

Article

Cholesky Factorization Based Online Sequential Multiple Kernel Extreme Learning Machine Algorithm for a Cement Clinker Free Lime Content Prediction Model

Pengcheng Zhao ^{1,*} , Ying Chen ¹ and Zhibiao Zhao ²

¹ Department of Physics and Electromechanical, Shijiazhuang College, Shijiazhuang 050035, China; chenyingyanshan@163.com

² School of Automation and Electrical Engineering, Tianjin University of Technology and Education, Tianjin 066004, China; zhaozhibiao@tute.edu.cn

* Correspondence: zhaopengcheng@ysu.edu.cn

Abstract: Aiming at the difficulty in real-time measuring and the long offline measurement cycle for the content of cement clinker free lime (fCaO), it is very important to build an online prediction model for fCaO content. In this work, on the basis of Cholesky factorization, the online sequential multiple kernel extreme learning machine algorithm (COS-MKELM) is proposed. The LDL^T form Cholesky factorization of the matrix is introduced to avoid the large operation amount of inverse matrix calculation. In addition, the stored initial information is utilized to realize online model identification. Then, three regression datasets are used to test the performance of the COS-MKELM algorithm. Finally, an online prediction model for fCaO content is built based on COS-MKELM. Experimental results demonstrate that the fCaO content model improves the performance in terms of learning efficiency, regression accuracy, and generalization ability. In addition, the online prediction model can be corrected in real-time when the production conditions of cement clinker change.

Keywords: Cholesky factorization; extreme learning machine; free lime; multiple kernel learning; online sequential



Citation: Zhao, P.; Chen, Y.; Zhao, Z. Cholesky Factorization Based Online Sequential Multiple Kernel Extreme Learning Machine Algorithm for a Cement Clinker Free Lime Content Prediction Model. *Processes* **2021**, *9*, 1540. <https://doi.org/10.3390/pr9091540>

Academic Editor: Jae-Yoon Jung

Received: 9 August 2021

Accepted: 25 August 2021

Published: 29 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Cement is an important building material. Its quality will affect the building safety directly [1,2]. In addition, the cement clinker is a key product during cement production. At the same time, the content of cement clinker free lime (fCaO) is an important index to evaluate the clinker quality [3]. It is the main factor that impacts the cement stability and clinker strength. Thus, operators and engineers should adjust the variables of cement production according to the fCaO content in real time. In the process of cement production, the cement clinker fCaO content should be controlled within a certain range [4]. When the fCaO content is higher than the standard range, the cement clinker is unqualified. On the other hand, when it is lower than the standard range, the clinker is over-burning and the energy consumption increases [5].

At present, common measuring methods of fCaO content are chemical analysis and fluorescence analyzers. The measurement of cement clinker fCaO content is mainly obtained by manual sampling intermittently, and then the sample cement clinker are sent to the laboratory for manual analysis [6]. According to the requirements of different cement production lines, the time interval of laboratory testing fCaO content varies from 1 to 4 h. However, there is a certain time difference between each process in the cement clinker firing process, which makes the measurement results of fCaO content have a great lag for the production guidance in the clinker firing process. The measurement results can not meet the requirements of real-time optimization control in the cement clinker firing process [7]. Therefore, the essential work of estimating fCaO content is to set up a high

precision online prediction model which is very important for monitoring and optimizing the operations of cement plants.

Considering the nonlinearity and uncertainty between the fCaO content and the production process related variables, some scholars have established the regression prediction model by using the method of neural networks modeling. Fuzzy entropy and a neural-net ensemble with random weights are used to build the soft sensor of fCaO content [8]. Szatvanyi et al. [9] and Li et al. [10] combined flame image features and process variables methods to build the prediction model for fCaO content. Liu et al. [11] proposed a support vector machine ensemble model to estimate fCaO content. A time series analysis method [12] and an improved combination modeling method [13] were used for soft measurement of cement clinker fCaO content. Some neural network methods of establishing a prediction model in the cement calcination process were also proposed, such as time-varying delay deep belief network [14], multi-channel CNN with moving window [15], deep belief network with sliding window [16], and two-dimensional convolutional neural network [17]. In addition, other researchers working on the online prediction model for fCaO content have been published. Pani et al. [18,19] used a feed-forward artificial neural network and fuzzy inference to build a soft sensor model for online prediction fCaO content. The multivariate time series analysis and convolutional neural network method [20] and the kernel extreme learning machine algorithm [21] were proposed for online cement clinker quality monitoring.

Some of the above scholars established the prediction models of cement clinker fCaO content based on the offline neural network. However, when the training sample data is too large or the working conditions of cement clinker firing changed, the modeling process of the prediction model needs to be repeated. This will greatly increase the computational amount of the cement clinker fCaO content modeling process. In addition, some of the above scholars established the prediction models of cement clinker fCaO content based on the online neural network. However, due to the complexity of the cement clinker firing process, the performance of the online prediction model of the clinker fCaO content is poor, and the learning speed of parameters training is slow, which leads to the poor prediction effect of the cement clinker fCaO content. Therefore, it is necessary to deeply study the new online sequential learning neural network algorithm applicable to the cement clinker fCaO content prediction model. This can not only solve the problem of establishing and predicting the cement clinker fCaO content prediction model but also overcome the disadvantages of the existing conventional neural network algorithm, such as poor network performance and slow learning speed.

In recent years, the extreme learning machine (ELM) [22,23] and kernel extreme learning machine (KELM) [24] are proposed and have been successfully applied to model identification, such as saliency detection [25], gesture recognition [26], image classification [27], nonlinear fault detection [28], seepage time soft sensor model of nonwoven fabric [29], and rolling bearing sub-health recognition [30]. However, different kernel functions have different characteristics, and the performances are varying in different applications. Thus, the multiple kernel extreme learning machine (MKELM) has been proposed in [31] as a combination of multiple kernel learning and ELM, and it has also been successfully applied to many applications, such as network intrusion detection [32,33] and human activity recognition [34]. Considering a strong nonlinear relationship between the fCaO content and the related variables during cement production, MKELM is adopted to build an online prediction model for fCaO content in this paper. However, it is necessary to study how to select the appropriate kernel functions for the cement clinker fCaO content online prediction model.

In addition, Liang et al. [35] developed the online sequential extreme learning machine (OS-ELM) algorithm, and the online sequential kernel extreme learning machine (OS-KELM) algorithm is proposed by Wang and Han [36]. These online learning algorithms have received widespread attention in relevant research, such as modeling time series [37], dynamic modeling [38], batch process [39], and discharge forecasting [40]. On account

of the large calculation of inverse matrix and the computational complexity of online sequential algorithm, the Cholesky factorization method is introduced to the extreme learning process [41,42]. However, the OS-ELM and OS-KELM algorithms need to compute the inverse of corresponding matrixes in each online sequential learning process. When the added data are too large, the matrix dimension in the network learning process of OS-ELM algorithm and OS-KELM algorithm is too large, which increases the computation of online sequential learning process and affects the learning speed of online training [43]. Therefore, in view of the limitations of the existing ELM algorithm, it is necessary to further study the neural network algorithm for online sequence learning to improve the structure of the existing neural network online learning.

In this research, considering different kernel function characteristics, three typical kernel functions are selected to construct an equivalent kernel function which is used in the MKELM algorithm. In order to avoid the large calculation of the inverse matrix and reduce the computational complexity of online sequential learning process, the LDL^T form Cholesky factorization based online sequential multiple kernel extreme learning machine (COS-MKELM) algorithm is proposed. In the COS-MKELM algorithm, there are two major differences: the LDL^T form Cholesky factorization of matrix is introduced, and the new matrix elements are calculated recursively based on the original matrix stored information. Then, three classical UCI regression datasets are used, which are available at <https://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html>, on 5 December 2020, to test the validity of COS-MKELM. In addition, the cement clinker fCaO content online prediction model is built by the COS-MKELM algorithm. Simulation results show that the online prediction model has good learning efficiency, regression accuracy, and generalization ability.

The rest of this paper is arranged as follows. In Section 2, the MKELM algorithm with three typical kernel functions is derived. The COS-MKELM algorithm is proposed and tested in Section 3. In Section 4, the cement clinker fCaO online prediction model is built and simulation verified. Finally, Section 5 concludes the paper.

2. The MKELM Algorithm

In this section, the MKELM algorithm is given to handle different heterogeneous data integrations. It can improve the model precision compared with single kernel function. The optimal kernel of MKELM is expressed as a linear combination of multiple different basic kernel functions.

Assuming the set of training data are x_i, y_i , where $i = 1, 2, \dots, N$, N is the number of samples. $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in \mathbf{R}^n$ denotes the input vector and $y_i = [y_{i1}, y_{i2}, \dots, y_{im}]^T \in \mathbf{R}^m$ is the output vector. n expresses the number of input layer nodes and m represents the number of output layer nodes. Then, the output function of MKELM can be given as follows:

$$f_L(x_i) = \sum_{k=1}^Q \sum_{p=1}^G \lambda_p \beta_{kp} g_p(\omega_k x_i + b_k) = \sum_{k=1}^Q \sum_{p=1}^G \lambda_p \beta_{kp} h_p(x_i) \quad (1)$$

where $\lambda_p \geq 0$ is a coefficient of the p th kernel function, $p = 1, 2, \dots, G$, G represents the number of base kernel functions; Q is the node number of hidden layer; $\omega_k = [\omega_{k1}, \omega_{k2}, \dots, \omega_{kl}, \dots, \omega_{km}]^T$ is the weight of connection between the k th node of hidden layer and each input node, $l = 1, 2, \dots, n$; The b_k sets threshold of the k th hidden node; $\beta_k = [\beta_{k1}, \beta_{k2}, \dots, \beta_{kq}, \dots, \beta_{km}]^T$ is the weight vector between hidden nodes and output nodes, $q = 1, 2, \dots, m$; $g(x)$ is the activation function.

According to the structural risk minimization principle, Equation (1) can be written as the minimum optimization problem:

$$\min_{\lambda} \min_{\beta, \xi} \frac{1}{2} \sum_{p=1}^G \lambda_p \|\beta_p\|^2 + \frac{1}{2} \gamma \sum_{i=1}^N \xi_i^2$$

$$\text{s.t.} \begin{cases} \sum_{p=1}^G h_p(x_i) \beta_p = y_i - \xi_i, i = 1, 2, \dots, N \\ \sum_{p=1}^G \frac{1}{\lambda_p} = 1, p = 1, 2, \dots, G \end{cases} \quad (2)$$

where γ is regularization coefficient, and ξ_i is the error between the i th actual output and expected output.

It is not difficult to verify that (2) is a joint-convex optimization problem, and the solution of (2) with Lagrangian method can be written as:

$$L(\beta, \xi, \alpha, \tau) = \frac{1}{2} \sum_{p=1}^G \lambda_p \|\beta_p\|^2 + \frac{\gamma}{2} \sum_{i=1}^N \xi_i^2$$

$$- \sum_{i=1}^N \alpha_i \left\{ \sum_{p=1}^G h_p(x_i) \beta_p + \xi_i - y_i \right\} - \sum_{i=1}^N \tau_i \left\{ \sum_{p=1}^G \frac{1}{\lambda_p} - 1 \right\} \quad (3)$$

where α and τ are the Lagrange multipliers.

According to the Karush–Kuhn–Tucker optimality condition, the partial differential of the Lagrangian function in (3) with respect to the variables (β , ξ , α and τ) can be calculated respectively as:

$$\begin{cases} \frac{\partial L}{\partial \beta} = \lambda_p \beta_p - \sum_{i=1}^N \alpha_i h_p(x_i) = 0 \\ \frac{\partial L}{\partial \xi} = \gamma \xi_i - \alpha_i = 0 \\ \frac{\partial L}{\partial \alpha} = \sum_{p=1}^G h_p(x_i) \beta_p + \xi_i - y_i = 0 \\ \frac{\partial L}{\partial \tau} = \sum_{p=1}^G \frac{1}{\lambda_p} - 1 = 0 \end{cases} \quad (4)$$

From the above formula, the following formula can be obtained:

$$\begin{cases} \beta_p = \frac{1}{\lambda_p} \sum_{i=1}^N \alpha_i h_p(x_i) \\ \xi_i = \frac{\alpha_i}{\gamma} \\ y_i = \sum_{p=1}^G h_p(x_i) \beta_p + \xi_i \\ \sum_{p=1}^G \frac{1}{\lambda_p} = 1 \end{cases} \quad (5)$$

Therefore, the Lagrange multiplier τ vanishes during the derivation process. Then, let β and ξ also vanish according to (5), and the following formula can be obtained:

$$f_L(x_i) = y_i = \sum_{p=1}^G \sum_{i=1}^N \frac{\alpha_i}{\lambda_p} h_p^T(x_i) h_p(x_i) + \frac{\alpha_i}{\gamma} \quad (6)$$

According to (4) and (5), the (6) can be rewritten into a matrix form as:

$$\left(\frac{\mathbf{I}}{\gamma} + \mathbf{\Omega}\right)\boldsymbol{\alpha} = \mathbf{Y} \quad (7)$$

where $\mathbf{\Omega}$ is the multiple kernel function matrix of MKELM, whose expression can be written as:

$$\mathbf{\Omega} = \sum_{p=1}^G \frac{1}{\lambda_p} h_p^T(\mathbf{x}_i) h_p(\mathbf{x}_i) = \sum_{p=1}^G \frac{1}{\lambda_p} K(\mathbf{x}, \mathbf{x}_i) \quad (8)$$

It is the multiple kernel functions matrix of the MKELM. From (7), the $\boldsymbol{\alpha}$ corresponding to the structural parameter of MKELM can be obtained by:

$$\boldsymbol{\alpha} = \left(\frac{\mathbf{I}}{\gamma} + \mathbf{\Omega}\right)^{-1} \mathbf{Y} \quad (9)$$

Considering the general characteristics of kernel functions and the model complexity, a new equivalent kernel is constructed, which uses the weighted combination of Polynomial kernel, Exponential Radial Basis kernel, and the Gaussian Radial Basis kernel:

Polynomial kernel:

$$\mathbf{K}_{\text{poly}} = [(x_i^T x) + C]^3, C > 0 \quad (10)$$

Exponential Radial Basis kernel:

$$\mathbf{K}_{\text{exp}} = \exp\left(-\|x - x_i\| / (2\sigma'^2)\right) \quad (11)$$

Gaussian Radial Basis kernel:

$$\mathbf{K}_{\text{RBF}} = \exp(-(\|x - x_i\|^2) / (2\sigma^2)) \quad (12)$$

where C , σ' and σ are parameters to be determined.

The equivalent kernel function of MKELM can be written as:

$$\mathbf{\Omega} = \frac{1}{\lambda_1} \mathbf{K}_{\text{poly}} + \frac{1}{\lambda_2} \mathbf{K}_{\text{exp}} + \frac{1}{\lambda_3} \mathbf{K}_{\text{RBF}} \quad (13)$$

where λ_1 , λ_2 , and λ_3 are the coefficients of corresponding kernel functions.

According to the constraint conditions of (2) and letting the coefficient λ_3 in (13) vanish, Equation (13) can be simplified as:

$$\mathbf{\Omega}_{\text{MKELM}} = \frac{1}{\lambda_1} \mathbf{K}_{\text{poly}} + \frac{1}{\lambda_2} \mathbf{K}_{\text{exp}} + \left(1 - \frac{1}{\lambda_1} - \frac{1}{\lambda_2}\right) \mathbf{K}_{\text{RBF}} \quad (14)$$

Therefore, the expression of the MKELM depicted in this research can be written as:

$$f_L(\mathbf{x}_i) = \sum_{i=1}^N \alpha_i \left\{ \frac{1}{\lambda_1} \cdot [(x_i^T x) + C]^3 + \frac{1}{\lambda_2} \cdot \exp\left(-\frac{\|x - x_i\|}{2\sigma'^2}\right) + \left(1 - \frac{1}{\lambda_1} - \frac{1}{\lambda_2}\right) \cdot \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) + \frac{1}{\gamma} \right\} \quad (15)$$

In summary, the parameters γ , C , σ' , σ , and λ in (15) can be determined empirically, and the parameter $\boldsymbol{\alpha}$ is obtained by matrix inversion calculation according to (9). Therefore, the main amount of calculation in MKELM algorithm is the solution of parameter $\boldsymbol{\alpha}$.

3. The Proposed COS-MKELM Algorithm and Performance Verification

It can be seen from the MKELM algorithm depicted in Section 2 that, if the modeling process is repeated with the new input training data, the parameter $\boldsymbol{\alpha}$ of MKELM needs

to be obtained by repeating the computation of the inverse matrix. In addition, when the new training data added into MKELM are too large, calculating the matrix inverse with a very large dimension will greatly increase the amount of calculation in MKELM online modeling. Thus, the LDL^T form Cholesky factorization method is used in this research to avoid the large matrix inverse calculation and simplify the online sequential calculation of MKELM. In this section, the COS-MKELM algorithm is proposed.

3.1. The Solution of COS-MKELM Parameter by Cholesky Factorization

According to (8) and (9), the expression of parameter α can be simplified as (16), where N is the number of initial training samples:

$$\alpha = \left(\mathbf{I}_N / \gamma + \sum_{p=1}^G \frac{1}{\lambda_p} \mathbf{H}_{p,N} \mathbf{H}_{p,N}^T \right)^{-1} \mathbf{Y}_N \quad (16)$$

Denote $\mathbf{E}_N = \mathbf{I}_N / \gamma + \sum_{p=1}^G \frac{1}{\lambda_p} \mathbf{H}_{p,N} \mathbf{H}_{p,N}^T$, and it can be obtained as $\mathbf{E}_N^T = \mathbf{E}_N$. Then, the (16) can be transformed as (17) when left multiplied matrix \mathbf{E}_N :

$$\mathbf{E}_N \alpha = \mathbf{Y}_N \quad (17)$$

The Cholesky factorization method can be used to solve the linear equation shown in (17). However, the conventional Cholesky factorization needs to be a square root, which is easy to reduce the computational precision and increase the computational complexity. In order to overcome this shortage, the matrix \mathbf{E}_N is factorized as the LDL^T form Cholesky factorization. In addition, the elements in the LDL^T forming the Cholesky factorization matrix are calculated recursively. Before getting the LDL^T form Cholesky factorization, it is necessary to prove that the matrix \mathbf{E}_N is a symmetric positive definite matrix. Assume that any column vector $\mathbf{W}_N = [W_1 \ W_2 \ \cdots \ W_N]^T \neq \mathbf{0}$ has the same dimension as \mathbf{E}_N , and the following formula can be obtained:

$$\begin{aligned} \mathbf{W}_N^T \mathbf{E}_N \mathbf{W}_N &= \mathbf{W}_N^T \left(\mathbf{I}_N / \gamma + \sum_{p=1}^G \frac{1}{\lambda_p} \mathbf{H}_{p,N} \mathbf{H}_{p,N}^T \right) \mathbf{W}_N \\ &= \frac{1}{\gamma} \mathbf{W}_N^T \mathbf{W}_N + \sum_{p=1}^G \frac{1}{\lambda_p} \mathbf{W}_N^T \mathbf{H}_{p,N} \mathbf{H}_{p,N}^T \mathbf{W}_N \\ &= \frac{1}{\gamma} [\mathbf{W}_N, \mathbf{W}_N] + \sum_{p=1}^G \frac{1}{\lambda_p} [\mathbf{H}_{p,N}^T \mathbf{W}_N, \mathbf{H}_{p,N}^T \mathbf{W}_N] \\ &= \frac{1}{\gamma} \|\mathbf{W}_N\|^2 + \sum_{p=1}^G \frac{1}{\lambda_p} \|\mathbf{H}_{p,N}^T \mathbf{W}_N\|^2 > 0 \end{aligned} \quad (18)$$

Equation (18) proves that the matrix \mathbf{E}_N is a symmetric positive definite matrix. Then, \mathbf{E}_N can be factorized to LDL^T form as follows:

$$\mathbf{E}_N = \mathbf{L}_N \mathbf{D}_N \mathbf{L}_N^T \quad (19)$$

where \mathbf{L}_N is a unit lower triangular matrix and \mathbf{D}_N is a diagonal matrix with diagonal positive.

From (19), (20) can be obtained:

$$e_{ij} = \sum_{k=1}^{j-1} l_{ik} d_k l_{jk} + l_{ij} d_j \quad (20)$$

where e , l , and d are the elements of matrix \mathbf{E}_N , \mathbf{L}_N and \mathbf{D}_N , $1 \leq j \leq i \leq N$.

Thus, the elements of L_N and D_N can be calculated recursively as follows:

$$\begin{cases} d_j = e_{jj} - \sum_{k=1}^{j-1} l_{jk}^2 d_k \\ l_{ij} = \left(e_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk} d_k \right) / d_j \end{cases} \tag{21}$$

where $k = 1, 2, \dots, j - 1; i = j + 1, j + 2, \dots, N; j = 1, 2, \dots, N$.

Put (19) into (17) and multiply L_N^{-1} on both sides of (17):

$$D_N L_N^T \alpha = L_N^{-1} Y_N = M_N \tag{22}$$

where M_N is a column vector.

The elements m_i in matrix M_N can be obtained as follows:

$$m_i = \begin{cases} y_i / l_{ii} & i = 1 \\ \left(y_i - \sum_{j=1}^{i-1} l_{ij} m_j \right) / l_{ii} & i > 1 \end{cases} \tag{23}$$

where $i = 1, 2, \dots, N$.

According to (22), the elements α_i in the parameter α can be calculated recursively as follows:

$$\alpha_i = \begin{cases} m_i / d_i & i = N \\ m_i / d_i - \sum_{j=i+1}^N l_{ij} \alpha_j & i < N \end{cases} \tag{24}$$

Therefore, in the solution of the COS-MKELM parameter calculation process, the LDL^T form Cholesky factorization method avoids the large matrix inverse and extracting a root calculation.

3.2. Online Sequential Learning COS-MKELM Parameter

During the online sequential learning process of the COS-MKELM algorithm, the new training data, which were added into the machine, will be learned immediately. Assuming that the number of new training data is \hat{N} , then the matrix $E_{N+\hat{N}}$ can be written as follows:

$$\begin{aligned} E_{N+\hat{N}} &= I_{N+\hat{N}} / \gamma + \sum_{p=1}^G \frac{1}{\lambda_p} \mathbf{H}_{p,N+\hat{N}} \mathbf{H}_{p,N+\hat{N}}^T \\ &= I_{N+\hat{N}} / \gamma + \sum_{p=1}^G \frac{1}{\lambda_p} \begin{bmatrix} \mathbf{H}_{p,N} \\ \mathbf{H}_{p,\hat{N}} \end{bmatrix} \begin{bmatrix} \mathbf{H}_{p,N} \\ \mathbf{H}_{p,\hat{N}} \end{bmatrix}^T \end{aligned} \tag{25}$$

According to E_N described in Section 3.1, (25) can be expanded as follows:

$$\begin{aligned} E_{N+\hat{N}} &= \begin{bmatrix} E_N & E_N^1 \\ E_N^2 & E_N^3 \end{bmatrix} \\ E_N^1 &= \sum_{p=1}^G \frac{1}{\lambda_p} \begin{pmatrix} \mathbf{H}_{p,1} \mathbf{H}_{p,\hat{N}}^T \\ \vdots \\ \mathbf{H}_{p,N} \mathbf{H}_{p,\hat{N}}^T \end{pmatrix} \\ E_N^2 &= \sum_{p=1}^G \frac{1}{\lambda_p} \left(\mathbf{H}_{p,\hat{N}} \mathbf{H}_{p,1}^T, \dots, \mathbf{H}_{p,\hat{N}} \mathbf{H}_{p,N}^T \right) \\ E_N^3 &= I_{N+\hat{N}} / \gamma + \sum_{p=1}^G \frac{1}{\lambda_p} \mathbf{H}_{p,\hat{N}} \mathbf{H}_{p,\hat{N}}^T \end{aligned} \tag{26}$$

The matrix $E_{N+\hat{N}}$ can be factorized in the LDL^T form as follows:

$$E_{N+\hat{N}} = L_{N+\hat{N}} D_{N+\hat{N}} L_{N+\hat{N}}^T \quad (27)$$

It can be obtained from (19) as follows:

$$\hat{e}_{ij} = \sum_{k=1}^{j-1} \hat{l}_{ik} \hat{d}_k \hat{l}_{jk} + \hat{l}_{ij} \hat{d}_j \quad (28)$$

Here, \hat{e} , \hat{l} , and \hat{d} are the elements of matrix $E_{N+\hat{N}}$, $L_{N+\hat{N}}$ and $D_{N+\hat{N}}$, $1 \leq j \leq i \leq N + \hat{N}$.

Form (26) and (27), it can be seen that, when $1 \leq j \leq i \leq N$, $L_{N+\hat{N}}$ and $D_{N+\hat{N}}$ are the same to L_N and D_N . Only the remaining new training data (the \hat{N} new training data) need to be calculated recursively according to (21). Thus, the elements of matrix $L_{N+\hat{N}}$ and $D_{N+\hat{N}}$ can be calculated recursively based on matrix L_N and D_N .

Similarly, the matrix $M_{N+\hat{N}}$ can be obtained according to (23). It is shown as (29):

$$M_{N+\hat{N}} = \begin{bmatrix} M_N \\ m_{\hat{N}} \end{bmatrix} \quad (29)$$

The elements α_i in the new parameter α of COS-MKELM can be obtained according to (24). Substituting the parameter α into (15), the output of COS-MKELM algorithm can be obtained. In summary, when the new training data are added into COS-MKELM, the new matrix elements are calculated recursively based on the original matrix information. Thus, the COS-MKELM algorithm has reduced the computational complexity and simplified the online sequential learning process.

3.3. COS-MKELM Performance Verification

In this subsection, the regression datasets of Auto MPG (d_1), Machine CPU (d_2), and Boston Housing (d_3) are used to test the performance of COS-MKELM. These regression datasets have been widely used to test the performance of the modeling method. All of the experiments are carried out under Windows 7 and Matlab R2010a, with 4 GB RAM, and an Inter Pentium G2020 CPU with 2.90 GHz.

The COS-MKELM algorithm is compared with the OS-ELM algorithm and the OS-KELM algorithm. The activation function of OS-ELM algorithm is Sigmoid function, and the number of hidden layer nodes of OS-ELM algorithm is 40. The kernel function of OS-KELM is selected as the Gaussian Radial Basis kernel, and the kernel function of the proposed COS-MKELM algorithm is the same as the kernel function presented in Section 2. The parameters of COS-MKELM algorithm are set as the following: $C = 60$, $\sigma' = 2.5$, $\sigma = 8$, $\lambda_1 = 10$, $\lambda_2 = 2.5$, $\gamma = 10,000$. 3/4 of each regression datasets is used as training data, and the remaining 1/4 is used as testing data. The number of initial training data of OS-ELM algorithm is set as 50, and the rest of the training data are added in 10%. In addition, 10% of the training data are added in OS-KELM and COS-MKELM each time.

In this research, the root mean square error (RMSE) is used to evaluate the precision performance index. In addition, RMSE can be expressed as follows:

$$RMSE = \sqrt{\frac{1}{N_T} \sum_{i=1}^{N_T} (y_i - \hat{y}_i)^2} \quad (30)$$

where y_i and \hat{y}_i stand for the actual value and estimated value, and N_T is the number of testing data.

The training time and RMSE for training data are listed in Table 1. In addition, the output values of each regression datasets for testing data are shown in Figures 1–3.

Table 1. Performance comparison of the three learning algorithms on UCI regression datasets.

	Dataset	OS-ELM	OS-KELM	COS-MKELM
d_1	Training time/s	0.8217	1.0545	0.7145
	RMSE	2.54	2.19	1.73
d_2	Training time/s	0.6529	0.9015	0.5648
	RMSE	23.15	21.17	16.58
d_3	Training time/s	0.9753	1.2045	0.8210
	RMSE	3.82	3.65	2.97

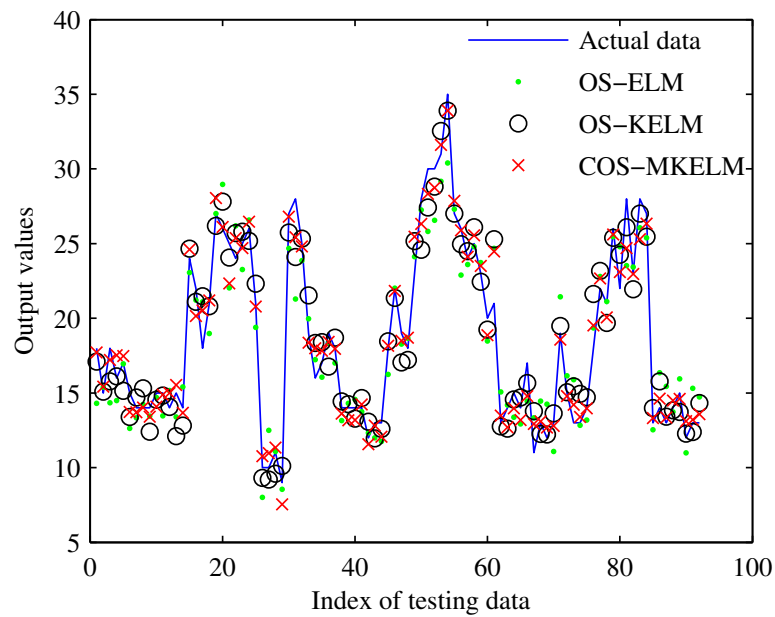


Figure 1. Output values for testing data of d_1 .

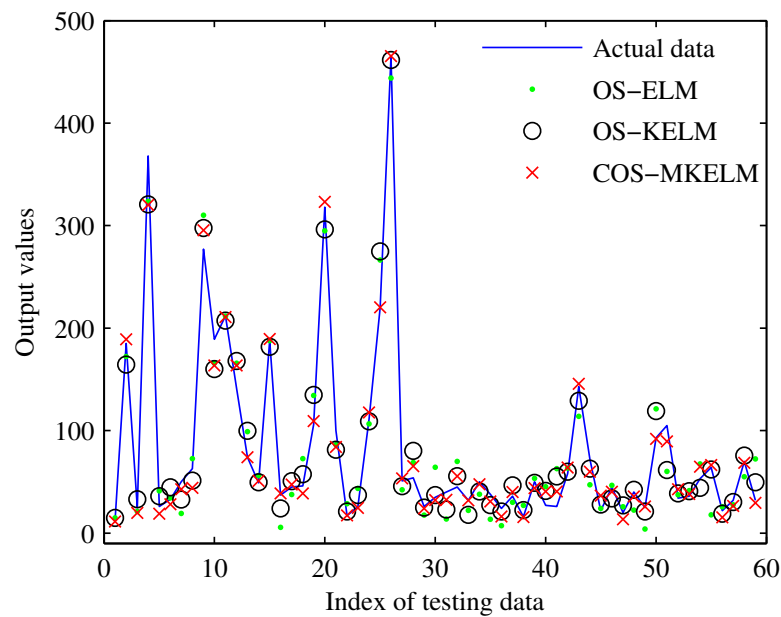


Figure 2. Output values for testing data of d_2 .

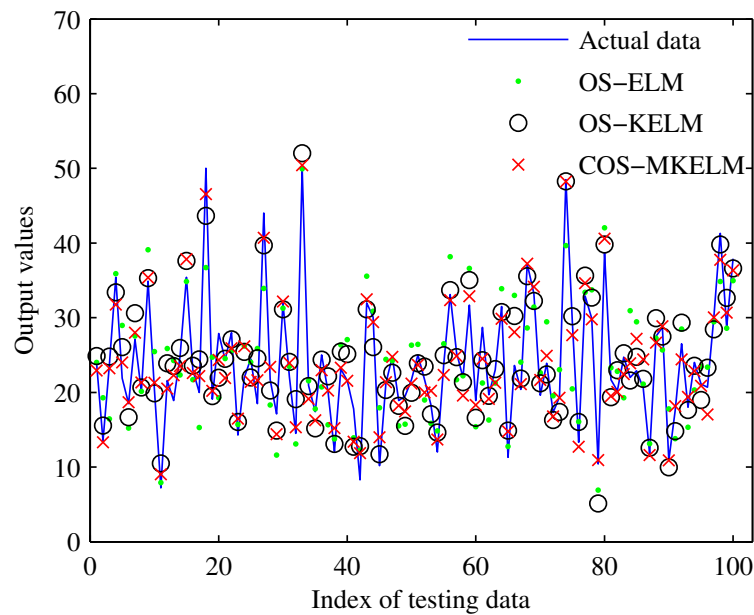


Figure 3. Output values for testing data of d_3 .

From Table 1, it can be seen that, in the COS-MKELM algorithm, the training times of the three regression datasets are all the smallest. It demonstrates that the COS-MKELM algorithm is better than the other comparison algorithms in terms of learning efficiency. As seen from the RMSE values for testing data listed in Table 1 and the output results shown in Figures 1–3, the predicted testing values based on COS-MKELM are closer to the actual values than the other comparison algorithms. It is concluded that the COS-MKELM algorithm proposed in this research shows much better performance in regression accuracy and generalization ability than the OS-ELM and OS-KELM.

Therefore, it can be seen that it is feasible to build a cement clinker fCaO online prediction model based on the COS-MKELM algorithm proposed in this paper.

4. Cement Clinker fCaO Model and Simulation

4.1. fCaO Content Online Prediction Model

In this subsection, the proposed COS-MKELM algorithm is adopted to build the cement clinker fCaO content online prediction model. The flowchart of the building cement clinker fCaO content online prediction model is shown in Figure 4.

Combined with the COS-MKELM algorithm, the fCaO online prediction modeling steps can be described in detail below:

Step 1: Collect the dataset of variables related to the fCaO content from the cement production line. Given the initial training data $\{x_i, y_i\}_{i=1}^N$. Initialize the values of $\gamma, C, \sigma', \sigma, \lambda_1$ and λ_2 .

Step 2: Factorize the matrix E_N to LDL^T form. Calculate the elements of L_N, D_N , and M_N recursively according to (21) and (23).

Step 3: Calculate the parameter α of COS-MKELM according to (24), and put it into (15) to obtain the online prediction output value of fCaO content.

Step 4: If the new training data are added into the COS-MKELM, update the number of online learning steps $k = k + 1$ and run **Step 5** to learn the new data; otherwise, output the parameter values ($\alpha, \gamma, C, \sigma', \sigma, \lambda_1$ and λ_2), and then obtain the fCaO content online prediction model.

Step 5: Assume the new training data added into COS-MKELM is $\{x_i, y_i\}_{i=1}^{\hat{N}}$. Based on the matrix L_k, D_k and M_k obtained from the k th step of online sequential learning, calculate the elements of matrix L_{k+1}, D_{k+1} and M_{k+1} recursively according to (21) and (23). Return **Step 3**.

After the cement clinker fCaO content online prediction model is obtained, in the process of fCaO content online prediction, the parameters of fCaO model are updated constantly. When new sample data are obtained or the production conditions of cement clinker change, the above modeling **Step 4** is repeated on the basis of the existing online prediction model. The cement clinker fCaO content online prediction model is trained continuously, and relevant parameters of the model are updated. The online process of updating parameters makes the model have strong applicability and anti-interference ability.

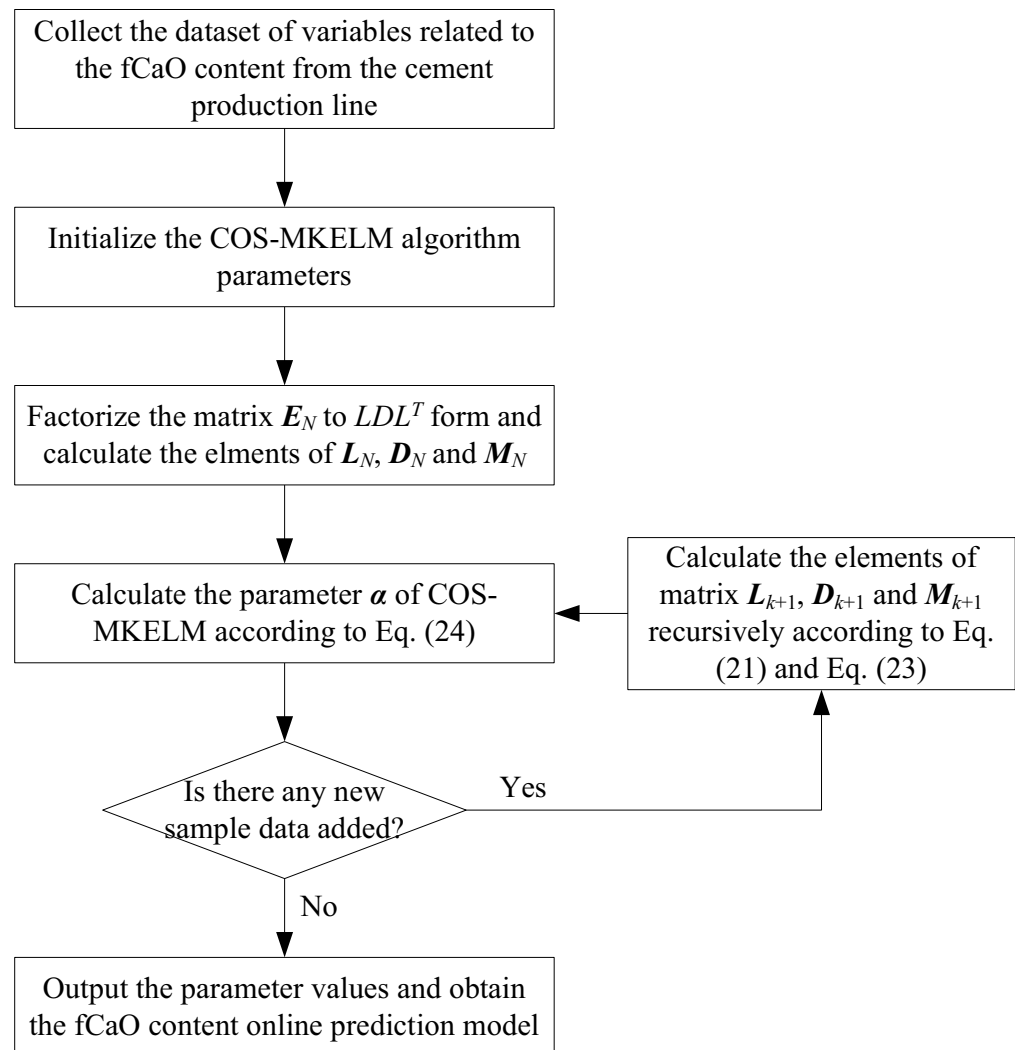


Figure 4. Flowchart of the fCaO online prediction model with the COS-MKELM algorithm.

4.2. Dataset and Ascertaining Model Parameters

In this research, the raw dataset includes simultaneous process variables and fCaO content measured values under various conditions for fCaO online prediction modeling. It is obtained from No.1 rotary kiln at the TangshanJidong cement plant with 5000 tons clinker production capacity per day. Through object linking and embedding for process control communication technology, the input variables data of cement clinker fCaO content prediction model were collected from the distributed control system of the cement clinker production line. The laboratory samples the clinker on site every hour for chemical testing to obtain the actual value of the cement clinker fCaO content, which are the output variable data of the cement clinker fCaO content prediction model. Thus, the data of cement clinker fCaO content prediction model are sampled once every hour, with 24 samples per day.

Because the collected data are affected by uncertain factors such as noise, the continuous mean filtering method is used to pre-process the collected initial data. After filtering the

collected initial data, a total of 215 pairs of data are taken as the training and testing data sets of the cement clinker fCaO content prediction model. From these data, 150 data are selected randomly as the training dataset (used for fCaO content online prediction model development), and the other 65 data are testing datasets (used for fCaO content online prediction model validation). Some of the collected process variables are shown in Table 2.

Table 2. Some collected process variables data samples.

Variables	1	2	3	...	214	215
W_r /(t/h)	390	390	388	...	380	387
W_{cc} /(t/h)	7.4	7.8	8.3	...	7.7	8.6
W_{kc} /(t/h)	11.9	12.1	12.8	...	13.1	12.7
S_f /(r/min)	815	824	829	...	837	841
T_c /°C	890	894	892	...	897	886
T_{kt} /°C	997	993	1002	...	1007	992
T_{kh} /°C	1018	1023	1014	...	1035	1024
C_n /ppm	868	878	984	...	682	719
I_k /A	886	884	849	...	812	919
KH /%	0.94	0.95	0.92	...	0.99	0.98
SM /%	2.71	2.74	2.73	...	2.74	2.76
AM /%	1.10	1.16	1.17	...	1.37	1.35
fCaO/%	1.33	0.78	0.31	...	0.63	0.90

W_r = weight of raw material; W_{cc} = weight of calciner coal; W_{kc} = weight of kiln coal; S_f = induced draft fan speed; T_c = temperature of calciner exit gas; T_{kt} = temperature of kiln tail; T_{kh} = temperature of kiln head; C_n = content of smoke chamber NOx; I_k = kiln main motor current; KH = lime saturation factor; SM = silica modulus; AM = alumina-oxygen modulus. The fCaO content would be taken as the output of online prediction model, and the above 12 variables are chosen as the inputs of online prediction model.

In order to prove the validity of the fCaO online prediction model based on COS-MKELM, OS-ELM and OS-KELM are used as comparative algorithms in this research. The parameters of the three models are set to the same as the parameters in Section 3.3.

4.3. Results and Discussion

In this subsection, the RMSE, mean absolute percentage error (MAPE), mean absolute error (MAE), maximum absolute error (MAXE), and coefficient of determination (R^2) are used as performance assessment indexes for the fCaO models' prediction. The expressions of RMSE are the same as (30), and the other criteria expressions are given as follows:

$$MAPE = \frac{1}{N_T} \sum_{i=1}^{N_T} |(y_i - \hat{y}_i) / y_i| \times 100\% \quad (31)$$

$$MAE = \frac{1}{N_T} \sum_{i=1}^{N_T} |y_i - \hat{y}_i| \quad (32)$$

$$MAXE = \max |y_i - \hat{y}_i| \quad (33)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N_T} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N_T} (y_i - \bar{y})^2}, \quad \bar{y} = \frac{1}{N_T} \sum_{i=1}^{N_T} y_i \quad (34)$$

The four performance criteria: RMSE, MAPE, MAE, and MAXE are applied to estimate the accuracy performance of the models. R^2 denotes the degree of matching between the estimated and actual values. The smaller the values of RMSE, MAPE, MAE and MAXE are and the closer R^2 is to 1, the better the accuracy performance of a model.

The performance comparison results are shown in Table 3. Compared with actual data, the simulation results of the fCaO online prediction model with three models for testing data are shown in Figures 5–7.

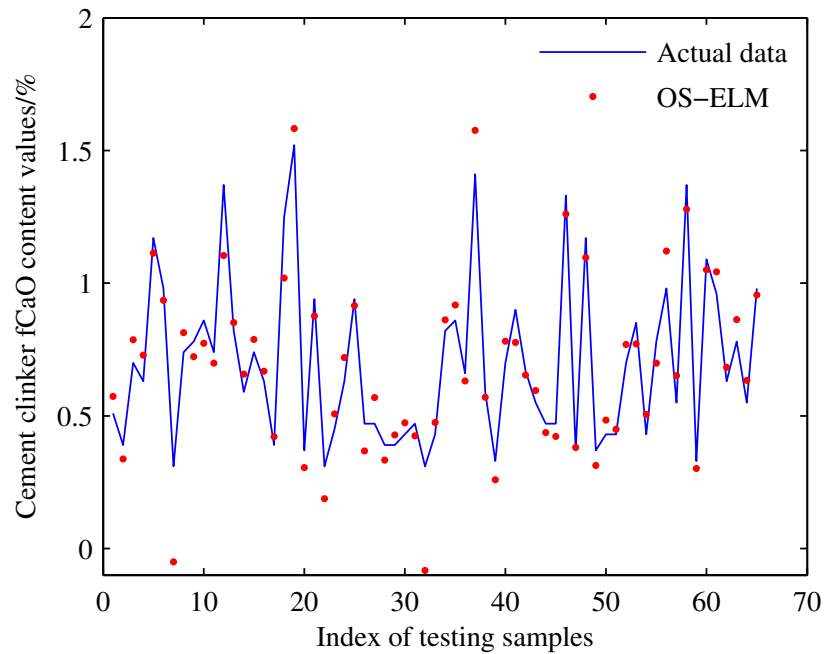


Figure 5. Simulation result of fCaO online prediction model with OS-ELM for testing data.

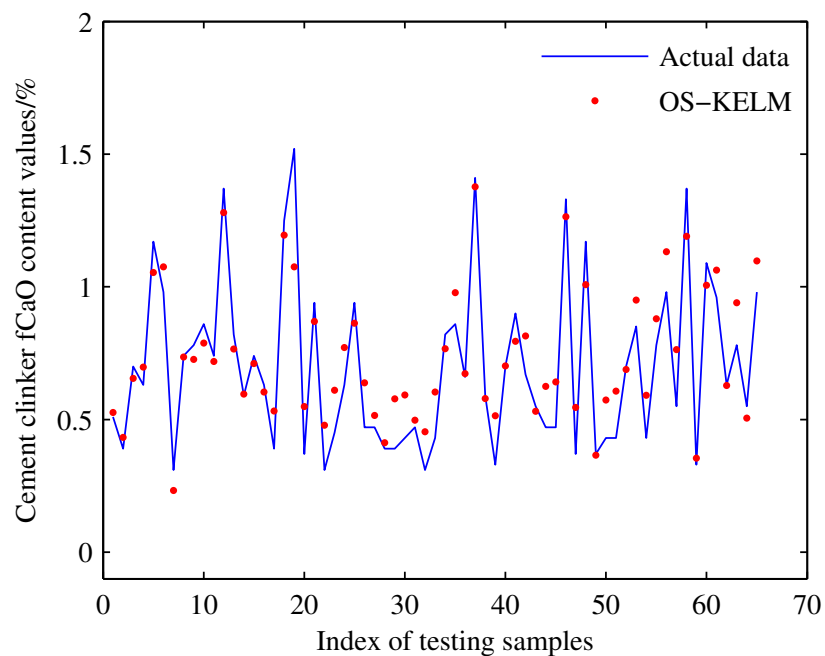


Figure 6. Simulation result of fCaO online prediction model with OS-KELM for testing data.

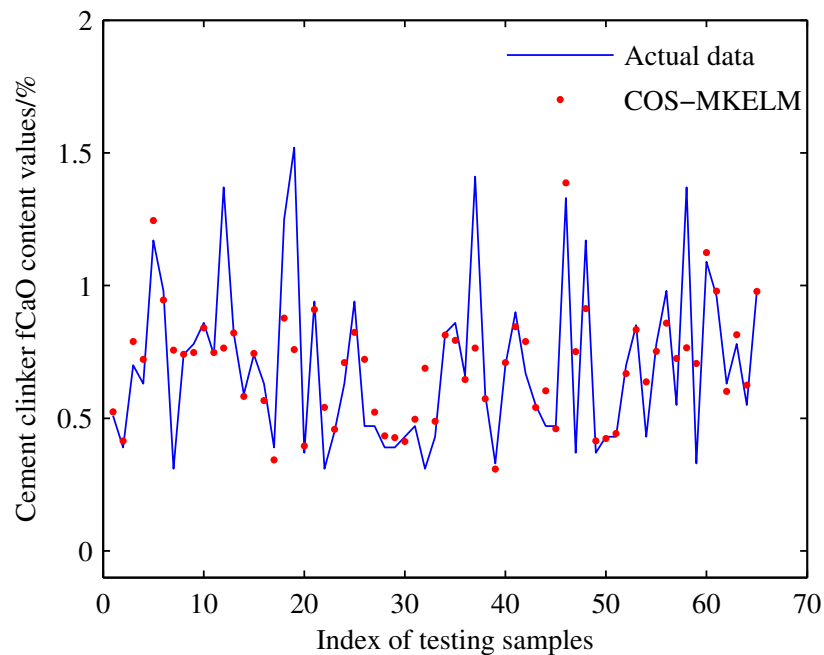


Figure 7. Simulation result of fCaO online prediction model with COS-MKELM for testing data.

Table 3. Performance comparison of the fCaO content online prediction models.

Models	Training Time/s	Testing Time/s	RMSE	MAPE/%	MAE	MAXE	R^2
OS-ELM	0.5327	0.0063	0.3182	20.46	0.1537	1.6725	0.9685
OS-KELM	0.7286	0.0152	0.1359	17.58	0.1257	1.2547	0.9729
COS-MKELM	0.3856	0.0041	0.1018	9.54	0.0758	1.0174	0.9798

As seen from the time data in Table 3, the training time and testing time of fCaO model based on COS-MKELM are smallest. The RMSE, MAPE, MAE, and MAXE performance criteria values of fCaO model based on COS-MKELM are the least, and the criterion R^2 value is the closest to 1 among the three comparison models. As seen from the precision index data in Table 3, the predicted values of the cement clinker fCaO content online prediction model based on the COS-MKELM algorithm are closer to the actual value of cement clinker fCaO content obtained by the laboratory test. In addition, the established cement clinker fCaO content online prediction model has strong nonlinear identification ability, which meets the requirements of prediction accuracy of the fCaO content in the cement clinker firing process.

The simulation results in Figures 5–7 show that the predicted fCaO content values based on the COS-MKELM algorithm are closer to the actual values than the other comparison algorithms. In conclusion, although the COS-MKELM algorithm has fewer initial samples in the simulation modeling process, the learning performance of COS-MKELM algorithm is improved greatly through the continuous arrival of sample data. The online prediction model of cement clinker fCaO content based on the proposed COS-MKELM algorithm shows better performance in both test accuracy and generalization ability, indicating that the cement clinker fCaO content online prediction model based on the proposed COS-MKELM algorithm is effective.

In short, the simulation results of the cement clinker fCaO content online prediction model using actual data show that the proposed COS-MKELM algorithm, which is based on the fusion of three kernel functions method and online sequential learning method, shows fast learning efficiency. In addition, the prediction model of cement clinker fCaO content has better nonlinear identification ability, regression accuracy, and generalization ability.

Therefore, the fCaO content value can be predicted by the fCaO content online prediction model based on the proposed COS-MKELM algorithm. When the new training data added or the production conditions of cement clinker changed during the online prediction of fCaO content, the parameters of fCaO model can be online sequential updated to adapt the current condition. With the continuous input of actual data, the characteristics of cement clinker fCaO content online prediction model are improved, so that the online prediction model can adapt to different production conditions and operate stably for a long time. Therefore, this paper provides a new idea for realizing the prediction of cement clinker fCaO content. Furthermore, according to the fCaO content predicted value, the operators and engineers can adjust the cement production process variables. In this way, the qualified rate of cement clinker can be guaranteed, and the energy consumption and the exhaust emission of cement clinker production can be reduced.

5. Conclusions

In this paper, the COS-MKELM algorithm is proposed to avoid the large matrix inverse calculation and reduce the computational complexity of online sequential learning process. Three regression datasets were used to test the performance of the COS-MKELM algorithm. Experiment results show that the proposed COS-MKELM algorithm, which is based on the fusion of the three kernel functions method and the online sequential learning method, shows fast learning efficiency. Although the COS-MKELM algorithm has fewer initial samples in the simulation modeling process, the learning performance of COS-MKELM algorithm is improved greatly through the continuous arrival of sample data.

In addition, online prediction fCaO content is crucial and useful to the cement process industry. The proposed COS-MKELM algorithm is adopted to build the cement clinker fCaO content online prediction model of rotary kiln with 5000 tons of clinker production capacity per day in this paper. It can be seen from experiment results that the online prediction model of cement clinker fCaO content based on the proposed COS-MKELM algorithm shows better performance in nonlinear identification ability, regression accuracy, and generalization ability. The characteristics of cement clinker fCaO content online prediction model can be improved with the iteration update of cement calcination process data. Therefore, this research provides a new idea for realizing the prediction of cement clinker fCaO content.

Based on the online prediction model proposed in this research, the cement clinker fCaO content can be mastered by operators and engineers in real time. This is beneficial to the clinker quality and the energy consumption of the cement calcination process. Furthermore, the fCaO online sequential model built in this paper provides the necessary prerequisite for the cement process industry to build the intelligent control system and minimize energy costs.

Author Contributions: Conceptualization, P.Z. and Y.C.; methodology, Z.Z. and P.Z.; software, P.Z.; validation, Y.C. and Z.Z.; data curation, P.Z. and Y.C.; writing—original draft preparation, P.Z.; writing—review and editing, Y.C. and Z.Z.; funding acquisition, P.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Science Technology Research and Development Plan Foundation of Shijiazhuang under Grant 191130701, in part by the Doctoral Research Start-up Foundation of Shijiazhuang University under Grant 20BS021, and in part by the Science Technology Research and Development Plan Foundation of Tianjin under Grant 2020KJ118.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

fCaO	The content of cement clinker free lime
ELM	Extreme Learning Machine
KELM	Kernel Extreme Learning Machine
MKELM	Multiple Kernel Extreme Learning Machine
OS-ELM	Online Sequential Extreme Learning Machine
OS-KELM	Online Sequential Kernel Extreme Learning Machine
COS-MKELM	Cholesky Factorization based Online Sequential Multiple Kernel Extreme Learning Machine

References

- Zhang, C.Y.; Yu, B.; Chen, J.M.; Wei, Y.M. Green transition pathways for cement industry in China. *Resour. Conserv. Recycl.* **2021**, *166*, 105355. [\[CrossRef\]](#)
- Shen, W.; Liu, Y.; Yan, B.; Wang, J.; He, P.; Zhou, C.; Huo, X.; Zhang, W.; Xu, G.; Ding, Q. Cement industry of China: Driving force, environment impact and sustainable development. *Renew. Sustain. Energy Rev.* **2017**, *75*, 618–628. [\[CrossRef\]](#)
- Nawaz, A.; Julnipitawong, P.; Krammart, P.; Tangtermsirikul, S. Effect and limitation of free lime content in cement-fly ash mixtures. *Constr. Build. Mater.* **2016**, *102*, 515–530. [\[CrossRef\]](#)
- Hao, X.; Gao, Y.; Yang, X.; Wang, J. Multi-objective collaborative optimization in cement calcination process: A time domain rolling optimization method based on Jaya algorithm. *J. Process Control* **2021**, *105*, 117–128. [\[CrossRef\]](#)
- Aghdasinia, H.; Hosseini, S.S.; Hamed, J. Improvement of a cement rotary kiln performance using artificial neural network. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *12*, 7765–7776. [\[CrossRef\]](#)
- Kaewmanee, K.; Krammart, P.; Sumranwanich, T.; Choktaweekarn, P.; Tangtermsirikul, S. Effect of free lime content on properties of cement-fly ash mixtures. *Constr. Build. Mater.* **2013**, *38*, 829–836. [\[CrossRef\]](#)
- Vogelbacher, M.; Waibel, P.; Matthes, J.; Keller, H.B. Image-based characterization of alternative fuel combustion with multifuel burners. *IEEE Trans. Ind. Inform.* **2017**, *14*, 588–597. [\[CrossRef\]](#)
- Li, W.; Wang, D.; Zhou, X.; Chai, T. An improved multi-source based soft sensor for measuring cement free lime content. *Inform. Sci.* **2015**, *323*, 94–105. [\[CrossRef\]](#)
- Szatvanyi, G.; Duchesne, C.; Bartolacci, G. Multivariate image analysis of flames for product quality and combustion control in rotary kilns. *Ind. Eng. Chem. Res.* **2006**, *45*, 4706–4715. [\[CrossRef\]](#)
- Li, W.; Wang, D.; Chai, T. Multisource data ensemble modeling for clinker free lime content estimate in rotary kiln sintering processes. *IEEE Trans. Syst. Man Cybern. Syst.* **2014**, *45*, 303–314.
- Liu, X.; Jin, J.; Wu, W.; Herz, F. A novel support vector machine ensemble model for estimation of free lime content in cement clinkers. *ISA Trans.* **2020**, *99*, 479–487. [\[CrossRef\]](#)
- Wu, W.; Liu, X.; Xu, X.; Jin, J.; Zhang, M. Time series analysis method for the soft measurement of cement clinker quality. *Control Theory Appl.* **2018**, *35*, 1029–1036.
- Zhao, P.; Zhao, Z.; Chen, Y. An improved combination modeling method for free lime content in cement clinker. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020; pp. 4400–4405.
- Hao, X.; Wang, Z.; Shan, Z.; Zhao, Y. Prediction of electricity consumption in cement production: A time-varying delay deep belief network prediction method. *Neural Comput. Appl.* **2019**, *31*, 7165–7179. [\[CrossRef\]](#)
- Shi, X.; Huang, G.; Hao, X.; Yang, Y.; Li, Z. A Synchronous Prediction Model Based on Multi-Channel CNN with Moving Window for Coal and Electricity Consumption in Cement Calcination Process. *Sensors* **2021**, *21*, 4284. [\[CrossRef\]](#) [\[PubMed\]](#)
- Hao, X.; Guo, T.; Huang, G.; Shi, X.; Zhao, Y.; Yang, Y. Energy consumption prediction in cement calcination process: A method of deep belief network with sliding window. *Energy* **2020**, *207*, 118256. [\[CrossRef\]](#)
- Liu, G.; Ouyang, Z.; Hao, X.; Shi, X.; Zheng, L.; Zhao, Y. Prediction of raw meal fineness in the grinding process of cement raw material: A two-dimensional convolutional neural network prediction method. *Proc. Inst. Mech. Eng. Part I* **2021**, *235*, 823–838.
- Pani, A.K.; Mohanta, H.K. Online monitoring of cement clinker quality using multivariate statistics and Takagi-Sugeno fuzzy-inference technique. *Control Eng. Pract.* **2016**, *57*, 1–17. [\[CrossRef\]](#)
- Pani, A.K.; Vadlamudi, V.K.; Mohanta, H.K. Development and comparison of neural network based soft sensors for online estimation of cement clinker quality. *ISA Trans.* **2013**, *52*, 19–29. [\[CrossRef\]](#)
- Zhao, Y.; Ding, B.; Zhang, Y.; Yang, L.; Hao, X. Online cement clinker quality monitoring: A soft sensor model based on multivariate time series analysis and CNN. *ISA Trans.* **2021**. [\[CrossRef\]](#)
- Lu, S.; Yu, H.; Wang, X.; Ning, F.; Zhao, P.; Zhang, R. Online Prediction Method of Cement Clinker f-Cao Based on K-ELM. In Proceedings of the 2018 Chinese Automation Congress (CAC), Xi'an, China, 30 November–2 December 2018; pp. 3355–3358.
- Huang, G.B.; Chen, L.; Siew, C.K. Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Trans. Neur. Net.* **2006**, *17*, 879–892. [\[CrossRef\]](#)
- Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: Theory and applications. *Neurocomputing* **2006**, *70*, 489–501. [\[CrossRef\]](#)

24. Huang, G.B.; Zhou, H.; Ding, X.; Zhang, R. Extreme learning machine for regression and multiclass classification. *IEEE Trans. Syst. Man Cybern. Part B* **2011**, *42*, 513–529.
25. Zhang, L.; Li, J.; Lu, H. Saliency detection via extreme learning machine. *Neurocomputing* **2016**, *218*, 103–112. [[CrossRef](#)]
26. Yavşan, E.; Uçar, A. Gesture imitation and recognition using Kinect sensor and extreme learning machines. *Measurement* **2016**, *94*, 852–861. [[CrossRef](#)]
27. Cao, J.; Zhang, K.; Luo, M.; Yin, C.; Lai, X. Extreme learning machine and adaptive sparse representation for image classification. *Neural Netw.* **2016**, *81*, 91–102. [[CrossRef](#)]
28. Shao, L.; Kang, R.; Yi, W.; Zhang, H. An Enhanced Unsupervised Extreme Learning Machine Based Method for the Nonlinear Fault Detection. *IEEE Access* **2021**, *9*, 48884–48898. [[CrossRef](#)]
29. Zhang, J.; Fan, Y.; Zhang, L.; Xu, C.; Dong, X.; Liu, L.; Zhang, Z.; Qiu, X. Seepage Time Soft Sensor Model of Nonwoven Fabric Based on the Extreme Learning Machine Integrating Monte Carlo. *Sensors* **2021**, *21*, 2377. [[CrossRef](#)] [[PubMed](#)]
30. Luo, H.; He, C.; Zhou, J.; Zhang, L. Rolling Bearing Sub-Health Recognition via Extreme Learning Machine Based on Deep Belief Network Optimized by Improved Fireworks. *IEEE Access* **2021**, *9*, 42013–42026. [[CrossRef](#)]
31. Liu, X.; Wang, L.; Huang, G.B.; Zhang, J.; Yin, J. Multiple kernel extreme learning machine. *Neurocomputing* **2015**, *149*, 253–264. [[CrossRef](#)]
32. Fossaceca, J.M.; Mazzuchi, T.A.; Sarkani, S. MARK-ELM: Application of a novel Multiple Kernel Learning framework for improving the robustness of Network Intrusion Detection. *Expert Syst. Appl.* **2015**, *42*, 4062–4080. [[CrossRef](#)]
33. Wang, Z.; Zeng, Y.; Liu, Y.; Li, D. Deep Belief Network Integrating Improved Kernel-based Extreme Learning Machine for Network Intrusion Detection. *IEEE Access* **2021**, *9*, 16062–16091. [[CrossRef](#)]
34. Wu, D.; Wang, Z.; Chen, Y.; Zhao, H. Mixed-kernel based weighted extreme learning machine for inertial sensor based human activity recognition with imbalanced dataset. *Neurocomputing* **2016**, *190*, 35–49. [[CrossRef](#)]
35. Liang, N.Y.; Huang, G.B.; Saratchandran, P.; Sundararajan, N. A fast and accurate online sequential learning algorithm for feedforward networks. *IEEE Trans. Neur. Net.* **2006**, *17*, 1411–1423. [[CrossRef](#)] [[PubMed](#)]
36. Wang, X.; Han, M. Online sequential extreme learning machine with kernels for nonstationary time series prediction. *Neurocomputing* **2014**, *145*, 90–97. [[CrossRef](#)]
37. Deng, W.Y.; Ong, Y.S.; Tan, P.S.; Zheng, Q.H. Online sequential reduced kernel extreme learning machine. *Neurocomputing* **2016**, *174*, 72–84. [[CrossRef](#)]
38. Zhang, J.; Li, Y.; Xiao, W. Adaptive online sequential extreme learning machine for dynamic modeling. *Soft Comput.* **2021**, *25*, 2177–2189. [[CrossRef](#)]
39. Xiao, D.; Wang, J.; Mao, Z.Z. The research on the modeling method of batch process based on OS-ELM-RMPLS. *Chemometr. Intell. Lab.* **2014**, *134*, 118–122. [[CrossRef](#)]
40. Yadav, B.; Ch, S.; Mathur, S.; Adamowski, J. Discharge forecasting using an online sequential extreme learning machine (OS-ELM) model: A case study in Neckar River, Germany. *Measurement* **2016**, *92*, 433–445. [[CrossRef](#)]
41. Shi, Y.; Li, P.; Yuan, H.; Miao, J.; Niu, L. Fast kernel extreme learning machine for ordinal regression. *Knowl. Based Syst.* **2019**, *177*, 44–54. [[CrossRef](#)]
42. Zhou, X.R.; Wang, C.S. Cholesky factorization based online regularized and kernelized extreme learning machines with forgetting mechanism. *Neurocomputing* **2016**, *174*, 1147–1155. [[CrossRef](#)]
43. Zhao, Y.P. Parsimonious kernel extreme learning machine in primal via Cholesky factorization. *Neural Netw.* **2016**, *80*, 95–109. [[CrossRef](#)] [[PubMed](#)]