance objective to maximize the classification accuracy. Thus, the architecture-related parameters are optimized in a reasonable manner, considering the computational complexity and the classification accuracy at the same time.

The efficiency fitness function consists of the number of the reservoir's internal connections in all the reservoirs and the number of the reservoir's external connections in the reservoir layer, shown as Eq. (13).

$$complexity = \sum_{i=1}^{M} S_{R_i} C_{R_i} + \sum_{i=1}^{M-1} S_{R_i} C_{exter(i)} \tag{13}$$

The performance fitness function is to directly compute the classification accuracy of the FDESN, shown as Eq. (14).

$$accuracy = \frac{number\ of\ correct\ classifications}{total\ number\ of\ classifications} \tag{14}$$

It divides into two stages: (1) train the FDESN using the training data; (2) obtain a classification accuracy value by using the FDESN to classify the testing dataset. As can be seen, the upper level optimization problem is a multi-objective optimization problem. Its solution is no longer an optimal solution, but a non-dominant solution set.

3.3.2. Lower level: weights optimization

In the lower level, the main purpose is to tune the weight-related parameters in the FDESN. So the decision variables are the weight-related parameters, which are denoted by a vector **y**. The objective is to maximize the classification accuracy. Hence, Eq. (14) is selected as the fitness function.

Different from the general bi-level optimization problem, **x** in the lower level optimization problem is in the non-dominated solution set, which is obtained from the upper level. Hence, the global optimal solution is determined by comparing the fitness function in the lower level.

3.4. The implementation of the proposed method

Algorithm 1 The pseudo-code of STA-FDESN

Require: Time series dataset D, the parameters of STA and MOSTA;
1: Set the parameters of STA and MOSTA;
2: Generate the initial candidate solutions of the upper level;
3: Randomly initialize the weights for the FDESN;
4: Apply the MOSTA to solve the upper level optimization problem;
5: Obtain the upper level initial non-dominated solutions **xBests**;
6: **for** each non-dominated solution **do**
7: Apply the STA to solve the lower level optimization problem;
8: Obtain the lower level best solution $\mathbf{yBest}_i$ for the ith non-dominated solution;
9: Save the $\mathbf{yBest}_i$;
10: **end for**
11: Get the initial best solution $\mathbf{Z} = (\mathbf{xBest}, \mathbf{yBest})$ by comparing the $\mathbf{yBest}_i$;
12: **while** the specified termination criterion is not met **do**
13: Update the upper level current best non-dominated solutions **xBests** based on the **yBest** in the **Z** by applying the MOSTA to solve the upper level optimization problem;
14: Obtain the lower level best solution $\mathbf{yBest}_i$ for the ith non-dominated solution according to the above for loop (lines 6–10);
15: Update the current best solution $\mathbf{Z} = (\mathbf{xBest}, \mathbf{yBest})$;
16: **end while**
17: **return** $\mathbf{Z} = (\mathbf{xBest}, \mathbf{yBest})$.

The proposed optimization model is a bi-level, asymmetric and nonlinear optimization problem. Because of the powerful global search ability and flexibility of state transition algorithm (STA) [32–35], the STA is used to solve the bi-level optimization problem. In order to meet the needs of the upper level optimization problem, multi-objective STA (MOSTA) is adopted to solve it. MOSTA is a Pareto-based multi-objective optimization algorithm, which is developed by Zhou et al. [31]. For the lower level optimization problem, the single objective STA is adopted to solve it. The implementation of the proposed method is presented as follows.

The proposed method is also called "STA-FDESN", and its flowchart is shown in Fig. 6. Meanwhile, Algorithm 1 shows the details of the STA-FDESN. First, the time series dataset and the important related parameters are entered into the algorithm (line 1). Then, the initial candidate solutions of the upper level are randomly generated and random weights are initialized for the FDESN (lines 2–3). After that, the set of initial non-dominated solutions **xBests** are obtained by using the MOSTA to solve the upper level optimization problem (lines 4–5). Then, each non-dominated solution of the upper level is sent to the lower level. For each non-dominated solution, the current best solution $\mathbf{yBest}_i$ of the lower level is obtained by executing the STA to optimize the lower level optimization problem (lines 6–10). Finally, the initial best solution $\mathbf{Z} = (\mathbf{xBest}, \mathbf{yBest})$ is obtained by comparing the $\mathbf{yBest}_i$ according to the lower fitness function (line 11). Afterwards, the same steps are repeated until the specified termination criterion is met (lines 11–16). It is worth noting that the lower level optimization is equivalent to a subroutine, which is to obtain the best solution **yBest** for the input non-dominated solutions **xBests**. Once the upper level obtains the current best non-dominant solutions **xBests**, the lower level optimization is called to obtain the best solution **yBest** based on the **xBests**, and finally **Z** is obtained by comparing the lower level fitness function. The process of the upper lower optimization is based on the current best solution **yBest** in the current best solution **Z**. Hence, the best non-dominant solutions of the upper level optimization problem are generated based on the **yBest**, and the best solution of the lower level optimization problem is generated based on the **xBests**. They are the results of the alternating iterative optimization.

Table 1
Description of the MTS datasets in the benchmark experiments.

Datasets	Variables	Length	Classes	Samples
Arabic digits	13	4∼93	10	8800
Australian language	22	45∼136	95	2565
Character trajectories	3	109∼205	20	2858
CMU subject 16	62	127∼580	2	58
ECG	2	39∼152	2	200
Japanese vowels	12	7∼29	9	640
Libras	2	45	15	360
Pen digits	2	8	10	10992
Robot failure LP1	6	15	4	88
Robot failure LP2	6	15	5	47
Robot failure LP3	6	15	4	47
Robot failure LP4	6	15	3	117
Robot failure LP5	6	15	5	164
Wafer	6	104∼198	2	1194

4. Experimental results and analysis

In this section, some benchmarks are first used to evaluate the performance of the proposed method STA-FDESN, which were obtained from UCI [37] and UCR [38]. Then the STA-FDESN is applied to identify the anode condition in aluminum electrolysis by classifying a multivariate time series data named anode current signals. The nine state-of-the-art algorithms: dynamic time warping (DTW) [5], derivative dynamic time warping (DDTW) [5], parametric derivative dynamic time warping DD_{DTW} [5], Conceptor-ADE (C_{ADE}) [8], COTE [39], CNN [40], DESN, FESN [29] and FDESN were selected to compare with the STA-FDESN. All experiments were executed on a personal computer with Intel Core i7 and 16-GB RAM using MATLAB.

4.1. Benchmark test

In the first experiment, fourteen real-world datasets were used to show the effectiveness of the STA-FDESN. Table 1 shows the detailed information of these MTS datasets. The names of the datasets are listed in the first column. The number of variables in

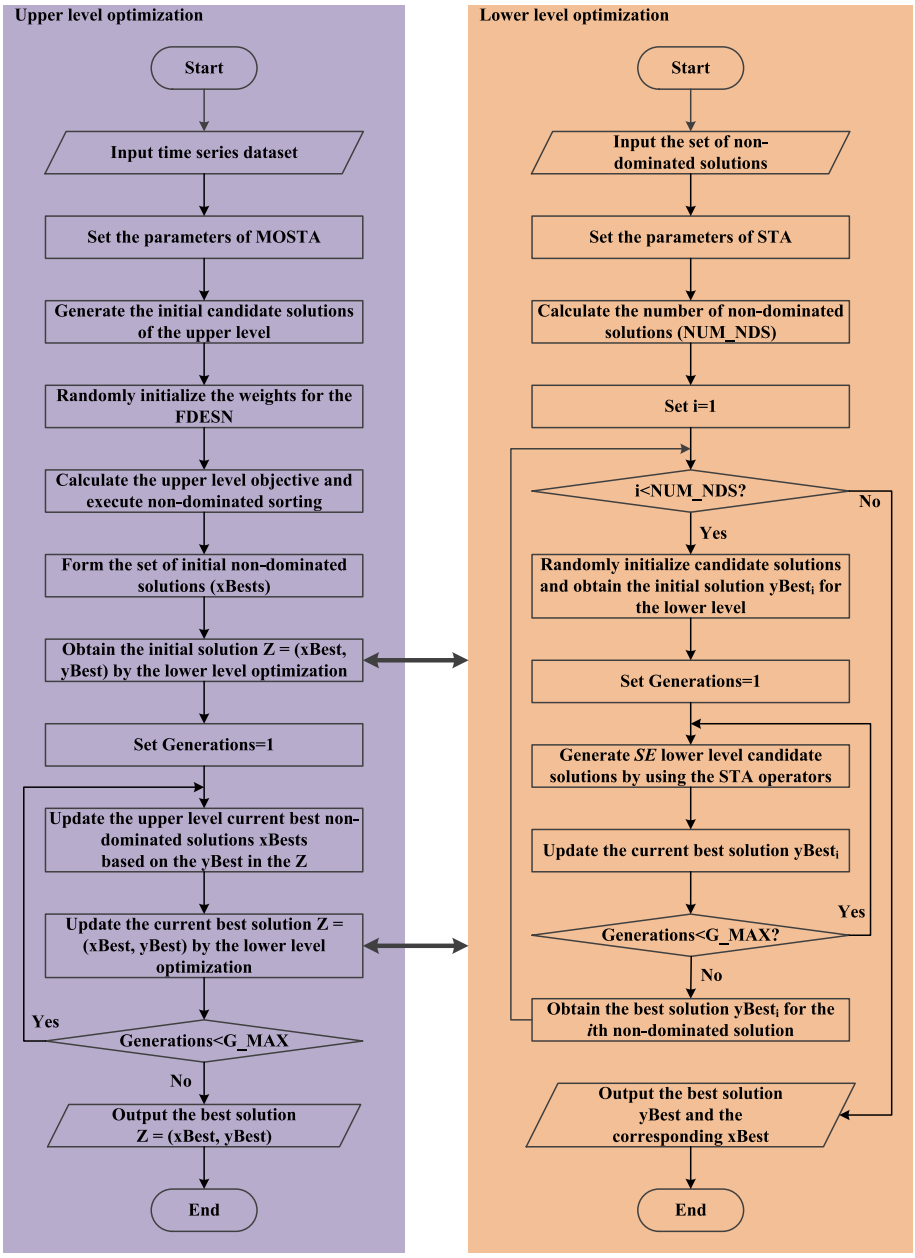

Fig. 6. The flowchart of the proposed method.

these datasets ranges from 2 to 62. The length of MTS in each dataset is different. The number of classes and samples range from 2 to 95 and from 47 to 10992, respectively.

In order to investigate the influence of key parameters to the proposed model's performance, a series of experiments are designed and conducted. Tables 2–4 show the average classification accuracy (%) of the proposed method under different circumstances. Table 2 shows the results of the FDESN's performance comparisons under different numbers of the reservoirs. It is obvious that the performance of the FDESN varies greatly with different reservoirs. However, too many reservoirs of the proposed model can easily lead to over-fitting. Table 3 shows the results of the FDESN's performance comparisons under different numbers of neurons. It is clear that the more neurons the proposed model had, the higher the accuracy. But too many neurons lead to increase the computational complexity. That is why we design a complexity & accuracy objective in the upper optimization problem. Table 4 shows the results of the STA-FDESN's performance comparisons under multiple random initializations. It is observed that the proposed method is stable in the experiment.

Table 2
Comparison of the FDESN's performance under different numbers of the reservoirs.

Datasets	The number of the reservoirs				
	1	5	10	15	20
Arabic digits	97.16	**99.05**	98.65	98.90	98.81
Australian language	72.44	78.95	**83.24**	81.60	82.46
Character trajectories	96.08	**98.36**	97.83	97.10	97.66
ECG	85.52	86.56	**89.52**	88.06	88.04
Japanese vowels	96.10	**98.91**	97.66	97.81	97.19
Pen digits	96.92	97.67	**99.10**	98.15	98.02

Next, the proposed method STA-FDESN is compared with nine state-of-the-art algorithms, including DTW, DDTW, DD_{DTW}, C_{ADE},

Table 3
Comparison of the FDESN's performance under different numbers of neurons.

Datasets	The size in each reservoir				
	100	200	300	400	500
Arabic digits	97.45	97.68	98.23	98.90	**99.10**
Australian language	76.18	79.88	81.72	83.63	**83.66**
Character trajectories	96.36	97.17	98.01	98.22	**98.46**
ECG	86.04	87.06	88.04	88.56	**89.52**
Japanese vowels	96.41	97.35	97.81	98.13	**98.60**
Pen digits	97.10	97.75	97.94	98.18	**98.90**

Table 4
Comparison of the STA-FDESN's performance under multiple random initializations.

Datasets	Random initialization				
	1	2	3	4	5
Arabic digits	99.52	99.48	99.36	99.45	99.54
Australian language	85.85	84.60	84.87	84.87	85.46
Character trajectories	98.92	98.78	98.46	98.99	98.64
ECG	90.52	89.52	89.04	90.52	89.52
Japanese vowels	99.07	98.91	99.07	98.44	98.75
Pen digits	99.38	99.21	99.42	99.35	99.18

Table 5
Cross validation classification accuracy (%).

Datasets	DTW	DDTW	DD_{DTW}	C_{ADE}	COTE
Arabic digits	**99.81**	89.50	**99.81**	99.23	98.45
Australian language	81.95	72.67	81.52	74.98	80.35
Character trajectories	98.64	98.22	**99.09**	98.06	98.78
CMU subject 16	96.33	89.33	96.33	99.60	93.57
ECG	81.50	86.00	85.50	88.80	85.52
Japanese vowels	97.97	60.94	97.97	99.00	95.16
Libras	91.39	95.83	95.00	95.84	90.83
Pen digits	99.35	99.39	**99.50**	98.44	99.25
Robot failure LP1	87.36	77.50	85.14	**99.25**	88.61
Robot failure LP2	68.00	62.00	68.00	75.82	64.50
Robot failure LP3	71.00	71.00	75.00	75.82	70.83
Robot failure LP4	89.92	79.55	89.92	96.17	85.61
Robot failure LP5	70.70	62.68	71.25	72.65	65.88
Wafer	97.99	90.79	98.08	98.26	97.57
Datasets	CNN	DESN	FESN	FDESN	STA-FDESN
Arabic digits	98.56	99.16	99.23	98.50	99.52
Australian language	72.83	78.48	81.44	79.53	**85.89**
Character trajectories	98.25	98.11	98.18	98.08	98.95
CMU subject 16	93.81	95.00	96.90	95.24	**100.00**
ECG	86.04	86.04	86.54	87.06	**90.52**
Japanese vowels	97.66	98.13	98.44	97.98	**99.07**
Libras	93.61	94.72	95.56	93.90	**96.67**
Pen digits	99.10	98.90	98.55	98.00	99.40
Robot failure LP1	95.67	96.89	98.00	96.78	98.89
Robot failure LP2	70.67	72.67	74.67	70.83	**76.67**
Robot failure LP3	70.83	72.83	74.67	72.50	**76.67**
Robot failure LP4	90.61	94.17	95.83	93.26	**96.67**
Robot failure LP5	70.77	71.39	72.61	70.75	**73.84**
Wafer	98.07	98.16	98.07	97.99	**99.33**

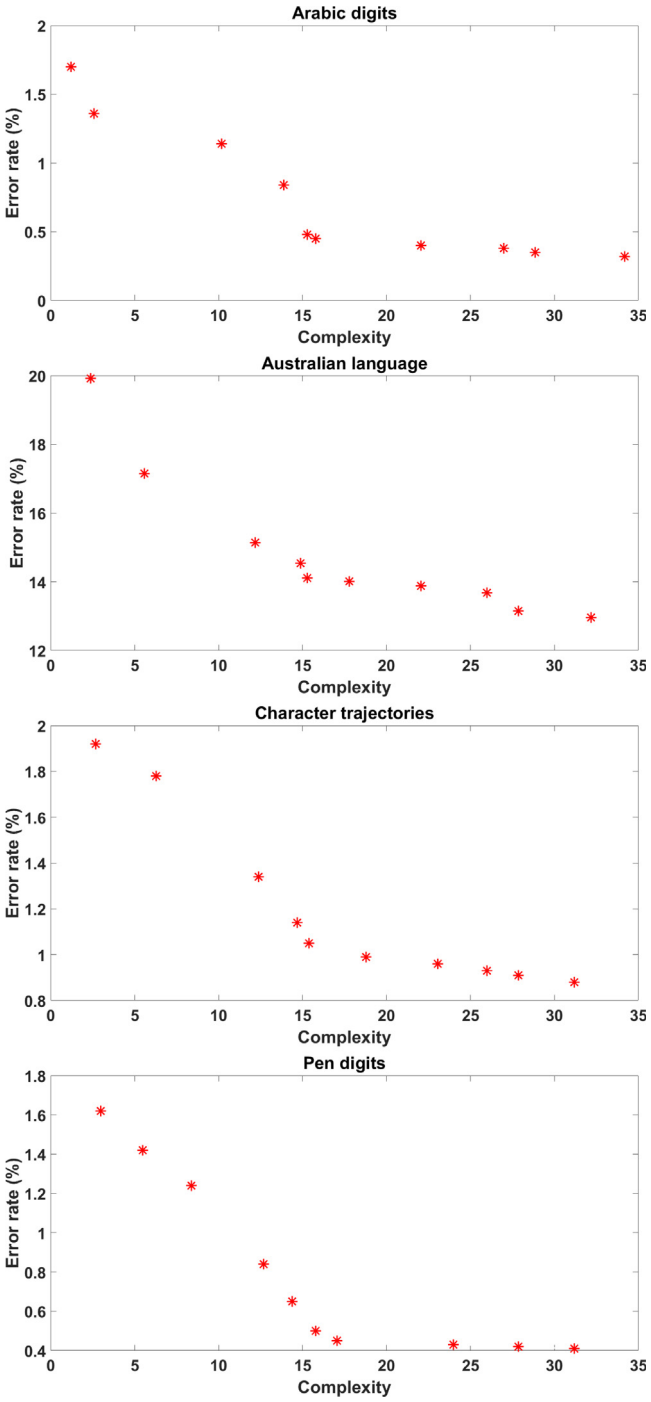

Fig. 7. Upper level: the non-dominant solutions obtained by the MOSTA for the four datasets.

COTE, CNN, DESN, FESN and FDESN. In order to make a fair comparison for the experimental datasets, the related parameters of the different methods are set according to the suggestions of their corresponding literatures. It is worth noting that the parameters of the FDESN are not set optimally. For each dataset, the 10-fold cross-validation method is used to evaluate the performance of the comparative methods.

Table 5 shows the results of the comparative experiments. It is clear that the STA-FDESN outperforms other methods on most datasets. This shows the superiority of the STA-FDESN. Some phenomena can be concluded from Table 5. First, the performance of the proposed method is better than that of C_{ADE} and FESN. The reason is simple: because the STA-FDESN has multiple reservoirs, the feature representation capability of the STA-FDESN is more powerful than that of C_{ADE} and FESN. Second, the performance of the proposed method is better than that of COTE, CNN and DESN. Because the proposed method introduced the spatio-temporal aggregation layer, which can consider the relative importance of multivariate time series data. Third, the performance of the proposed method is relatively worse than that of DTW, DDTW and DD_{DTW} on Arabic digits, Character trajectories and Pen digits. The same characteristic is that these datasets has a large number of samples. It is probable that too much data may cause the proposed method to over-fit. Fourth, for the four datasets, that is, Arabic digits, Character trajectories, Pen digits and Robot failure

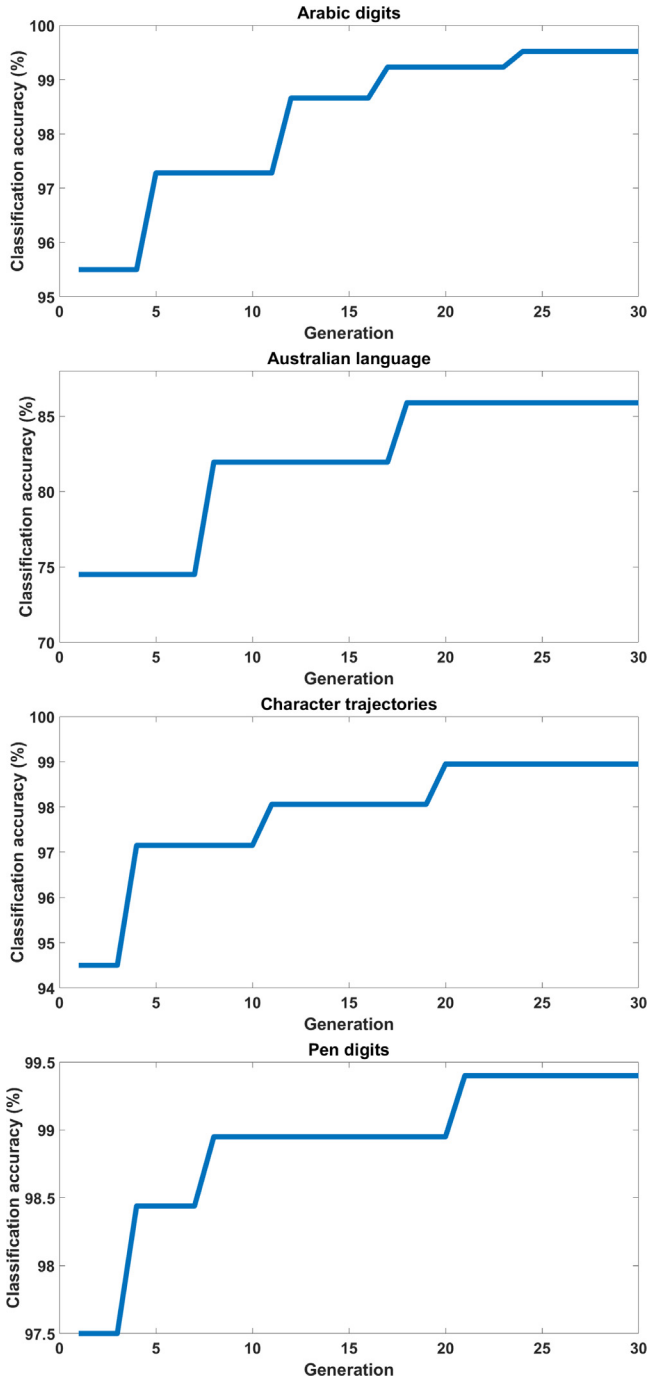

Fig. 8. Iterative curves of the classification accuracy obtained by the proposed method for the four datasets.

LP1, the classification results of the STA-FDESN is basically close to the best results. The non-dominant solutions obtained by the MOSTA for the four datasets are shown in Fig. 7. It is important to note that the y-coordinate here is the error rate. Because maximizing accuracy equals minimizing the error rate. It is clear that the pareto front is obtained by the MOSTA, which is the optimal results of the trade-off between accuracy and complexity. Iterative curves of the classification accuracy obtained by the proposed method for the four datasets are shown in Fig. 8. The optimization process ends when the generation reaches 30. At the same time, the best parameter values of the FDESN are obtained. In addition, the execution time is tracked for some

Table 6
Execution times (in seconds) of the comparative methods on some datasets.

Datasets	DTW	C_{ADE}	COTE	CNN	STA-FDESN
Arabic digits	8804	6818	7210	10618	7018
CMU subject 16	78	61	75	92	68
ECG	386	268	296	401	286
Japanese vowels	521	360	411	684	389

Table 7
Comparison of classification accuracy between several evolutionary algorithms on some datasets.

Datasets	Australian language	ECG	Libras	Pen digits
FDESN	79.53	87.06	93.90	98.00
STA-FDESN	**85.89**	**90.52**	**96.67**	**99.40**
DE-FDESN	83.98	89.04	95.84	99.30
PSO-FDESN	83.90	88.56	95.56	99.20
GA-FDESN	83.47	88.08	95.29	99.10

Table 8
Execution times (in seconds) of the comparative methods on some datasets.

Datasets	Australian language	ECG	Libras	Pen digits
STA-FDESN	3225	286	329	1569
DE-FDESN	3659	325	384	1840
PSO-FDESN	3745	336	391	1886
GA-FDESN	4568	381	426	2001

comparative algorithms and comparative results are shown in Table 6. Although the STA-FDESN takes a little more time than the C_{ADE}, its running time is still acceptable. Hence, in general, the performance of the proposed method is relatively more superior than other methods.

To further evaluate the performance of the STA to optimize the FDESN, a performance comparison between several evolutionary algorithms on some datasets is carried out. The comparative evolutionary algorithms include differential evolution (DE), particle swarm optimization (PSO) and genetic algorithm (GA) [41]. Their related parameters are also set according to the suggestions of their corresponding literatures. Table 7 shows the results of the comparative experiments. It is obvious that the STA produces more effective accuracy results compared to other evolutionary algorithms. Next, the execution time is tracked for all comparative algorithms and the comparative results are shown in Table 8. It can be seen that the all the comparative algorithms take a relative long time to optimize the FDESN, but the STA takes relatively little time.

4.2. Application to anode condition identification in aluminum electrolysis

In aluminum electrolysis, the aluminum reduction cell is a complex system with multivariable, nonlinear, time delay and time variation, and its mechanism model is difficult to describe [35,42–44]. The aluminum reduction cell is usually divided into four parts: cathode structure, upper structure, bus structure and electrical insulation, as shown in Fig. 9. As an important part of the upper structure, anode is one of the most important modules in the aluminum reduction cell, which is known as the "heart" of aluminum electrolysis. Since the anode process is closely related to the quality of aluminum electrolysis production, how to judge the working conditions of anode is very important. Fortunately, with the rapid development of sensor technology, anode current signal appears as a new process variable, which can reflect the working conditions of anode. The anode current signal is a multivariate time series data, which usually has 24 variables in a 400 kA cell. If the working condition of anode changes, the anode current signal changes immediately. Hence, it is a promising

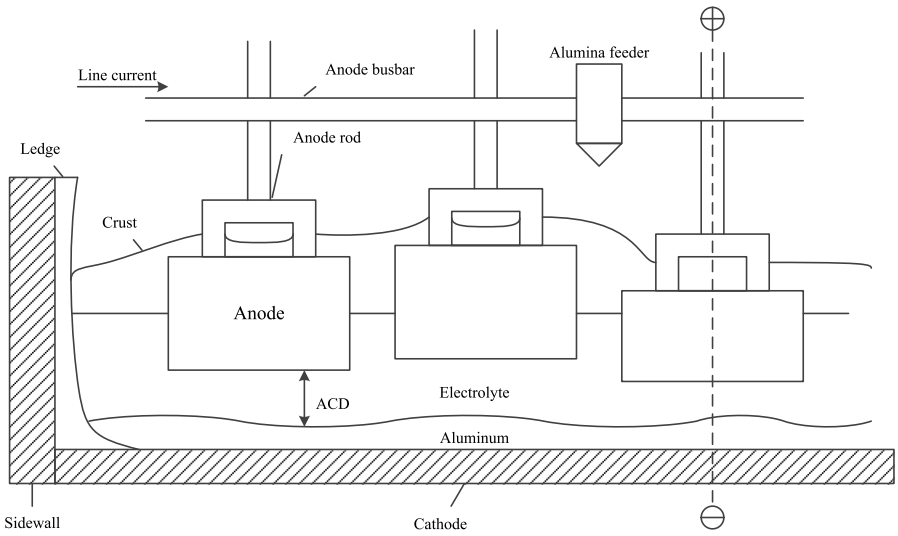

Fig. 9. Sketch map of an aluminum reduction cell.

Table 9
Description of the anode current signals datasets in aluminum electrolysis.

Datasets	Variables	Length	Classes	Samples
CS2501	24	150∼200	4	896
ACS2502	24	150∼200	4	756
ACS2503	24	150∼200	3	698
ACS2504	24	150∼200	2	566

Table 10
Cross validation classification accuracy (%).

Datasets	DTW	DDTW	DD_{DTW}	C_{ADE}	COTE
ACS2501	88.62	79.91	85.38	95.20	90.07
ACS2502	83.60	80.42	77.78	87.96	80.16
ACS2503	77.51	72.78	80.66	82.66	75.50
ACS2504	71.55	73.15	74.03	77.92	72.97
Datasets	CNN	DESN	FESN	FDESN	STA-FDESN
ACS2501	92.19	94.64	91.97	92.97	**96.21**
ACS2502	83.60	86.38	83.47	84.00	**89.02**
ACS2503	80.52	83.10	80.95	82.95	**85.39**
ACS2504	76.50	77.03	76.15	76.85	**82.69**

method to identify the working condition of anode by classifying the anode current signals.

In this subsection, four datasets about anode current signals were used to illustrate the performance of the proposed method in this study. These datasets were gathered from an aluminum electrolysis plant located in Shandong, China. Table 9 shows the detailed information of these datasets. Here, the number of variables is 24, the length of these datasets is from 150 to 200, and the number of classes and samples range from 2 to 4 and from 50 to 200, respectively. In addition, each sample has one output class indicating the working condition of anode (i.e., anode effect, anode slippage, anode deformation and normal anode).

Table 10 shows the results of the comparative experiments. In addition, iterative curves of the classification accuracy obtained by the proposed method for the four datasets are shown in Fig. 10. It is obvious that the STA-FDESN outperforms other methods on all the datasets. Some interesting conclusions can be obtained from Table 10. First, the performance of the proposed method is also better than that of C_{ADE} and FESN. Second, the performance of the proposed method becomes better with the increase of the number of samples. Third, from the relative accuracy rates, we can see that the performance of the STA-FDESN is significantly improved. Hence, in general, the performance of the proposed method is superior than that of other methods in aluminum electrolysis datasets.

4.3. Discussion

As shown by the experimental results reported in this section, the proposed method has a significant impact on the precision of the FDESN model. The bi-level optimization approach has improved the performance of the FDESN considerably in terms of precision, stability and robustness. The advantage of the upper level is that it considers a trade-off between classification and complexity, not a single objective optimization. This not only optimizes the architecture of the FDESN, but also maintains a good precision, while providing diversity for the optimization of the lower level. In addition, compared to the FDESN without optimization, the robustness of the STA-FDESN is obviously enhanced. Meanwhile, It can be seen that the multi-layer architecture has outperformed the single-layer structure in terms of accuracy, in the majority of benchmark datasets. Thus, the STA-FDESN seems to be a very important paradigm for providing effective data representations. Finally, since the proposed approach is bi-level, the computational complexity may increase. The STA is selected for the optimization task as it is simple and fast. Also, the training of the FDESN is very simple and fast as it is based on a non-iterative linear regression method. Overall, the STA-FDESN has demonstrated its utility in the complex tasks.

5. Conclusion

In this study, a novel method called functional deep echo state network (FDESN) was proposed for MTS classification by introducing temporal aggregation and spatial aggregation. Then the parameter selection problem in the FDESN was transformed into the bi-level optimization problem. In the bi-level optimization problem, the upper level is to optimize the architecture-related parameters, and the lower level is to optimize the weight related parameters. Moreover, the STA was introduced to find the best parameter values of the FDESN. The experimental results indicated that the proposed method outperforms the other nine state-of-the-art algorithms. In addition, the proposed method was successfully applied to anode condition identification in aluminum electrolysis. Meanwhile, I think the theoretical analysis of convergence and stability of the STA-FDESN needs to be further

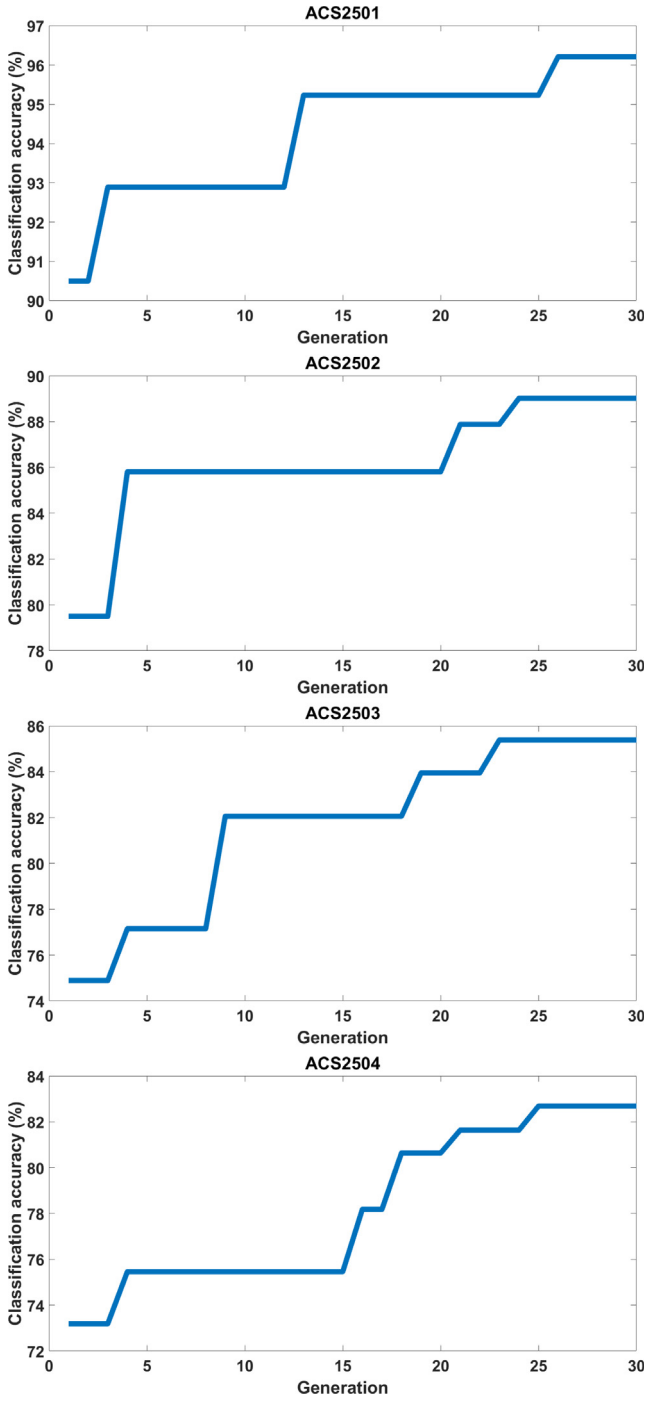

Fig. 10. Iterative curves of the classification accuracy obtained by the proposed method for the four datasets.

strengthened. In the future, the proposed method is expected to extend for the MTS classification problem with noise.

CRediT authorship contribution statement

Zhaoke Huang: Conceptualization, Methodology, Software, Writing - original draft. **Chunhua Yang:** Supervision. **Xiaofang Chen:** Formal analysis, Investigation, Data curation. **Xiaojun Zhou:** Methodology, Visualization, Writing - review & editing. **Guo Chen:** Methodology, Investigation. **Tingwen Huang:** Visualization, Writing - review & editing, Supervision. **Weihua Gui:** Conceptualization, Supervision.

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

The authors thank the National Key Research and Development Program of China (Grant No. 2018YFB1701100), the National Natural Science Foundation of China (Grant Nos. 61873285, 62073344), the International Cooperation and Exchange of the National Natural Science Foundation of China (Grant No. 61860206014), the 111 Project (Grant No. B17048), the Science and Technology Innovation Program of Hunan Province, China (Grant No. 2020RC2008) and the Postdoctoral Science Foundation of Central South University, China for their funding support.

References

[1] Tak Chung Fu, A review on time series data mining, Eng. Appl. Artif. Intell. 24 (1) (2011) 164–181.
[2] Guoliang He, Yong Duan, Rong Peng, Xiaoyuan Jing, Tieyun Qian, Lingling Wang, Early classification on multivariate time series, Neurocomputing 149 (2015) 777–787.
[3] Hailin Li, Accurate and efficient classification based on common principal components analysis for multivariate time series, Neurocomputing 171 (2016) 744–753.
[4] Robert Moskovitch, Yuval Shahar, Classification of multivariate time series via temporal abstraction and time intervals mining, Knowl. Inf. Syst. 45 (1) (2015) 35–74.
[5] Tomasz Górecki, Maciej Łuczak, Multivariate time series classification with parametric derivative dynamic time warping, Expert Syst. Appl. 42 (5) (2015) 2305–2312.
[6] Mustafa Gokce Baydogan, George Runger, Learning a symbolic representation for multivariate time series classification, Data Min. Knowl. Discov. 29 (2) (2015) 400–422.
[7] Jiangyuan Mei, Meizhu Liu, Yuan-Fang Wang, Huijun Gao, Learning a mahalanobis distance-based dynamic time warping measure for multivariate time series classification, IEEE Trans. Cybern. 46 (6) (2015) 1363–1374.
[8] Lin Wang, Zhigang Wang, Shan Liu, An effective multivariate time series classification approach using echo state network and adaptive differential evolution algorithm, Expert Syst. Appl. 43 (2016) 237–249.
[9] Yann LeCun, Yoshua Bengio, Geoffrey Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.
[10] Jürgen Schmidhuber, Deep learning in neural networks: An overview, Neural Netw. 61 (2015) 85–117.
[11] Li Deng, Dong Yu, Deep learning: Methods and applications, Found. Trends® Signal Process. 7 (3–4) (2014) 197–387.
[12] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, Pierre-Alain Muller, Deep learning for time series classification: a review, Data Min. Knowl. Discov. 33 (4) (2019) 917–963.
[13] Herbert Jaeger, The "echo state" approach to analysing and training recurrent neural networks, German National Research Center for Information Technology, 2001.
[14] L. Liu, Y. Liu, S. Tong, Fuzzy-based multierror constraint control for switched nonlinear systems and its applications, IEEE Trans. Fuzzy Syst. 27 (8) (2019) 1519–1531.
[15] Xianshuang Yao, Zhanshan Wang, Broad echo state network for multivariate time series prediction, J. Franklin Inst. B 356 (9) (2019) 4888–4906.
[16] Xianshuang Yao, Zhanshan Wang, Huaguang Zhang, Prediction and identification of discrete-time dynamic nonlinear systems based on adaptive echo state network, Neural Netw. 113 (2019) 11–19.
[17] Zhigang Wang, Yu Rong Zeng, Sirui Wang, Lin Wang, Optimizing echo state network with backtracking search optimization algorithm for time series forecasting, Eng. Appl. Artif. Intell. 81 (2019) 117–132.
[18] Junxiu Liu, Tiening Sun, Yuling Luo, Su Yang, Yi Cao, Jia Zhai, An echo state network architecture based on quantum logic gate and its optimization, Neurocomputing 371 (2020) 100–107.
[19] Zeeshan Khawar Malik, Amir Hussain, Qingming Jonathan Wu, Multilayered echo state machine: A novel architecture and algorithm, IEEE Trans. Cybern. 47 (4) (2017) 946–959.
[20] Claudio Gallicchio, Alessio Micheli, Luca Pedrelli, Deep reservoir computing: a critical experimental analysis, Neurocomputing 268 (2017) 87–99.

[21] Claudio Gallicchio, Alessio Micheli, Echo state property of deep reservoir computing networks, Cogn. Comput. 9 (3) (2017) 337–350.
[22] Claudio Gallicchio, Alessio Micheli, Luca Pedrelli, Design of deep echo state networks, Neural Netw. 108 (2018) 33–47.
[23] Shaohui Zhang, Zhenzhong Sun, Man Wang, Jianyu Long, Yun Bai, Chuan Li, Deep fuzzy echo state networks for machinery fault diagnosis, IEEE Trans. Fuzzy Syst. 28 (7) (2020) 1205–1218.
[24] Meiling Xu, Min Han, Hongfei Lin, Wavelet-denoising multiple echo state networks for multivariate time series prediction, Inform. Sci. 465 (2018) 439–458.
[25] Xiaochuan Sun, Tao Li, Qun Li, Yue Huang, Yingqi Li, Deep belief echo-state network and its application to time series prediction, Knowl.-Based Syst. 130 (2017) 17–29.
[26] Jianyu Long, Shaohui Zhang, Chuan Li, Evolving deep echo state networks for intelligent fault diagnosis, IEEE Trans. Ind. Inf. 16 (7) (2019) 4928–4937.
[27] Ying Li, Fanjun Li, PSO-based growing echo state network, Appl. Soft Comput. 85 (2019) 105774.
[28] Qianli Ma, Lifeng Shen, Garrison W. Cottrell, DeePr-ESN: A deep projection-encoding echo-state network, Inform. Sci. 511 (2020) 152–171.
[29] Qianli Ma, Lifeng Shen, Weibiao Chen, Jiabin Wang, Jia Wei, Zhiwen Yu, Functional echo state network for time series classification, Inform. Sci. 373 (2016) 1–20.
[30] Xiaojun Zhou, Chunhua Yang, Weihua Gui, State transition algorithm, J. Ind. Manag. Optim. 8 (4) (2012) 1039–1056.
[31] X. Zhou, J. Zhou, C. Yang, W. Gui, Set-point tracking and multi-objective optimization-based PID control for the goethite process, IEEE Access 6 (2018) 36683–36698.
[32] Zhaoke Huang, Chunhua Yang, Xiaojun Zhou, Weihua Gui, A novel cognitively inspired state transition algorithm for solving the linear bi-level programming problem, Cogn. Comput. 10 (5) (2018) 816–826.
[33] Z. Huang, C. Yang, X. Zhou, T. Huang, A hybrid feature selection method based on binary state transition algorithm and reliefF, IEEE J. Biomed. Health Inf. 23 (5) (2019) 1888–1898.
[34] Zhaoke Huang, Chunhua Yang, Xiaojun Zhou, Shengxiang Yang, Energy consumption forecasting for the nonferrous metallurgy industry using hybrid support vector regression with an adaptive state transition algorithm, Cogn. Comput. 12 (2019) 357–368.
[35] Zhaoke Huang, Chunhua Yang, Xiaofang Chen, Keke Huang, Yongfang Xie, Adaptive over-sampling method for classification with application to imbalanced datasets in aluminum electrolysis, Neural Comput. Appl. 32 (2020) 7183–7199.
[36] Xingui He, Shaohua Xu, Process Neural Networks: Theory and Applications, Springer Science & Business Media, 2010.
[37] Dheeru Dua, Casey Graff, UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, 2017.
[38] Hoang Anh Dau, Eamonn Keogh, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, Yanping, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, Gustavo Batista, The UCR time series classification archive, 2018, https://www.cs.ucr.edu/~eamonn/time_series_data_2018/, October.
[39] Anthony Bagnall, Jason Lines, Jon Hills, Aaron Bostrom, Time-series classification with COTE: the collective of transformation-based ensembles, IEEE Trans. Knowl. Data Eng. 27 (9) (2015) 2522–2535.
[40] Bendong Zhao, Huanzhang Lu, Shangfeng Chen, Junliang Liu, Dongya Wu, Convolutional neural networks for time series classification, J. Syst. Eng. Electron. 28 (1) (2017) 162–169.
[41] Hamid Afshari, Warren Hare, Solomon Tesfamariam, Constrained multi-objective optimization algorithms: Review and comparison with application in reinforced concrete structures, Appl. Soft Comput. 83 (2019) 105631.
[42] Weichao Yue, Xiaofang Chen, Weihua Gui, Yongfang Xie, Hongliang Zhang, A knowledge reasoning Fuzzy-Bayesian network for root cause analysis of abnormal aluminum electrolysis cell condition, Front. Chem. Sci. Eng. 11 (3) (2017) 414–428.
[43] Weichao Yue, Weihua Gui, Xiaofang Chen, Zhaohui Zeng, Yongfang Xie, Knowledge representation and reasoning using self-learning interval type-2 fuzzy Petri nets and extended TOPSIS, Int. J. Mach. Learn. Cybern. 43 (2019) 3499–3520.
[44] Weichao Yue, Weihua Gui, Xiaofang Chen, Zhaohui Zeng, Yongfang Xie, Evaluation strategy and mass balance for making decision about the amount of aluminum fluoride addition based on superheat degree, J. Ind. Manag. Optim. 13 (5) (2017) 130–147.