IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 21, 2021, accepted April 19, 2021, date of publication April 26, 2021, date of current version May 12, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3075554

Dual Extreme Learning Machine Based Online Spatiotemporal Modeling With Adaptive Forgetting Factor

KANGKANG XU[1], XI TAN[1], BI FAN[2], TENGFEI XIAO[3], XI JIN[1], AND CHENGJIU ZHU[1]

[1]School of Electromechanical Engineering, Guangdong University of Technology, Guangzhou 510006, China
[2]Department of Management Science, Shenzhen University, Shenzhen 518060, China
[3]School of Intelligent System Engineering, Sun Yat-Sen University, Guangzhou 510006, China

Corresponding author: Chengjiu Zhu (dave_email@foxmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 51905109, Grant 71701136, and Grant 61703444; and in part by the Natural Science Foundation of Guangdong Province under Grant 2021A1515011971.

ABSTRACT Many industrial thermal processes are large-scale time-varying nonlinear distributed parameter systems (DPSs). To effectively model such systems, dual extreme learning machine based online spatiotemporal modeling with adaptive forgetting factor (AFFD-ELM) is proposed in this paper. This method can recursively update the parameters of the low-order temporal model by using newly arriving data under Karhunen- Loève (KL) based space/time separation. In this way, the time-varying dynamics can be tracked real-time very well as output data increases over time. Besides, since the training samples are usually timeliness, adaptive forgetting factor (AFF) is also embedded in this method to improve the online learning effects by adding a reasonable weight to previous data. This online learning strategy makes the process promising for online modeling under continuously samples environment. The proposed method is utilized for online temperature prediction of the curing oven. Simulation results verify the efficiency and viability of the online spatiotemporal model.

INDEX TERMS Dual extreme learning machine, forgetting mechanism, online sequential learning algorithm, online spatiotemporal modeling.

I. INTRODUCTION

Distributed parameter systems (DPSs) widely exist in the industrial processes [1]–[4]. Unlike lumped parameter systems (LPSs), DPSs are often described by one or a set of partial differential equations (PDEs) with corresponding initial and boundary conditions [5]. Their input, output, and even state parameters are space/time coupled, which leads to the infinite-dimensional characteristic of such systems [6], [7]. Modeling such systems is difficult but essential for process prediction, control, and optimization [8]–[12].

There are many spatiotemporal modeling methods researched in recent years [13]–[16]. Among them, space/time separation-based modeling methods have been widely used and proven viable and practical to model DPSs. The main idea of space/time separation methods is to suppose that the spatiotemporal variables can be decomposed into a series of dominant spatial basis functions (BFs) and corresponding time coefficients according to Fourier transform. Thus, the BFs and time coefficients can be learned separately using physical-based or data-based algorithms.

For the BFs learning, if the PDEs description of systems is known, the typically used method is the spectral method [17], which can acquire representative spatial BFs for space/time separation and synthesis [18]. However, this method often requires the systems to have a regular space domain and homogeneous boundary conditions. In many industrial processes, their accurate PDEs description is often entirely unknown for the lack of enough physical information and uncertainty. Under this situation, data-based algorithms are very suitable for the optimal learning of BFs. Karhunen–Loève (KL) method, as known as principal component analysis (PCA) or proper orthogonal decomposition (POD) [19]–[22], has been popularly used in many types of research related to spatiotemporal modeling and turned out to be an effective method for BFs learning.

The associate editor coordinating the review of this manuscript and approving it for publication was Pengcheng Liu.

Once the BFs are learned adequately, the corresponding time coefficients should be modeled to approximate the temporal dynamics of DPSs. If the PDEs description of systems is known, Galerkin's method [23], [24] can be used to reduce the infinite-dimensional PDEs into finite-dimensional ordinary differential equations (ODEs) with learned spatial BFs. Thus, the unknown parameters or structures that existed in ODEs can be identified and approximated using conventional LPSs methods. If the PDEs description of systems in completely anonymous, traditional modeling methods, such as support vector machines (SVM) [25], neural networks (NN) [26], block-oriented models [27], and fuzzy model [28] can be applied to model the time coefficients.

From the above literature review, it can be found that traditional spatiotemporal modeling methods are mainly developed in an offline environment. That is, all the training data is collected and ready before the modeling process. However, in many real industrial processes, systems often work with large-scale time-varying features, which require the spatiotemporal model online updated using newly coming samples for preserving satisfactory performance. Therefore, a novel online spatiotemporal modeling method should be developed for such complex DPSs.

In recent years, there are few studies reported devoting to the online spatiotemporal modeling. Li and Qi [29] proposed an incremental modeling method for DPSs, which refers to adding the hierarchical spatiotemporal kernels incremental algorithm for online learning. Jiang *et al.* [30] proposed a precision online spatiotemporal model to predict the thermal crown in hot rolling processes, where a hybrid intelligent model given in state-space formulation is applied for online learning implementation. However, this method required the mathematical structure of the systems known. Lu *et al.* [31] proposed an adaptive spatiotemporal modeling method for time-varying DPSs with the application of an adaptive Takagi-Sugeno fuzzy model. Wang and Li [32] proposed an incremental KL method for online spatiotemporal modeling of DPSs. This method calculates the newly BFs with previous learned BFs and newly arriving spatiotemporal samples. The above mentioned online spatiotemporal modeling methods can achieve satisfactory model performance. However, they didn't consider the inherent system structure features. Though the online spatiotemporal model proposed by Jiang *et al.* [30] is developed based on the PDEs of the hot rolling processes, the application of NN leads this online model to low computational efficiency. Furthermore, all the above methods cannot reflect the timeliness of online sequential training data well.

In this paper, a novel dual extreme learning machine based online spatiotemporal modeling method with adaptive forgetting factor (AFFD-ELM) is proposed for large-scale time-varying DPSs. Firstly, a traditional KL based spatiotemporal model is developed under the offline environment. Since two coupled nonlinearities are embedded in the general distributed thermal systems [33], the low-order temporal model should be designed according to such structure characteristics. Recent years have witnessed an increasing interest in the topic of extreme learning machine (ELM) [34]–[37], which has the advantage of universal approximation capability and fast learning speed. Therefore, ELM is applied here to approximate the two coupled nonlinear functions, where the derived model is called dual ELM (D-ELM) in this paper. Since the model structure of D-ELM matches well with the systems, it can achieve better performance [37]. Secondly, an online sequential learning algorithm (OSLA) [36] will be designed for online updating of the spatiotemporal model. This algorithm can update the model parameters with previously calculated parameters and newly arriving samples. Since the training samples usually have timeliness [38], that is, they have a specific validity period, an online sequential learning algorithm with a forgetting mechanism (FOS-ELM) [39] is developed. However, this method straightforward abandons the oldest data during online learning and ignores their real contribution in the current learning situation. Though timeliness online sequential extreme learning machine (TOS-ELM) [38] can achieve satisfactory model performance with adaptive weight scheme, the contribution of each data set cannot be adaptive adjust with different online learning environment once it was determined. Therefore, adaptive forgetting factor (AFF) is designed by using a fast leave-one-out cross-validation (FLOO-CV) method [41] and is embedded into the online learning process. The embedding of AFF can improve the learning effects and reduce the lousy affection of previous data. Thirdly, by space/time synthesis, AFFD-ELM based online spatiotemporal model can be acquired. Finally, experiments on a real industrial thermal process will be carried out to verify the proposed method's effectiveness. The main novelty and contributions of our work can be summarized as below:

1) D-ELM is designed to resolve the inherent coupled structure issue of systems, where the current online spatiotemporal modeling methods fail.

2) Different from the traditional online spatiotemporal methods which ignore the timeliness of sequential training data, online sequential learning algorithm with AFF is developed for the online update of D-ELM-based spatiotemporal model. This novel algorithm can deal with data timeliness well and improve the learning effects during the online updating process.

3) The proposed method does not require the analytical expression of DPSs, which implies that it is very proper for general industrial applications.

The rest of this paper is organized as follows: Section II is the problem description. The detailed description of the proposed online spatiotemporal modeling method is presented in section III. Experimental verification is discussed in section IV, and the conclusions are given in section V.

II. PROBLEM DESCRIPTION

According to the heat transfer laws, the general mathematical description of industrial thermal processes can be expressed

by the following nonlinear PDE [33]:

$$\rho c \frac{\partial T(S,t)}{\partial t} = k \cdot \nabla^2 (T(S,t)) + f(T(S,t)) + \rho Q(S,t), \tag{1}$$

with the Neumann boundary conditions and initial condition as:

$$\frac{\partial T}{\partial S}|_{S=0} = 0, \frac{\partial T}{\partial S}|_{S=S_0} = 0, \quad T(S,0) = T_0(S), \tag{2}$$

where $T(S,t)$ denotes the spatiotemporal output at time t and location $S = (x, y, z) \in (0, S_0)$. c is the specific heat coefficient, correspondingly. $f(T(S,t))$ is the unknown nonlinear thermal dynamics related to the spatiotemporal output T. $Q(S,t)$ is the heating source, which is a nonlinear function of the manipulated input signals $u(t) = [u_1(t), u_2(t), \ldots u_n(t)]^T$. $\nabla^2 = \frac{\partial}{\partial x^2} + \frac{\partial}{\partial y^2} + \frac{\partial}{\partial z^2}$ is the Laplacian space operator. $T_0(S,0)$ is the spatiotemporal initial output. ρ and k are variables with respect to temperature T, their expressions can be written as follows:

$$k = k_0 + \bar{k}(T), \quad \rho = \frac{\rho_0}{1 + \bar{\rho}(T)}, \tag{3}$$

where k_0 and ρ_0 are the parameter initial values, $\bar{k}(T)$ and $\bar{\rho}(T)$ are functions of $T(S,t)$.

Then (1) can be transformed into the following equation:

$$\frac{\partial T}{\partial t} = k_1 \nabla^2 T + F(T) + \frac{1}{c} Q, \tag{4}$$

where: $k_1 = k_0 / \rho_0 c$,

$$F(T) = \frac{k_0 \bar{\rho}(T)}{\rho_0 c} \nabla^2 T + \frac{1 + \bar{\rho}(T)}{\rho_0 c} (\bar{k}(T) \nabla^2 T + \frac{\partial \bar{k}(T)}{\partial x} \frac{\partial T}{\partial x} + \frac{\partial \bar{k}(T)}{\partial y} \frac{\partial T}{\partial y} + \frac{\partial \bar{k}(T)}{\partial z} \frac{\partial T}{\partial z} + f(T)).$$

It can be found that the right side of (4) consists of two nonlinear functions $F(T)$ and Q.

The model (1) described by nonlinear PDE is space/time coupled that cannot directly be used for online prediction and control. The commonly used method is the space/time separation methods [7], [33], by which the spatiotemporal variable $T(S,t)$ can be decoupled into a set of spatial BFs $\{\phi_i(S)\}_{i=1}^{\infty}$ with corresponding time coefficients $\{a_i(t)\}_{i=1}^{\infty}$. Usually, the first *n-th* order of BFs $\{\phi_i(S)\}_{i=1}^{n}$ can capture the dominant dynamics of the DPS. Then, $T(S,t)$ and $Q(S,t)$ can be described as:

$$T(S,t) = \sum_{i=1}^{n} \phi_i(S) a_i(t), \tag{5}$$

$$Q(S,t) = \sum_{i=1}^{n} \phi_i(S) q_i(t). \tag{6}$$

The unit orthogonal BFs can be learned using the KL method with collected spatiotemporal distribution snapshots. Substituting (5) and (6) into (1), the equation residual can be calculated as:

$$R = \frac{\partial T_n}{\partial t} - k_1 \nabla^2 T_n - F(T_n) - \frac{1}{c} Q. \tag{7}$$

With Galerkin's method [23], [24], take the spatial BFs ϕ_j as weight function, the residual (7) can be minimized by solving:

$$(R, \phi_j) = 0, \tag{8}$$

where (R, ϕ_j) denotes the inner product: $\int R \cdot \phi_j d\Omega$.

Substituting (5) and (6) into (8):

$$\int \frac{\partial \left[\sum_{i=1}^{n} \phi_i \cdot a_i(t)\right]}{\partial t} \cdot \phi_j d\Omega = \int k_1 \nabla^2 \left(\sum_{i=1}^{n} \phi_i \cdot a_i(t)\right) \cdot \phi_j d\Omega + \int F(T_n) \cdot \phi_j d\Omega + \int \frac{1}{c} \left(\sum_{i=1}^{n} \phi_i \cdot q_i(t)\right) \cdot \phi_j d\Omega. \tag{9}$$

Since the BFs $\{\phi_i(S)\}_{i=1}^{n}$ are orthonormal, (9) can be calculated as:

$$\dot{a}_i(t) = \sum_{i=1}^{n} k_{ij} a_j(t) + \tilde{F}_i(a_1(t), \ldots, a_n(t)) + \frac{1}{c} q_i(t), \tag{10}$$

where: $k_{ij} = \int (k_1 \nabla^2 \phi_i) \phi_j d\Omega \; (j = 1, 2, \ldots, n)$,

Ω is the domain $(0 \le x \le x_0, 0 \le y \le y_0, 0 \le z \le z_0)$,

$$\tilde{F}_i(a_1(t), \ldots, a_n(t)) = \int F(T_n) \phi_j d\Omega.$$

Ignoring the coupling effect among the low-order models, (10) is simplified as:

$$\dot{a}_i(t) = k_{ii} a_i(t) + \tilde{F}_i(a_i(t)) + \frac{1}{c} q_i(t). \tag{11}$$

The discrete form of (11) can be rewritten as:

$$a_i(t) = \tilde{k}_{ii} a_i(t-1) + \tilde{F}_i(a_i(t-1)) + \frac{\Delta t}{c} q_i(t-1), \tag{12}$$

where $\tilde{k}_{ii} = 1 + \Delta t k_{ii}$, Δt is the discrete interval.

Define the nonlinear functions:

$$g_i(a_i(t)) = \tilde{k}_{ii} a_i(t) + \tilde{F}_i(a_i(t)), f_i(u(t)) = \frac{\Delta t}{c} q_i(t). \tag{13}$$

Then the mathematical structure of the time coefficients can be described as:

$$a_i(t) = g_i(a_i(t-1)) + f_i(u(t-1)). \tag{14}$$

Obviously, model (14) has two coupled nonlinear structure $g_i(\cdot)$ and $f_i(\cdot)$ as depicted in Fig. 1. There are many data-based identification methods to estimate the low-order temporal model (14). While the temporal model is trained very well, the spatiotemporal model can be finally reconstructed by space/time synthesis.

Though traditional spatiotemporal modeling methods have a satisfactory model performance on industrial thermal processes, there are still some problems required to solve for the online implementation of the model, which can be summarized as follows:

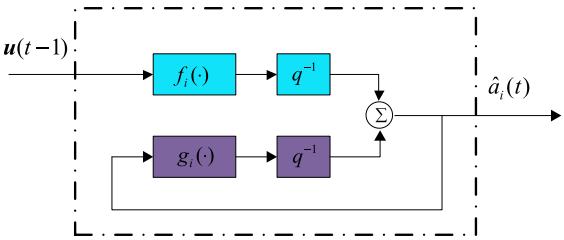

FIGURE 1. Structure of the low-order ODE model.

1) **Strong nonlinear structure**: Two inherent coupled nonlinear dynamics are existed in the process. To approximate the process well, the proposed model required to be designed according to the nonlinear structure characteristics of the system.
2) **Online update**: Traditional spatiotemporal modeling methods are conducted in an offline environment, which often leads to the model drift for large-scale, time-varying systems. Therefore, model parameters online learning strategy with newly arriving snapshots is necessary, which can adapt to real-time dynamics variation.
3) **Computational accuracy**: For the large-scale time-varying systems, there are significant differences between the new and old samples. If the spatiotemporal model only continuously adds new snapshots to the training samples set without any treatment of the ancient samples, it will lead to its limited learning ability to the latest training samples, and thus difficult to accurately depict the time-varying system characteristics.

The research of this paper will focus on solving the above three problems. Detailed methodology is given in the following section.

III. ONLINE SPATIOTEMPORAL MODELING WITH ADAPTIVE FORGETTING FACTOR

A. MODELING FRAMEWORK

The framework of the proposed modeling method is shown in Fig. 2. The modeling issue can be summarized as three aspects: 1) D-ELM-based model design, 2) Online sequential learning algorithm, and 3) Forgetting mechanism.

In the first aspect, the traditional KL method is used to decompose the spatiotemporal variables into a series of dominant spatial BFs and corresponding time coefficients. Subsequently, taking time coefficients as model output, D-ELM is designed to model the temporal dynamics according to model (14).

In the second aspect, two steps are needed for online sequential learning. 1) update of the modeling coefficient $\boldsymbol{\theta}$, which can be expressed as a function of the AFF ω. 2) update of the AFF ω, where a FLOO-CV method is applied to weaken the previously collected samples by giving a reasonable weight.

In the third aspect, spatiotemporal variables can be reconstructed by space/time synthesis.

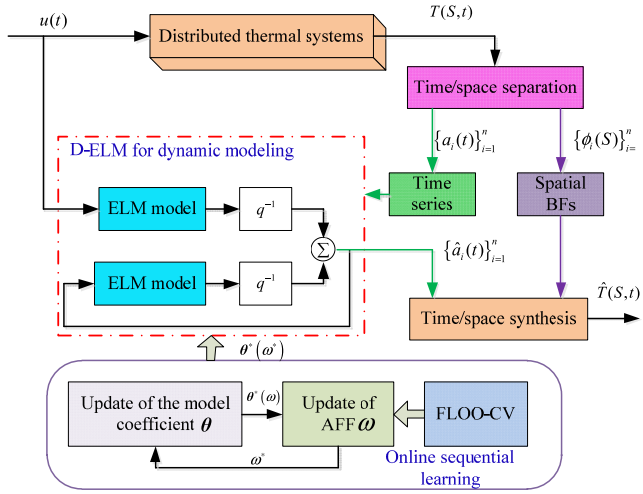

FIGURE 2. The framework of the proposed online spatiotemporal modeling method.

Detailed description and development can be found as follows.

B. DUAL EXTREME LEARNING MACHINE

Suppose the spatiotemporal snapshots of an industrial thermal process are $\{T(S,t)\}_{t=1}^{L}$ and the input signals are $\{u(t)\}_{t=1}^{L}$, where L is the time length. As shown by(1), the spatiotemporal domain of the thermal process is time/space coupled. To perform time/space decouple, KL method is first adopted to learn spatial BFs $\{\phi_i(S)\}_{i=1}^{n}$ from $\{T(S,t)\}_{t=1}^{L}$. The detailed description of time/space separation procedure by KL can be found in [13]. Due to the orthogonality of the spatial BFs, temporal dynamics $a(t) = \{a_i(t)\}_{i=1}^{n}$ can be obtained by $a_i(t) = (\phi_i(t), T(S,t))$, where $(\phi_i(t), T(S,t))$ is defined as the inner product operation of $\phi_i(t)$ and $T(S,t)$. To consider the inherent system structure features, the D-ELM model is developed to approximate the two coupled nonlinear blocks in(14). For mathematical convenience, let $g_i(\cdot) = g(\cdot), f_i(\cdot) = f(\cdot)$. These two nonlinear functions can be approximated using the ELM model, respectively, as [34], [35]:

$$g(a(t)) = \sum_{\sigma}^{N_1} \beta_\sigma G_1(\omega_\sigma \cdot a(t) + \eta_\sigma), \tag{15}$$

$$f(u(t)) = \sum_{\delta}^{N_2} \beta'_\delta G_2\left(\omega'_\delta \cdot u(t) + \eta'_\delta\right), \tag{16}$$

where β_σ and β'_δ are the output weights linking the output node and corresponding hidden nodes, ω_σ and ω'_δ are the input weights connecting the input nodes and corresponding hidden nodes, η_σ and η'_δ are the threshold of corresponding hidden nodes, N_1 and N_2 are the numbers of the hidden nodes, G_1 and G_2 are the activation functions of the hidden layer. With (15) and (16), (14) can be rewritten as the following

equation:

$$a(t) = \sum_{\sigma}^{N_1} \beta_\sigma G_1 (\omega_\sigma \cdot a(t-1) + \eta_\sigma) + \sum_{\delta}^{N_2} \beta'_\delta G_2 \left(\omega'_\delta \cdot u(t-1) + \eta'_\delta\right). \quad (17)$$

The above equation is the mathematical description of the Dual ELM, designed according to the coupled two nonlinear structures. The parameters identification process is similar to general ELM. The unknown parameters ω_σ, ω'_δ, η_σ, η'_δ are randomly generated. The remained unknown parameters β_σ and β'_δ need to be determined according to the input-output data.

Equation (17) can be expressed in a linear regression form, as:

$$a(t) = \boldsymbol{h}^T(t)\,\boldsymbol{\theta}, \quad (18)$$

where $\boldsymbol{h}(t)$ is a parameter vector related to the input-output data and has the form as follows:

$$\boldsymbol{h}(t) = \begin{bmatrix} G_1(\omega_1 \cdot a(t-1)+\eta_1), \ldots, G_1(\omega_{N_1} \cdot a(t-1)+\eta_{N_1}), \\ G_2(\omega'_1 \cdot u(t-1)+\eta_1), \ldots, G_2(\omega'_{N_2} \cdot u(t-1)+\eta_{N_2}) \end{bmatrix}^T, \quad (19)$$

$\boldsymbol{\theta} = \left[\beta_1, \ldots, \beta_{N_1}, \beta'_1, \ldots, \beta'_{N_2}\right]^T$ is the unknown parameter vector to be identified.

To calculate the parameter vector (19), ω_σ, ω'_δ, η_σ and η'_δ are all randomly generated without the knowledge of the training data. They are not only independent of the training data but also of each other. Once they are estimated, the value will be fixed in the following learning procedure.

Equation (18) can be written in a matrix form as:

$$\mathbf{A} = \mathbf{H}\boldsymbol{\theta}, \quad (20)$$

where $\mathbf{A} = [a(2), a(3), \ldots, a(L)]^T$ is the output vector, $\mathbf{H} = \left[\boldsymbol{h}^T(2) \ldots \boldsymbol{h}^T(L)\right]^T$ is the regression matrix. Since $\mathbf{A}$ and $\mathbf{H}$ are known, the unknown parameters $\boldsymbol{\theta}$ can be calculated analytically as:

$$\hat{\boldsymbol{\theta}} = \mathbf{H}^\dagger \mathbf{A}, \quad (21)$$

where $\mathbf{H}^\dagger$ is the Moore-Penrose (MP) generalized inverse of matrix $\mathbf{H}$. Here, we only consider the nonsingular case: $\mathbf{H}^\dagger = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T$

C. ONLINE SEQUENTIAL LEARNING ALGORITHM

The above developed D-ELM method can model the spatiotemporal dynamics of DPSs very well since its model structure matches the general thermal system. However, this model developed in an offline environment should be online updated to keep satisfactory model performance [31], [32]. A simple method to achieve online update is supposed the model to be retrained from scratch repeatedly by combining existing data and newly arriving data, which will lead to a high computational burden in real applications. Therefore, OSLA is required for the spatiotemporal model online updating, which only uses the learned model parameters and the newly arriving data involving time coefficients from KL and the input signals.

Given a chunk of the initial training set $\aleph_0 = \{z(t), a(t)\}_{t=1}^{L_0}$, where $z(t) = [a(t-1), u(t-1)]$, according to (21), the initial model parameters can be calculated by $\hat{\boldsymbol{\theta}}_0 = \mathbf{K}_0^{-1}\mathbf{H}_0^T\mathbf{A}_0$ with $\mathbf{K}_0 = \mathbf{H}_0^T\mathbf{H}_0$, where the subscript 0 denotes the corresponding initial vector or matrix concerning the initial training set $\aleph_0$. The online data are usually arriving one-by-one or chunk-by-chunk. Here, the chunk-by-chunk case will be taken into consideration in the algorithm development. Suppose that a new chunk of data $\aleph_1 = \{z(t), a(t)\}_{t=L_0+1}^{L_0+L_1}$ comes. Equation (11) can be expressed as:

$$\hat{\boldsymbol{\theta}}_1 = \mathbf{K}_1^{-1} \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix}^T \begin{bmatrix} \mathbf{A}_0 \\ \mathbf{A}_1 \end{bmatrix}, \quad (22)$$

where:

$$\mathbf{K}_1 = \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix}^T \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix}, \quad (23)$$

$\mathbf{H}_1$ and $\mathbf{A}_1$ is the regression matrix and output vector for $\aleph_1$, respectively. For online sequential learning, we have to express $\hat{\theta}_1$ as a function of $\hat{\boldsymbol{\theta}}_0$, $\mathbf{K}_1$, $\mathbf{H}_1$ and $\mathbf{A}_1$. Now, $\mathbf{K}_1$ can be written as:

$$\mathbf{K}_1 = \mathbf{H}_0^T\mathbf{H}_0 + \mathbf{H}_1^T\mathbf{H}_1 = \mathbf{K}_0 + \mathbf{H}_1^T\mathbf{H}_1 \quad (24)$$

Then $\hat{\theta}_1$ can be expressed as [36]:

$$\hat{\boldsymbol{\theta}}_1 = \hat{\boldsymbol{\theta}}_0 + \mathbf{K}_1^{-1}\mathbf{H}_1^T\left(\mathbf{A}_1 - \mathbf{H}_1\hat{\boldsymbol{\theta}}_0\right). \quad (25)$$

Generalizing the previous arguments, when (m+1)-*th* trunk of data $\aleph_{m+1} = \{z(t), a(t)\}_{t=\left(\sum_{j=0}^{m} L_j\right)+1}^{\sum_{j=0}^{m+1} L_j}$ arrives, we have:

$$\begin{cases} \mathbf{K}_{m+1} = \mathbf{K}_m + \mathbf{H}_{m+1}^T\mathbf{H}_{m+1} \\ \hat{\boldsymbol{\theta}}_{m+1} = \hat{\boldsymbol{\theta}}_m + \mathbf{K}_{m+1}^{-1}\mathbf{H}_{m+1}^T\left(\mathbf{A}_{m+1} - \mathbf{H}_{m+1}\hat{\boldsymbol{\theta}}_m\right) \end{cases}, \quad (26)$$

where: $\mathbf{A}_{m+1} = \left[a\left(\left(\sum_{j=0}^{m} L_j\right)+1\right), a\left(\left(\sum_{j=0}^{m} L_j\right)+2\right), \ldots a\left(\sum_{j=0}^{m+1} L_j\right)\right]^T$, $\mathbf{H}_{m+1} = \left[\boldsymbol{h}^T\left(\left(\sum_{j=0}^{m} L_j\right)+1\right), \boldsymbol{h}^T\left(\left(\sum_{j=0}^{m} L_j\right)+2\right), \ldots, \boldsymbol{h}^T\left(\sum_{j=0}^{m+1} L_j\right)\right]^T$. By (26), the model parameters $\hat{\boldsymbol{\theta}}_{m+1}$ can be updated by the known parameters $\hat{\boldsymbol{\theta}}_m$ and newly arriving data $\aleph_{m+1} = \{z(t), a(t)\}_{t=\left(\sum_{j=0}^{m} L_j\right)+1}^{\sum_{j=0}^{m+1} L_j}$.

D. ADAPTIVE FORGETTING FACTOR

The OLSA considers that the contribution of newly arriving samples and previously collected samples are equal. Therefore, it adopts equal weight treatment that fails to highlight the role of newly arriving snapshots. Besides, this algorithm

recursively updates the network weights as soon as it gets a new trunk of snapshots. This kind of automatic network weight update mode lacks the flexibility to adjust according to the actual situation and is easy to increase the unnecessary computation. To overcome this problem, the adaptive forgetting factor (AFF) is applied to weaken the previously collected samples by giving a reasonable weight.

Suppose that the model parameters $\hat{\boldsymbol{\theta}}_1$ corresponding to the first trunk of data $\aleph_1 = \{z(t), a(t)\}_{t=L_0+1}^{L_0+L_1}$ can be expressed as:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_1 &= \left(\begin{bmatrix}\mathbf{H}_0\\ \mathbf{H}_1\end{bmatrix}^T\begin{bmatrix}\mathbf{H}_0\\ \mathbf{H}_1\end{bmatrix}\right)^{-1}\begin{bmatrix}\mathbf{H}_0\\ \mathbf{H}_1\end{bmatrix}^T\begin{bmatrix}\mathbf{A}_0\\ \mathbf{A}_1\end{bmatrix}\\ &= \left(\mathbf{H}_0^T\mathbf{H}_0 + \mathbf{H}_1^T\mathbf{H}_1\right)^{-1}\left(\mathbf{H}_0^T\mathbf{A}_0 + \mathbf{H}_1^T\mathbf{A}_1\right). \quad (27)\end{aligned}$$

Since $\mathbf{H}_0^T\mathbf{H}_0$ and $\mathbf{H}_0^T\mathbf{A}_0$ in (27) are related to the previously collected samples, a reasonable weight will be given to them. Thus, (27) can be transformed into the following equation:

$$\hat{\boldsymbol{\theta}}_1 = \left(\omega_0\mathbf{H}_0^T\mathbf{H}_0 + \mathbf{H}_1^T\mathbf{H}_1\right)^{-1}\left(\omega_0\mathbf{H}_0^T\mathbf{A}_0 + \mathbf{H}_1^T\mathbf{A}_1\right), \quad (28)$$

where the weight $\omega_0 \in (0, 1)$ is the AFF, whose function is to weaken the influence of the previously collected samples and indirectly enhance the effect of the newly arriving snapshots.

Define:

$$\mathbf{K}_1 = \omega_0\mathbf{K}_0 + \mathbf{H}_1^T\mathbf{H}_1. \quad (29)$$

Thus, $\hat{\boldsymbol{\theta}}_1$ can be expressed as:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_1 &= \mathbf{K}_1^{-1}\left(\omega_0\mathbf{H}_0^T\mathbf{A}_0 + \mathbf{H}_1^T\mathbf{A}_1\right)\\ &= \mathbf{K}_1^{-1}\left(\omega_0\mathbf{K}_0\mathbf{K}_0^{-1}\mathbf{H}_0^T\mathbf{A}_0 + \mathbf{H}_1^T\mathbf{A}_1\right)\\ &= \mathbf{K}_1^{-1}\left(\omega_0\mathbf{K}_0\hat{\boldsymbol{\theta}}_0 + \mathbf{H}_1^T\mathbf{A}_1\right)\\ &= \mathbf{K}_1^{-1}\left[\left(\mathbf{K}_1 - \mathbf{H}_1^T\mathbf{H}_1\right)\hat{\boldsymbol{\theta}}_0 + \mathbf{H}_1^T\mathbf{A}_1\right]\\ &= \hat{\boldsymbol{\theta}}_0 + \mathbf{K}_1^{-1}\mathbf{H}_1^T\left(\mathbf{A}_1 - \mathbf{H}_1\hat{\boldsymbol{\theta}}_0\right). \quad (30)\end{aligned}$$

Generalizing the previous arguments, when (m+1)-*th* trunk of data $\aleph_{m+1} = \{z(t), a(t)\}_{t=\left(\sum_{j=0}^{m} L_j\right)+1}^{\sum_{j=0}^{m+1} L_j}$ arrives, we have:

$$\begin{cases}\mathbf{K}_{m+1} = \omega_m\mathbf{K}_m + \mathbf{H}_{m+1}^T\mathbf{H}_{m+1}\\ \hat{\boldsymbol{\theta}}_{m+1} = \hat{\boldsymbol{\theta}}_m + \mathbf{K}_{m+1}^{-1}\mathbf{H}_{m+1}^T\left(\mathbf{A}_{m+1} - \mathbf{H}_{m+1}\hat{\boldsymbol{\theta}}_m\right)\end{cases}. \quad (31)$$

From (31), the model parameters $\hat{\boldsymbol{\theta}}_{m+1}$ can be recursively updated by the known model parameters $\hat{\boldsymbol{\theta}}_m$, the newly arriving trunk of data $\aleph_{m+1}$, and the weight ω_m. Since ω_m represents the timeliness effect of previously collected samples, and the contribution of each trunk of data will not be equal in a different situation, the weight ω_m should be adaptively calculated in continuous incremental learning. Here, a novel FLOO-CV method is developed for the optimal calculating of ω_m.

E. OPTIMAL DESIGN OF THE AFF

The FLOO-CV method can be used to evaluate the generalization ability of neural network models. Its generalization error estimation is unbiased and unaffected by random factors. The verification process can be repeated entirely with high computational efficiency. To select the optimal AFF, a FLOO-CV method proposed by Mao *et al.* [40] is developed here.

Take the trunk of data $\aleph_1 = \{z(t), a(t)\}_{t=1}^{L_1}$ into consideration, let each input-output data $(z(s), a(s))$ in set $\aleph_1$ as the testing sample, and all the remaining data are used as training samples.

By (31), we can update the model parameter as follows:

$$\begin{cases}\mathbf{K}_s = \omega_0\mathbf{K}_0 + \mathbf{H}_s^T\mathbf{H}_s\\ \hat{\boldsymbol{\theta}}_s = \hat{\boldsymbol{\theta}}_0 + \mathbf{K}_s^{-1}\mathbf{H}_s^T\left(\mathbf{A}_s - \mathbf{H}_s\hat{\boldsymbol{\theta}}_0\right)\end{cases} \quad (32)$$

where:

$$\mathbf{A}_s = [a(1), \ldots, a(s-1), a(s+1), \ldots, a(L_1)]^T,$$
$$\mathbf{H}_s = \left[\boldsymbol{h}^T(1), \ldots, \boldsymbol{h}^T(s-1), \boldsymbol{h}^T(s+1), \ldots, \boldsymbol{h}^T(L_1)\right]^T,$$

Using model (32), the predicted error using the testing sample $z(s)$ can be calculated, which can be expressed as follows:

$$\bar{e}(s) = a(s) - \boldsymbol{h}^T(s)\hat{\boldsymbol{\theta}}_s \quad (33)$$

In [40], the generalization error expressed in (33) can be calculated as:

$$\bar{e}(s) = \frac{a_s - \mathbf{H}_{z_s}\mathbf{H}^{\dagger}\mathbf{A}_s}{1 - \left(\mathbf{H}_{z_s}\mathbf{H}^{\dagger}\right)_s} \quad (34)$$

where $(\cdot)_s$ means the *i*th element, $\mathbf{H}$ is hidden layer matrix, and H_{z_s} is the row about the sample z_s in $\mathbf{H}$.

The main difference our method and the method in [40] is that the old sample is directly delete in [40], while the old sample is given a AFF to weak its affect to the new model in our method. Therefore, the only affected element is $\mathbf{H}$. The $\mathbf{H}^{\dagger}$ in our method can be expressed as:

$$\begin{aligned}\mathbf{H}^{\dagger} &= \mathbf{K}^{-1}\mathbf{H}^T\\ &= \left(\omega_0\mathbf{K}_0 + \mathbf{H}_s^T\mathbf{H}_s\right)\mathbf{H}^T \quad (35)\end{aligned}$$

Substituting (35) into (34), the generalization error in *s*th can be calculated. Finally, the optimal selection of weight ω_0 can be constructed as follows:

$$\begin{aligned}&\arg\min J(\omega_0) = \frac{1}{L_1}\sum_{s=1}^{L_1}\bar{e}^2(s)\\ &\text{subject to: } \omega_0 \in (0, 1) \quad (36)\end{aligned}$$

Many evolutionary algorithms (EAs), such as the genetic algorithm (GA) and Newton method, can be applied here to solve the optimization problem (36) and find the optimal weight value.

The proposed online sequential learning algorithm with a forgetting mechanism for dual ELM can be summarized as follows.

Proposed Algorithm:

Step 1. Initialization phase :

1) Set $m = 0$. Choose the activation function G_1 and G_2 of hidden layer and the number of hidden nodes N_1 and N_2. Randomly assign the hidden parameters $\omega_\sigma, \omega'_\delta, \eta_\sigma, \eta'_\delta$ and fixed in the following learning process.
2) Compute the initial output matrix of the hidden layer:

$$\mathbf{H}_0 = \left[h^T(2) \cdots h^T(L_0) \right]^T. \tag{37}$$

3) Calculate the initial output weight:

$$\hat{\boldsymbol{\theta}}_0 = \mathbf{K}_0^{-1}\mathbf{H}_0^T\mathbf{A}_0, \tag{38}$$

where $\mathbf{K}_0 = \mathbf{H}_0^T\mathbf{H}_0$ and $\mathbf{A}_0 = [a(2), a(3), \ldots, a(L_0)]^T$.

Step 2. Online sequential learning with adaptive forgetting factor:

1) When the (m+1)-*th* trunk of data $\aleph_{m+1} = \{z(t), a(t)\}_{t=\left(\sum_{j=0}^{m} L_j\right)+1}^{\sum_{j=0}^{m+1} L_j}$ arrives, calculate the activation function $\mathbf{H}_{m+1}$:

$$\mathbf{H}_{m+1} = \left[\boldsymbol{h}^T\left(\left(\sum_{j=0}^{m} L_j\right)+1\right), \boldsymbol{h}^T\left(\left(\sum_{j=0}^{m} L_j\right)+2\right), \ldots, \boldsymbol{h}^T\left(\sum_{j=0}^{m+1} L_j\right)\right]^T. \tag{39}$$

2) Establishing the optimal value of weight ω_m by repeating (32) to (36).
3) Calculate:

$$\mathbf{K}_{m+1} = \omega_m\mathbf{K}_m + \mathbf{H}_{m+1}^T\mathbf{H}_{m+1}. \tag{40}$$

4) Calculate the output weight

$$\hat{\boldsymbol{\theta}}_{m+1} = \hat{\boldsymbol{\theta}}_m + \mathbf{K}_{m+1}^{-1}\mathbf{H}_{m+1}^T\left(\mathbf{A}_{m+1} - \mathbf{H}_{m+1}\hat{\boldsymbol{\theta}}_m\right). \tag{41}$$

5) Used the calculated output weight $\hat{\boldsymbol{\theta}}_{m+1}$ to estimate the dual ELM model (17).
6) Set $m = m + 1$, then go to (1) of step 2.

Remark 1: To ensure rank $(\mathbf{H}_0) = N_1 + N_2$, the number of initialization training data L_0 should not be less than the sum of the hidden neurons numbers of two ELM networks.

Remark 2: From (29) and (30), it is easy to find that the online sequential implementation of the least-square solution (21) is similar to the recursive least-square (RLS) algorithm. Therefore, all the convergence proof of RLS can be extended to the proposed algorithm.

Remark 3: From (29), we can get that:

$$\mathbf{K}_2 = \omega_1\mathbf{K}_1 + \mathbf{H}_2^T\mathbf{H}_2. \tag{42}$$

Combining (42) and (29), the above equation can be further expressed as:

$$\mathbf{K}_2 = \omega_1\omega_0\mathbf{K}_0 + \omega_1\mathbf{H}_1^T\mathbf{H}_1 + \mathbf{H}_2^T\mathbf{H}_2. \tag{43}$$

Expand (43) continually, we can find that:

$$\mathbf{K}_{m+1} = \prod_{i=0}^{m}\omega_i\mathbf{K}_0 + \prod_{i=1}^{m}\omega_i\mathbf{H}_1^T\mathbf{H}_1 + \cdots + \omega_m\mathbf{H}_m^T\mathbf{H}_m + \mathbf{H}_{m+1}^T\mathbf{H}_{m+1}. \tag{44}$$

Since $\omega_i \in (0, 1)$, the effect of previous samples decreases exponentially as online learning continuously proceeds. Especially, when $\omega_i = 1$, the contributions of previous samples and new samples are equal. This will be the case of OSLA. Thus, the proposed method can be regarded as a general form of OSLA.

F. SPACE/TIME SYNTHESIS

With the obtained KL-based spatial BFs and the estimated temporal model, spatiotemporal variables can be reconstructed online as follows:

$$\hat{T}(S, t) = \sum_{i=1}^{n}\phi_i(S)\hat{a}_i(t). \tag{45}$$

IV. EXPERIMENT VALIDATION

A. EXPERIMENT SETUP

To verify the proposed model's effectiveness, a typical industrial thermal process of curing oven is studied. As shown in Fig. 3, the curing oven is a critical equipment to provide the required temperature distribution during the curing process in the semiconductor back-end packaging industry. There are four same heaters placed on the top of the oven, controlled by pulse-width modulation (PWM) signals. Sixteen thermocouples are placed on the lead frame uniformly for data collected, as shown in Fig. 4. In total, about 2100 time-series snapshots are collected by all the sensors in a fixed sampling interval $\Delta t = 10s$, which will be used for model training and online learning. In the online spatiotemporal modeling, the collected samples are separated into two parts. The first 800 samples are used for D-ELM-based spatiotemporal model construction, and the last 1300 samples are used for model testing and updating.

FIGURE 3. The snap curing oven system.

For the convenience of model verification and comparison, the following error indexes are given:

1) Spatiotemporal prediction error (e)

$$e(S_i, t) = T(S_i, t) - \hat{T}(S_i, t), \quad i = 1, \ldots, N. \tag{46}$$

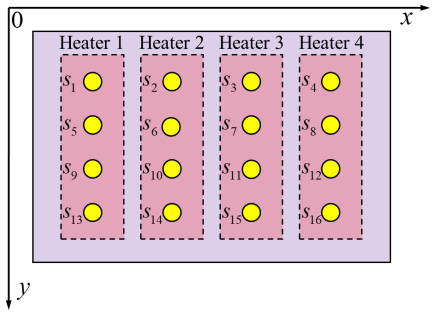

FIGURE 4. Sensor locations for snapshots collected.

2) Absolute relative error (ARE)

$$ARE = \frac{\left|T(S_i, t) - \hat{T}(S_i, t)\right|}{T(S_i, t)}, \quad i = 1, \ldots, N. \tag{47}$$

3) Root mean square error ($RMSE$)

$$RMSE = \sqrt{\frac{1}{NL}\sum_{i=1}^{N}\sum_{t=1}^{L}\left(T(S_i, t) - \hat{T}(S_i, t)\right)^2}. \tag{48}$$

4) Temporal normalized absolute error ($TNAE$)

$$TNAE = \frac{1}{L}\sum_{t=1}^{L}\left|T(S_i, t) - \hat{T}(S_i, t)\right|, \quad i = 1, \ldots, N. \tag{49}$$

B. D-ELM BASED OFFLINE SPATIOTEMPORAL MODEL

For the D-ELM-based spatiotemporal model construction. KL method is first used to calculate the spatial BFs, where the orders are selected to 3 [33]. Then, the time coefficients $\{a_i(t)\}_{i=1}^{n}$ can be derived by projecting spatiotemporal samples onto the BFs. With time coefficients and the corresponding input signal $u(t)$, the low-order temporal model can be estimated using (17) to (21). Finally, D-ELM based spatiotemporal model can be reconstructed using space/time synthesis. To evaluate the performance of the model, the last 1300 samples are used for testing. To verify the effectiveness of the D-ELM method, comparisons between the measured and predicted temporal coefficients are shown in Fig. 5, where the red lines correspond to the predicted temporal coefficients and the black lines to the measured ones. They show satisfactory agreement between the predicted and measured dynamics. Besides, the predicted temperature distribution and corresponding ARE indexes at the 2100-th sample are simulated, as shown in Fig. 6. It can be seen that the maximum ARE of the predicted result is within 1.2%, which means the acquired D-ELM-based spatiotemporal model can approximate the actual DPS system well.

To evaluate the efficiency of the proposed D-ELM based model, Dual least square SVM (LS-SVM) [33], MLP [26], and random vector functional-link (RVFL) [42] are also developed for the spatiotemporal model construction. Model performance in terms of TNAE, RMSE, and simulation time

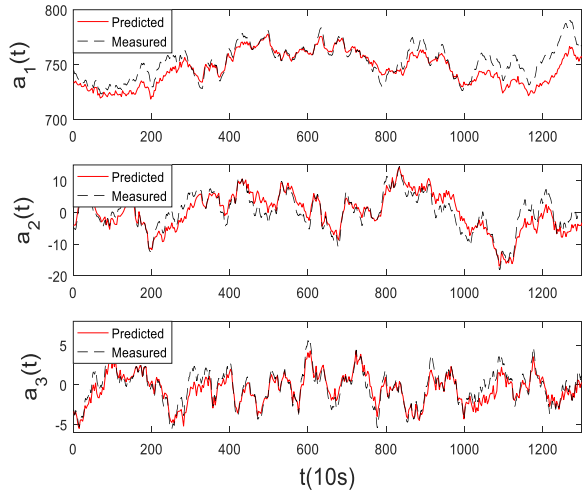

FIGURE 5. Comparisons between the predicted and measured time coefficients.

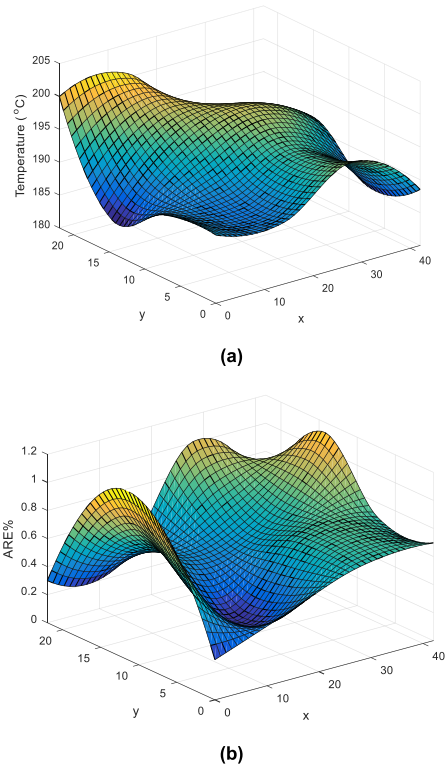

FIGURE 6. Simulation results at 2100-th sample. (a) the real temperature distribution; (b) ARE distribution.

is shown in Fig. 7 and Table 1. It can be seen that both D-ELM and Dual LS-SVM show better accuracy on RMSE due to the dual-model structure that fits the two inherently coupled nonlinearities of the thermal system. In addition, the D-ELM based model has the best TNAE performance at each sensor location among these four models. Meanwhile, the learning speed of D-ELM has the same order of magnitude as the RVFL based model, which highlights the advantages of faster computing speed compared with Dual LS-SVM.

TABLE 1. Performance comparison in terms of RMSE and simulation time.

Method	Time	RMSE		
		Min	Max	Avg
Dual LS-SVM	0.616	1.70	2.14	1.92
RVFL	**0.045**	2.16	3.08	2.54
MLP	17.052	2.24	3.39	2.73
D-ELM	0.055	1.65	2.03	**1.81**

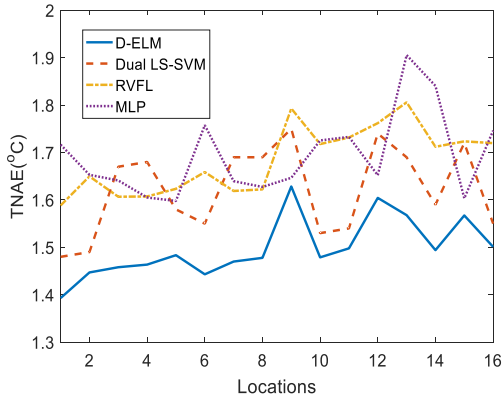

FIGURE 7. Performance comparison in terms of TNAE.

C. AFFD-ELM BASED ONLINE SPATIOTEMPORAL MODEL

For the online learning process, suppose the 1300 samples are coming chunk-by-chunk with a fixed chunk size, is set as 10. Thus, the 1300 snapshots can be separated into 130 continuous blocks. The trace of the weight value ω is depicted in Fig. 8. It is clear to find that the optimal ω value is overtime for the time-varying dynamics of the nonlinear curing process. To evaluate the model performance of the proposed online method and D-ELM model, predicted temperature evolutions and corresponding ARE indexes at sensor 6 and 11 are simulated, as shown in Fig. 9 and Fig. 10. As can be seen from the figures, the proposed online model can extremely approximate the actual dynamic evolution.

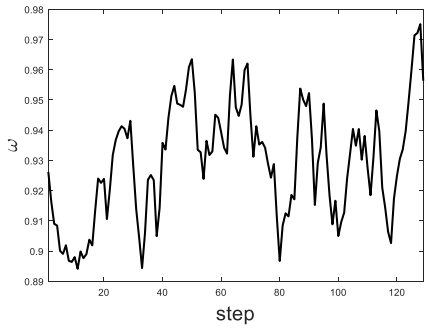

FIGURE 8. Trace of ω value.

For online model comparison, FOS-ELM and TOS-ELM-based spatiotemporal models are also developed for the thermal curing process under the same experimental conditions. To verify the superiority of the proposed model,

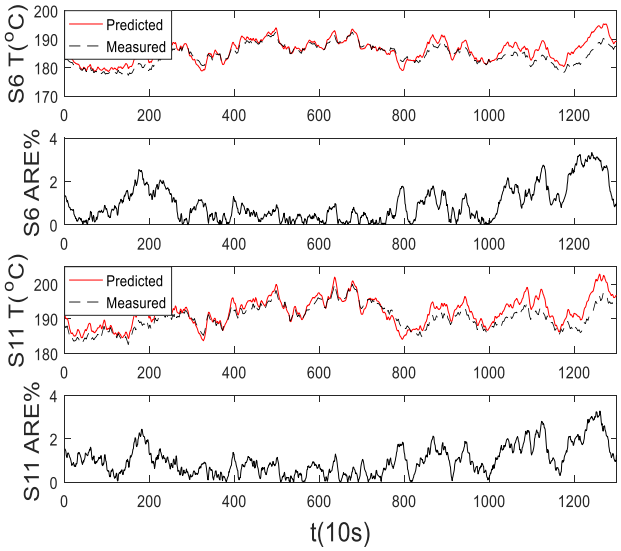

FIGURE 9. Performance comparisons at sensor 6 and 11 using D-ELM based model.

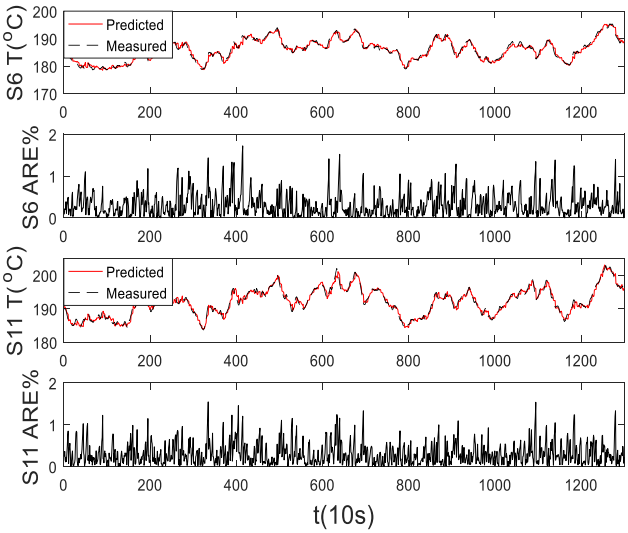

FIGURE 10. Performance comparisons at sensor 6 and 11 using the AFFD-ELM based online model.

the error-index ARE at the 2100-th testing sample is shown in Fig. 11. Compared with Fig. 6(b), these three online models can all present better ARE performance than the D-ELM model. The maximum ARE of these three models reduced from 1.2% of D-ELM to below 0.5%, 0.6% and 0.7%, respectively, which means they have the higher prediction accuracy. The reason is that D-ELM, as an off-line model, cannot reflect the variation of model parameters over time, so it is difficult to adapt to the large-scale time-varying nonlinear DPS systems. For further performance comparisons of these models, the error indexes TNAE and RMSE are also calculated, as shown in Fig. 12 and Table 2 . It is obvious that the proposed method has the best RMSE performance and the minimum TNAE compared to those of the TOS-ELM and the FOS-ELM, indicating that the proposed model has

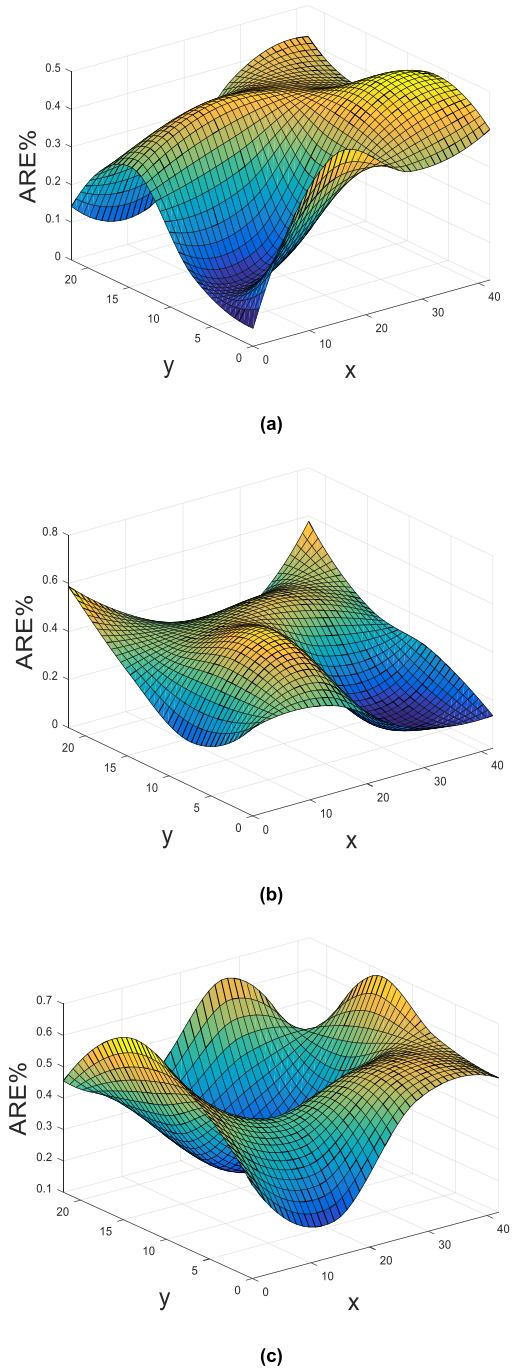

FIGURE 11. ARE distributions at 2100-th sample. (a) proposed model; (b) FOS-ELM based model; (c) TOS-ELM based model.

a higher accuracy than the other two online spatiotemporal models. The reason is that the optimal AFF design can make the proposed model more suitable for the modeling problem of time-varying DPSs.

D. RESULT ANALYSIS

According to the simulation and experiment results, the proposed AFFD-ELM has the following advantages:

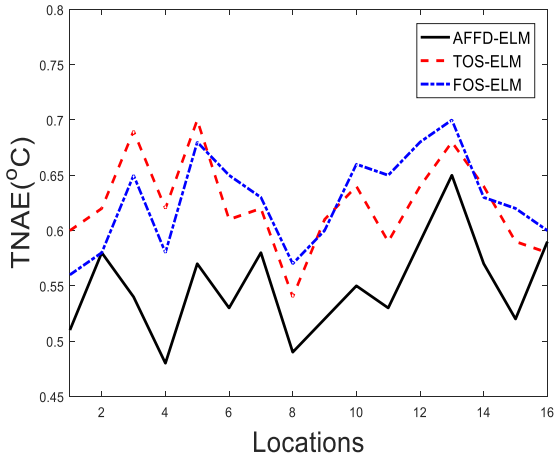

FIGURE 12. Online spatiotemporal model performance comparison in terms of TNAE.

TABLE 2. Model performance comparison in terms of RMSE.

Method	samples (1-400)	samples (401-800)	samples (801-1300)	All samples
TOS-ELM	1.1059	1.1237	1.1436	1.1257
FOS-ELM	1.1326	1.1457	1.1639	1.1402
Proposed model	**0.8287**	**0.8913**	**0.9015**	**0.9543**

1) Improved nonlinear modeling performance: Compared to other popular spatiotemporal models, D-ELM has the best average RMSE of 1.81 in Table 1. This indicates that it can capture the nonlinear mapping relationship between the system inputs and time coefficients well.
2) Adapted to the time-varying dynamics of DPSs: the experiment results show that whenever a new chunk of data comes, the proposed method outperforms FOS-ELM and TOS-ELM-based spatiotemporal models in terms of ARE, TNAE and RMSE in online learning process.
3) Fast model execution time: DELM requires only 0.055s to complete simulation, which is close to that of RVFL based model. Furthermore, Mao *et al.* [40] has proved that FLOO-CV method has the good time performance in online update process. All these shows that the proposed method is enough practical.

V. CONCLUSION

In this work, AFFD-ELM based online spatiotemporal modeling is proposed for large-scale time-varying DPSs. This method is based on recursively updating the low-order temporal model through online sequential learning of the continuously arriving samples. In this way, the spatiotemporal model can be inherited and updated through the whole online learning process. From the perspective of modeling accuracy, the proposed model is based on the accurate D-ELM-based spatiotemporal model and online updating strategy. Thus, it gives an extremely close approximation to the actual process. From the perspective of modeling efficiency, the

application of ELM can lead to a simple model structure and high learning speed. Embedding of AFF can ensure the online modeling has the advantages of improving the learning effects and reduce the lousy affection of previous data. Therefore, this method can be applied to a class of time-varying DPSs. Real-time experiments on a curing oven demonstrate the efficiency and viability of the proposed model.

REFERENCES

[1] B. Luo, T. Huang, H. Wu, and X. Yang, "Data-driven H_∞ control for nonlinear distributed parameter systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 11, pp. 2949–2961, Aug. 2015.

[2] B. Luo and H.-N. Wu, "Online policy iteration algorithm for optimal control of linear hyperbolic PDE systems," *J. Process Control*, vol. 22, no. 7, pp. 1161–1170, Aug. 2012.

[3] Y. Zhou, H. Deng, H.-X. Li, and S. Xie, "Dual separation-based spatio-temporal modeling methodology for battery thermal process under non-homogeneous boundary conditions," *IEEE Trans. Transport. Electrific.*, early access, Feb. 23, 2021, doi: 10.1109/TTE.2021.3061426.

[4] J. Dong, Q. Wang, M. Wang, and K. Peng, "Data-driven quality monitoring techniques for distributed parameter systems with application to hot-rolled strip laminar cooling process," *IEEE Access*, vol. 6, pp. 16646–16654, 2018.

[5] S. Gupta, A. K. Sahoo, and U. K. Sahoo, "Volterra and Wiener model based temporally and spatio-temporally coupled nonlinear system identification: A synthesized review," *IETE Tech. Rev.*, vol. 38, no. 3, pp. 303–327, 2020.

[6] R. Zhang, J. Tao, R. Lu, and Q. Jin, "Decoupled ARX and RBF neural network modeling using PCA and GA optimization for nonlinear distributed parameter systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 2, pp. 457–469, Feb. 2018.

[7] T. Xiao, X.-D. Li, and S. Wang, "Dominant-modes-based sliding-mode observer for estimation of temperature distribution in rapid thermal processing system," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 2673–2681, May 2019.

[8] J. A. Rad, S. Kazem, and K. Parand, "Optimal control of a parabolic distributed parameter system via radial basis functions," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 19, no. 8, pp. 2559–2567, Aug. 2014.

[9] T. Dragicevic, X. Lu, J. C. Vasquez, and J. M. Guerrero, "DC microgrids—Part I: A review of control strategies and stabilization techniques," *IEEE Trans. Power Electron.*, vol. 31, no. 7, pp. 4876–4891, Sep. 2016.

[10] X.-X. Zhang, Z.-Q. Fu, S.-Y. Li, T. Zou, and B. Wang, "A time/space separation based 3D fuzzy modeling approach for nonlinear spatially distributed systems," *Int. J. Autom. Comput.*, vol. 15, no. 1, pp. 52–65, Feb. 2018.

[11] J.-W. Wang and H.-N. Wu, "Exponential pointwise stabilization of semilinear parabolic distributed parameter systems via the Takagi–Sugeno fuzzy PDE model," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 1, pp. 155–173, Feb. 2018.

[12] Y. Feng and H.-X. Li, "Detection and spatial identification of fault for parabolic distributed parameter systems," *IEEE Trans. Ind. Electron.*, vol. 66, no. 9, pp. 7300–7309, Sep. 2019.

[13] B.-C. Wang and H.-X. Li, "A sliding window based dynamic spatiotemporal modeling for distributed parameter systems with time-dependent boundary conditions," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2044–2053, Apr. 2019.

[14] X. Lu, W. Zou, and M. Huang, "Robust spatiotemporal LS-SVM modeling for nonlinear distributed parameter system with disturbance," *IEEE Trans. Ind. Electron.*, vol. 64, no. 10, pp. 8003–8012, Oct. 2017.

[15] M. Jiang, S. Liu, and J. Wu, "Modified empirical eigenfunctions and its applications for model reduction of nonlinear spatiotemporal systems," *Math. Problems Eng.*, vol. 2018, pp. 1–12, Nov. 2018.

[16] X.-B. Meng, H.-X. Li, and H.-D. Yang, "Evolutionary design of spatio–temporal learning model for thermal distribution in lithium-ion batteries," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 2838–2848, May 2019.

[17] H. Deng, H.-X. Li, and G. Chen, "Spectral-approximation-based intelligent modeling for distributed thermal processes," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 5, pp. 686–700, Sep. 2005.

[18] J. Shen, T. Tang, and L. L. Wang, *Spectral Methods: Algorithms, Analysis and Applications*. New York, NY, USA: Springer, 2011.

[19] D. You, X. Gao, and S. Katayama, "WPD-PCA-based laser welding process monitoring and defects diagnosis by using FNN and SVM," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 628–636, Jan. 2015.

[20] J. Hu and H. Zhang, "Numerical methods of Karhunen–Loève expansion for spatial data," *Econ. Qual. Control*, vol. 30, no. 1, pp. 49–58, Jan. 2015.

[21] W. Shen, K. Xu, L. Deng, and S. Zhang, "A karhunen-loeve Galerkin online modeling approach for the thermal dynamics of li-ion batteries," *IEEE Access*, vol. 8, pp. 187893–187901, 2020.

[22] Y. Fan, K. Xu, H. Wu, Y. Zheng, and B. Tao, "Spatiotemporal modeling for nonlinear distributed thermal processes based on KL decomposition, MLP and LSTM network," *IEEE Access*, vol. 8, pp. 25111–25121, 2020.

[23] K. A. Hoo and D. Zheng, "Low-order control-relevant models for a class of distributed parameter systems," *Chem. Eng. Sci.*, vol. 56, no. 23, pp. 6683–6710, Dec. 2001.

[24] Z. Ren, Z. Zhao, Z. Wu, and T. Chen, "Dynamic optimal control of a one-dimensional magnetohydrodynamic system with bilinear actuation," *IEEE Access*, vol. 6, pp. 24464–24474, 2018.

[25] C. Qi, H. Li, X. Zhang, X. Zhao, S. Li, and F. Gao, "Time/space-separation-based SVM modeling for nonlinear distributed parameter processes," *Ind. Eng. Chem. Res.*, vol. 50, no. 1, pp. 332–341, 2011.

[26] S. S. Liew, M. Khalil-Hani, and R. Bakhteri, "An optimized second order stochastic learning algorithm for neural network training," *Neurocomputing*, vol. 186, pp. 74–89, Apr. 2016.

[27] W. Xiong, X. Yang, L. Ke, and B. Xu, "EM algorithm-based identification of a class of nonlinear Wiener systems with missing output data," *Nonlinear Dyn.*, vol. 80, nos. 1–2, pp. 329–339, Apr. 2015.

[28] M. Wang, N. Li, S. Li, and H. Shi, "Embedded interval type-2 T-S fuzzy time/space separation modeling approach for nonlinear distributed parameter system," *Ind. Eng. Chem. Res.*, vol. 50, no. 24, pp. 13954–13961, Dec. 2011.

[29] H.-X. Li and C. Qi, "Incremental modeling of nonlinear distributed parameter processes via spatiotemporal kernel series expansion," *Ind. Eng. Chem. Res.*, vol. 48, no. 6, pp. 3052–3058, Mar. 2009.

[30] M. Jiang, X. Li, J. Wu, and G. Wang, "A precision on-line model for the prediction of thermal crown in hot rolling processes," *Int. J. Heat Mass Transf.*, vol. 78, pp. 967–973, Nov. 2014.

[31] X. Lu, W. Zou, and M. Huang, "An adaptive modeling method for time-varying distributed parameter processes with curing process applications," *Nonlinear Dyn.*, vol. 82, nos. 1–2, pp. 865–876, Oct. 2015.

[32] Z. Wang and H.-X. Li, "Incremental spatiotemporal learning for online modeling of distributed parameter systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 12, pp. 2612–2622, Dec. 2019.

[33] K.-K. Xu, H.-X. Li, and H.-D. Yang, "Dual least squares support vector machines based spatiotemporal modeling for nonlinear distributed thermal processes," *J. Process Control*, vol. 54, pp. 81–89, Jun. 2017.

[34] G. Huang, D. H. Wang, and L. Yuan, "Extreme learning machines: A survey," *Int. J. Mach. Learn. Cybern.*, vol. 2, pp. 107–122, Jun. 2011.

[35] W. Mao, M. Jiang, J. Wang, and Y. Li, "Online extreme learning machine with hybrid sampling strategy for sequential imbalanced data," *Cognit. Comput.*, vol. 9, no. 6, pp. 780–800, Dec. 2017.

[36] N.-y. Liang, G.-b. Huang, P. Saratchandran, and N. Sundararajan, "A fast and accurate online sequential learning algorithm for feedforward networks," *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1411–1423, Nov. 2006.

[37] X. Jin, H. D. Yang, K. K. Xu, and C. J. Zhu, "Dual extreme learning machines-based spatiotemporal modeling for nonlinear distributed thermal processes," *Int. J. Comput. Methods*, vol. 18, no. 1, Feb. 2021, Art. no. 2050026.

[38] Y. Gu, J. Liu, Y. Chen, X. Jiang, and H. Yu, "TOSELM: Timeliness online sequential extreme learning machine," *Neurocomputing*, vol. 128, pp. 119–127, Mar. 2014.

[39] J. Zhao, Z. Wang, and D. S. Park, "Online sequential extreme learning machine with forgetting mechanism," *Neurocomputing*, vol. 87, pp. 79–89, Jun. 2012.

[40] W. Mao, J. Wang, L. He, and Y. Tian, "Online sequential prediction of imbalance data with two-stage hybrid strategy by extreme learning machine," *Neurocomputing*, vol. 261, pp. 94–105, Oct. 2017.

[41] Z. Shao, M. J. Er, and N. Wang, "An efficient leave-one-out cross-validation-based extreme learning machine (ELOO-ELM) with minimal user intervention," *IEEE Trans. Cybern.*, vol. 46, no. 8, pp. 1939–1951, Aug. 2016.

[42] Y.-H. Pao, G.-H. Park, and D. J. Sobajic, "Learning and generalization characteristics of the random vector functional-link net," *Neurocomputing*, vol. 6, no. 2, pp. 163–180, Apr. 1994.

KANGKANG XU received the B.E. and Ph.D. degrees in mechatronics engineering from Central South University, Changsha, China, in 2012 and 2017, respectively.

He is currently an Associate Professor with the School of Electromechanical Engineering, Guangdong University of Technology, Guangzhou, China. His research interests include distributed parameter systems and intelligent manufacturing.

XI TAN received the B.E. degree in mechanical design manufacture and automation from the Hunan Institute of Science and Technology, Yueyang, China, in 2020. He is currently pursuing the master's degree with the School of Electromechanical Engineering, Guangdong University of Technology.

His research interests include distributed parameter systems and intelligent modeling.

BI FAN received the B.E. degree in automation from the Ocean University of China, in 2006, the M.E. degree in mechatronics from the University of Shanghai for Science and Technology, in 2010, and the Ph.D. degree in system engineering and engineering management from the City University of Hong Kong, in 2015.

He is currently an Assistant Professor with the Department of Management Science, Shenzhen University. His research interests include machine learning and uncertainty quantification in nonlinear engineering systems.

TENGFEI XIAO received the M.S. degree in pattern recognition and intelligent systems from Sun Yat-Sen University, China, in 2011, and the Ph.D. degree from the Department of Systems Engineering and Engineering Management, City University of Hong Kong, Hong Kong, in 2016.

Since 2016, he has been a Distinguished Associate Research Fellow with the School of Intelligent System Engineering, Sun Yat-Sen University. His research interests include intelligent control and distributed parameter systems.

XI JIN received the B.E. degree in mechatronics engineering from the Xi'an University of Technology, China, in 2013, and the M.E. degree in mechatronics from the Guangdong University of Technology, China, in 2018. He is currently pursuing the Ph.D. degree with the Guangdong University of Technology.

His research interests include distributed parameter systems and optimization applications.

CHENGJIU ZHU received the M.E. degree in mechanical engineering from the Guangdong University of Technology, Guangzhou, China, in 2015. He is currently pursuing the Ph.D. degree with the Guangdong University of Technology.

His research interests include green manufacturing and intelligent manufacturing.