Model identification of continuous stirred tank reactor based on QKLMS algorithm

LI Jun[1,2,3], LI Xiang-yue[1]

(1. *School of Automation & Electrical Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China*;
2. *Gansu Provincial Engineering Technology Center for Informatization of Logistics & Transport Equipment, Lanzhou 730070, China*;
3. *Gansu Provincial Industry Technology Center of Logistics & Transport Equipment, Lanzhou 730070, China*)

Abstract: The continuous stirred tank reactor (CSTR) is one of the typical chemical processes. Aiming at its strong nonlinear characteristics, a quantized kernel least mean square(QKLMS) algorithm is proposed. The QKLMS algorithm is based on a simple online vector quantization technology instead of sparsification, which can compress the input or feature space and suppress the growth of the radial basis function (RBF) structure in the kernel learning algorithm. To verify the effectiveness of the algorithm, it is applied to the model identification of CSTR process to construct a nonlinear mapping relationship between coolant flow rate and product concentration. In additiion, the proposed algorithm is further compared with least squares support vector machine (LS-SVM), echo state network (ESN), extreme learning machine with kernels (KELM), etc. The experimental results show that the proposed algorithm has higher identification accuracy and better online learning ability under the same conditions.

Key words: kernel learning algorithm; quantized kernel least mean square (QKLMS); continuous stirred tank reactor (CSTR); system identification

CLD number: TP183 **doi**: 10.3969/j.issn.1674-8042.2020.04.010

0 Introduction

The continuous stirred tank reactor (CSTR) is a highly nonlinear chemical reactor and is widely used in the chemical process industry including chemical reagents, fuels, synthetic materials, etc[1-3]. It is difficult to obtain an accurate mathematical model of the CSTR due to its features of nonlinearity and time-varying. In addition, if the system model is not accurate enough, the analysis, prediction and control of the system will be affected[4]. Hence, it is necessary to construct a nonlinear dynamic identification model based on the input-output data of the system, and further design a controller based on the identification model to realize the adjustment and control of the CSTR process[5-7].

In recent years, neural networks, as an effective artificial intelligence method, play an important role in the identification and control of chemical processes. In Refs.[8] and [9], Chen et al applied back propagation (BP) and generalized radial basis function (RBF) neural networks to CSTR system modeling, which effectively improved the identification accuracy. A dynamic recurrent neural network, echo state network (ESN), was used in CSTR process, and the identification effect was significantly improved[10]. In order to overcome the shortcomings of the feedforward neural network which is easy to fall into the local minimum, a nonlinear autoregressive exogenous input (NARX) identification method based on extrem learning machine with kernel (KELM) was proposed and applied to the modeling of CSTR, which obtained high identification accuracy[11]. However, when the training data increases, the computational complexity of the neural network also increases, which brings inconvenience to subsequent learning.

In order to further reduce the computational complexity and improve the online learning ability, relevant research in the field of online kernel learning algorithms has attracted wide attention of scholars[12-14]. In Ref.[15], Li et al proposed a kernel least mean square (KLMS) algorithm with low computational complexity and good robustness,

Received date: 2019-05-26

Foundation items: National Natural Science Foundation of China (No. 51467008); Scientific Research Projects of Colleges and Universities in Gansu Province (Nos. 2018C-10, 2017D-09)

Corresponding author: LI Jun (lijun691201@mail.lzjtu.cn)

which has been effectively used to the single-step and multi-step prediction of online traffic flow. Chen et al put forward a quantized kernel least mean square (QKLMS) algorithm[16], which is different from sparsification and uses the redundant data to update the coefficient of the closest centre. In Ref. [16], the algorithm was successfully applied to the prediction of chaotic time-series.

Therefore, for the identification of CSTR with strong nonlinearity features, we propose a novel identification method using QKLMS based on kernel learning algorithm to further improve the identification accuracy. In addition, to verify the effectiveness of the proposed algorithm, the identification experimental results are compared with the existing methods under the same conditions.

1 QKLMS

QKLMS is an online sequence estimation algorithm. Firstly, giving the training data $\{\boldsymbol{x}_i, y_i\} \in \mathbf{R}^m \times \mathbf{R}^1 (i=1,2,\cdots,N)$, we can define $\boldsymbol{X} \in \mathbf{R}^{N\times m}$ as the input matrix and $\boldsymbol{Y} \in \mathbf{R}^{N\times 1}$ as the output matrix. When the ith data set is obtained, the online update of the learning algorithm is performed on the basis of the estimation of the previous $(i-1)$th data (denoted as f_{i-1}) to obtain the estimated value of the current nonlinear mapping relationship f, recorded as f_i. The nonlinear mapping between the input and output data is expressed as

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}), \tag{1}$$

where the nonlinear mapping relationship is built by the linear combination of the kernel function constructed by the corresponding support vector $\boldsymbol{x}_i$.

Based on the QKLMS algorithm, firstly, $\boldsymbol{x}_i$ needs to be mapped to the high-dimensional feature space, i. e. $\boldsymbol{\phi}: \boldsymbol{x} \rightarrow \boldsymbol{\phi}(\boldsymbol{x}) \in \boldsymbol{F} \subseteq \mathbf{R}^M$, and the kernel function is defined as

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\phi}(\boldsymbol{x}_i)^{\mathrm{T}} \boldsymbol{\phi}(\boldsymbol{x}_j). \tag{2}$$

Then the kernel matrix $\boldsymbol{K} = \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi}$ that satisfies the Mercer condition can be obtained, where $\boldsymbol{\Phi} = [\boldsymbol{\phi}(\boldsymbol{x}_1) \ \boldsymbol{\phi}(\boldsymbol{x}_2) \ \cdots \ \boldsymbol{\phi}(\boldsymbol{x}_N)]$.

In the experiment, the kernel learning algorithm will use three kernel functions as follows.

1) Polynomial kernel

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = ((\boldsymbol{x}_i \cdot \boldsymbol{x}_j) + 1)^p, \tag{3}$$

where p is the order of the kernel function.

2) Sigmoid kernel

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \tanh(v(\boldsymbol{x}_i \cdot \boldsymbol{x}_j) + c), \tag{4}$$

where v is the input weight and c is the offset of the kernel function.

3) Gaussian kernel

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = -\frac{\| \boldsymbol{x}_i - \boldsymbol{x}_j \|^2}{2\sigma^2}, \tag{5}$$

where $\sigma(\sigma > 0)$ is the kernel width.

The KLMS algorithm extends the linear LMS algorithm into the feature space $\boldsymbol{F}$. The input $\boldsymbol{\phi}(\boldsymbol{x}_i)$ of the high-dimensional nonlinear feature space is denoted as $\boldsymbol{\phi}_i$. For the sequence data $\{\boldsymbol{\phi}_i, y_i\}$, the LMS algorithm is applied, then

$$\begin{cases} \boldsymbol{\omega}_0 = \mathbf{0}, \\ e_i = y_i - \boldsymbol{\omega}_{i-1}^{\mathrm{T}} \boldsymbol{\phi}_i, \\ \boldsymbol{\omega}_i = \boldsymbol{\omega}_{i-1} + \eta e_i \boldsymbol{\phi}_i, \end{cases} \tag{6}$$

where e_i is the prediction error when the ith data have been acquired, η is the learning rate, and $\boldsymbol{\omega}_i$ is the estimation value of weight vector in the feature space.

Due to $f_i = \boldsymbol{\omega}_i^{\mathrm{T}} \boldsymbol{\phi}(\cdot)$, and according to Eq. (6) and property $\boldsymbol{\phi}(\boldsymbol{x}) = k(\boldsymbol{x}, \cdot)$, the KLMS algorithm in the original space is

$$\begin{cases} f_0 = 0, \\ e_i = y_i - f_{i-1}(\boldsymbol{x}_i), \\ f_i = f_{i-1} + \eta e_i k(\boldsymbol{x}_i, \cdot). \end{cases} \tag{7}$$

Eq. (7) shows that the KLMS algorithm is essentially equivalent to a growing RBF network, i. e., as each new data are acquired, a new core unit centred on input $\boldsymbol{x}_i$ is assigned, and ηe_i is its coefficient.

The QKLMS algorithm is obtained by quantifying the feature vector $\boldsymbol{\phi}_i$, which is embodied in the weight update equation in Eq. (6). At this point, the KLMS algorithm on Eq. (7) is

$$\begin{cases} f_0 = 0, \\ e_i = yMi - f_{i-1}(\boldsymbol{x}_i), \\ f_i = f_{i-1} + \eta e_i k(Q[\boldsymbol{x}_i], \cdot), \end{cases} \tag{8}$$

where $Q[\cdot]$ is the quantization operator in the original space U, and additionally, let $\boldsymbol{x}_q(i) = Q[\boldsymbol{x}_i]$.

The key to the QKLMS algorithm is how to design the vector quantization technique, i. e., how to select the data as a codebook vector and how to find the nearest codebook vector representation. To adapt to online learning, the codebook vector needs to be trained directly from the online data to make it grow

adaptively. Let $\boldsymbol{D}_i$ be the matrix of all codebook vectors when the ith data are obtained, and $\boldsymbol{D}_i^j$ be the codebook vector of the elements of the jth column of the matrix. In addition, the data distance in the metric space is

$$\| \boldsymbol{\phi}_i - \boldsymbol{\phi}_j \| = \sqrt{(\boldsymbol{\phi}_i - \boldsymbol{\phi}_j)^{\mathrm{T}}(\boldsymbol{\phi}_i - \boldsymbol{\phi}_j)} = \sqrt{2 - 2k(\boldsymbol{x}_i, \boldsymbol{x}_j)} = \sqrt{2 - 2\exp\left(\frac{-\| \boldsymbol{x}_i - \boldsymbol{x}_j \|^2}{2\sigma^2}\right)}, \quad (9)$$

where $\| \cdot \|_{\boldsymbol{F}}$ is the norm in the feature space $\boldsymbol{F}$. Eq. (9) indicates that the distance in the feature space $\boldsymbol{F}$ monotonically increases as the distance of $\boldsymbol{U}$ in the original space changes.

Therefore, the quantization threshold can be defined by Eq. (9), then

$$\varepsilon_{\mathrm{F}} = \sqrt{2 - 2\exp\left(\frac{-\varepsilon_{\mathrm{U}}^2}{2\sigma^2}\right)}, \quad (10)$$

where $\varepsilon_{\mathrm{U}} = \| \boldsymbol{x}_i - \boldsymbol{x}_j \|$ is the quantization threshold in the original space $\boldsymbol{U}$ and $\| \boldsymbol{x}_q(i) - \boldsymbol{x}_j \| \leqslant \varepsilon_{\mathrm{U}}$. In addition, when $\varepsilon_{\mathrm{U}} = 0$, it is the KLMS algorithm.

In summary, the specific steps to implement the QKLMS algorithm are as follows:

Step 1: Give the data sets $\{\boldsymbol{x}_i \in \boldsymbol{U}, y_i\}$, $i = 1, 2, \cdots$

Step 2: Training phase. Let $\eta > 0$, $\sigma > 0$ and $\varepsilon_{\mathrm{U}} > 0$, when $i = 1$, let the initial value of the codebook vector matrix (the set of data centre) $\boldsymbol{D}_1 = [\boldsymbol{x}_1]$, and the coefficient vector $\boldsymbol{\alpha}_1 = [\eta y_1]$.

Step 3: Let $i = i + 1$, $L = \mathrm{size}(\boldsymbol{D}_{i-1})$, the model output is calculated by

$$\hat{y}_i = \sum_{j=1}^{L} \boldsymbol{\alpha}_{i-1}^j k(\boldsymbol{D}_{i-1}^j, \boldsymbol{x}_i), \quad (11)$$

where $\boldsymbol{\alpha}_{i-1}^j$ represents the jth element of the coefficient vector obtained when the ith data arrive.

Step 4: Calculation error is

$$e_i = y_i - \hat{y}, \quad (12)$$

and the minimum distance between the input vector and all codebook vectors is calculated by

$$d(\boldsymbol{x}_i, \boldsymbol{D}_{i-1}) = \min_{1 \leqslant j \leqslant L} \| \boldsymbol{x}_i - \boldsymbol{D}_{i-1}^j \|. \quad (13)$$

Step 5: If $d(\boldsymbol{x}_i, \boldsymbol{D}_{i-1}) \leqslant \varepsilon_{\mathrm{U}}$, the codebook vector matrix remains unchanged, i.e., $\boldsymbol{D}_i = \boldsymbol{D}_{i-1}$. Quantizing $\boldsymbol{x}_i$ to the nearest centre, by updating the coefficient vector of the nearest centre, that is

$$\alpha_i^{j^*} = \alpha_{i-1}^{j^*} + \eta e_i, \quad (14)$$

where the index of the centre is $j^* = \arg\min_{1 \leqslant j \leqslant L} \| \boldsymbol{x}_i - \boldsymbol{D}_{i-1}^j \|$.

Otherwise, setting $\boldsymbol{x}_i$ to the new codebook vector and updating the coefficient vector, that is

$$\begin{cases} \boldsymbol{D}_i = [\boldsymbol{D}_{i-1}, \boldsymbol{x}_i], \\ \boldsymbol{\alpha}_i = [\boldsymbol{\alpha}_{i-1}, \eta e_i]. \end{cases} \quad (15)$$

Step 6: Iteratively calculate Step 3— Step 5 until all training data are learned in turn.

Step 7: Testing phase. Based on the trained model, given the testing data, the final model output is calculated by Eq. (11).

2 Experiment of CSTR process

2.1 CSTR process

CSTR is a typical nonlinear reaction process, and its basic structure is shown in Fig. 1. In this paper, the CSTR chemical process with exothermic reaction feature is an irreversible reaction (A→B) process, and the producing heat will slow the reaction down. By introducing a coolant flow rate q_{c}, the temperature can be varied and hence the product concentration C_{B} can also be controlled.

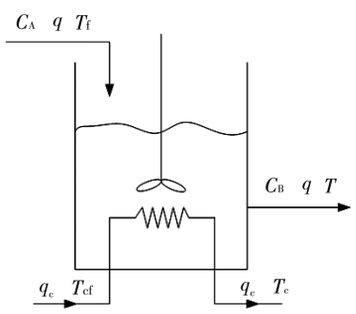

Fig. 1 Basic structure of CSTR

The dynamic nonlinear differential equations of the CSTR are expressed as

$$\frac{\mathrm{d}C_{\mathrm{B}}}{\mathrm{d}t} = \frac{q}{V}(C_{\mathrm{A}} - C_{\mathrm{B}}) - K_0 C_{\mathrm{B}} \exp\left(-\frac{E}{RT}\right), \quad (16)$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{q}{V}(T_{\mathrm{f}} - T) + \frac{(-\Delta H)K_0 C_{\mathrm{B}}}{\rho C_{\mathrm{p}}} \exp\left(-\frac{E}{RT}\right) + \frac{\rho_{\mathrm{c}} C_{\mathrm{pc}}}{\rho C_{\mathrm{p}} V} q_c \left[1 - \exp\left(-\frac{hA}{q_{\mathrm{c}} \rho C_{\mathrm{pc}}}\right)\right](T_{\mathrm{cf}} - T). \quad (17)$$

where q is the process flow rate; C_{A} is the inlet feed concentration; T_{f} and T_{cf} are the inlet feed and coolant temperatures, respectively; T is the temperature of the product. In addition, the remaining chemical reaction parameters are detail listed in Ref. [17].

2.2 Data collection and model building

During the CSTR process, when the reaction reaches equilibrium, the product concentration is

$C_B=8.36\times10^{-2}$ mol/L, simultaneously, the temperature of the product and the coolant flow rate are $T=440.2$ K and $q_c=103.4$ L/min, respectively. On the basis that the coolant flow rate q_c is a steady state value, the randomly distributed white noise in the interval $[-0.002,0.002]$ is added to enhance the stability of the model training.

The coolant flow of the CSTR input is shown in Fig. 2. Among the obtained 4 000 input-output data, the first half is used as the training set, and the remaining part is used as the test set. In addition, all data sets need to be normalized on the interval $[0,1]$.

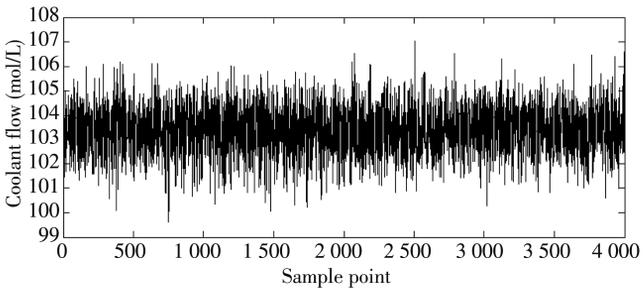

Fig. 2 Coolant flow q_c of CSTR input

Assuming that the CSTR system model is unknown, only the above input and output data being known, the identification model of the CSTR is selected as

$$\hat{y}_{C_B}(i)=f[q_c(i-1),q_c(i-2),q_c(i-3),C_B(i-1),C_B(i-2),C_B(i-3)]. \tag{18}$$

3 Experimental results and analysis

In the experiment, the mean square error(MSE)is selected as the performance indicator of the identification model, that is

$$E=\frac{1}{N}\sum_{i=1}^{N}(y_i-\hat{y}_i), \tag{19}$$

where y_i and $\hat{y}_i$ are the actual output and test output, respectively; and N is the number of the testing data set.

In the experiment, QKLMS selects three common kernel functions according to Eqs. (3)—(5), and the corresponding parameters are set as follows: the order of Polynomial kernel function is $p=2$; the input weight and offset of Sigmoid kernel function are $v=0.8$ and $c=1$, respectively; and the kernel width of Gaussian kernel function is $\sigma=1$.

When the quantization threshold ε_U varies between (0,1), the identification performance on the test set can be obtained, and the experiment results are shown in Table 1.

Table 1 Comparison of identification performance of QKLMS based on three different kernel functions

ε_U	Polynomial kernel		Sigmoid kernel		Gaussian kernel	
	MSE_test	Train (s)	MSE_test	Train (s)	MSE_test	Train (s)
0.1	$3.112\,5\times10^{-6}$	2.569 3	$5.262\,0\times10^{-7}$	3.210 4	$2.014\,7\times10^{-7}$	3.215 8
0.2	$8.589\,6\times10^{-7}$	3.136 9	$3.662\,5\times10^{-7}$	3.701 1	$1.233\,2\times10^{-7}$	2.682 6
0.3	$2.369\,3\times10^{-7}$	3.056 9	$6.316\,8\times10^{-8}$	3.596 3	$4.363\,5\times10^{-8}$	3.313 3
0.4	2.8521×10^{-8}	2.133 8	$2.142\,6\times10^{-8}$	3.332 1	$3.771\,5\times10^{-9}$	3.369 8
0.5	$6.745\,3\times10^{-8}$	3.140 1	$3.658\,6\times10^{-9}$	2.910 3	$5.931\,1\times10^{-10}$	2.501 4
0.6	$9.432\,8\times10^{-8}$	2.090 1	$8.325\,9\times10^{-9}$	3.075 2	$8.782\,3\times10^{-11}$	3.329 3
0.7	$1.696\,3\times10^{-7}$	2.187 2	$3.437\,7\times10^{-8}$	2.861 0	$6.252\,1\times10^{-9}$	2.978 8
0.8	$7.362\,9\times10^{-7}$	3.373 0	$2.693\,0\times10^{-8}$	2.753 0	$5.158\,2\times10^{-8}$	3.072 2
0.9	$3.512\,8\times10^{-6}$	3.132 4	$4.364\,1\times10^{-7}$	3.009 8	$3.396\,4\times10^{-7}$	3.142 8

It can be seen from Table 1 that the MSE changes with the change of quantization threshold ε_U. The comparison shows that when the Gaussian kernel function is selected, the identification accuracy is the best and can reach $8.782\,3\times10^{-11}$. It can also be further observed that the training time is concentrated in 2.50 s—3.36 s.

Considering further experiment based on Gaussian kernel function, that is, when the learning rate η varies between (0,1), the identification performance on the test set can be observed, and the results are shown in Table 2. It is observed from Table 2 that the MSE of QKLMS changes with the change of η, and the training time is concentrated in 2.89 s—3.81 s. When the learning rate is 0.45, the identification accuracy is the highest and the MSE of the testing set is $3.850\,9\times10^{-11}$.

Table 2 Comparison of identification performance when learning rate changes

η	MSE_test	Train (s)	Test (s)
0.01	$6.284\,9\times10^{-8}$	3.321 5	4.732 5
0.05	$3.725\,1\times10^{-8}$	3.409 8	5.325 8
0.15	$5.523\,9\times10^{-9}$	3.063 1	4.691 2
0.20	$6.158\,2\times10^{-10}$	3.525 5	5.410 5
0.30	$1.329\,5\times10^{-10}$	2.956 3	4.901 1
0.40	$5.025\,9\times10^{-10}$	3.644 5	5.365 4
0.45	$3.850\,9\times10^{-11}$	3.811 2	5.221 7
0.55	$7.365\,2\times10^{-11}$	2.891 1	5.093 6
0.70	$4.132\,8\times10^{-10}$	3.755 3	4.868 2
0.85	$3.285\,6\times10^{-9}$	3.414 7	5.283 3

When $\eta=0.45$, the identification performance of the test set can be observed from Table 3 by changing the sizes of ε_U and σ. It can be seen that the MSE changes with the changes of ε_U and σ, and the training time is relatively concentrated in 2.72 s—3.61 s. When $\varepsilon_U=0.60$, and $\sigma=6$, the identification accuracy of QKLMS is the highest and the MSE of the test set is $1.232\ 8\times10^{-14}$. Moreover, the identification accuracy of QKLMS is nearly one order of magnitude higher than that of KLMS ($\varepsilon_U=0$) under the same conditions.

Table 3 Comparison of identification performance of QKLMS

ε_U	σ	MSE_test	Train (s)	Test (s)
0.10	83	$2.360\ 2\times10^{-8}$	2.725 8	5.052 3
0.25	15	$8.147\ 3\times10^{-9}$	3.563 4	4.664 4
0.35	33	$3.582\ 4\times10^{-10}$	3.373 2	5.456 9
0.40	56	$6.621\ 1\times10^{-11}$	3.052 0	5.123 5
0	6	$1.850\ 9\times10^{-13}$	3.240 9	4.982 2
0.50	38	$1.053\ 2\times10^{-13}$	2.901 2	4.496 3
0.55	17	$7.257\ 3\times10^{-14}$	2.887 3	5.521 4
0.60	6	$1.232\ 8\times10^{-14}$	3.252 1	5.337 2
0.65	33	$5.214\ 7\times10^{-13}$	3.617 8	4.833 6
0.75	42	$3.562\ 3\times10^{-12}$	2.831 2	5.374 4
0.80	61	$4.362\ 5\times10^{-10}$	3.577 9	346 112

When $\varepsilon_U=0.60$, $\sigma=6$ and $\eta=0.45$, Fig. 3 shows the error curve of each sample point on the test set. Fig. 4 shows the comparison curve of the actual value and identification value of output concentration. The magnitude of the identification error is approximately 10^{-7}. Combining with Figs. 3—4, it can be seen that QKLMS can achieve excellent results in the modeling of CSTR.

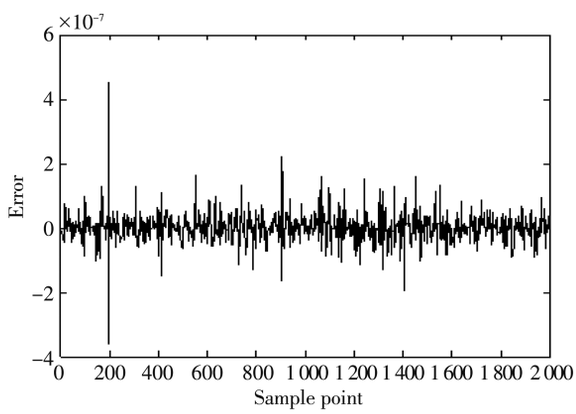

Fig. 3 Error curve of each sample point on test set

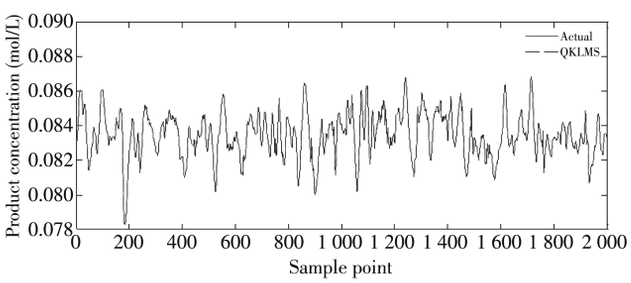

Fig. 4 Comparison curve of the actual value and identification value of output concentration

To further verify the validity of the QKLMS algorithm, Table 4 shows the MSE of the QKLMS and the existing identification algorithm on the test set under the same conditions. The comparison shows that the identification accuracy of QKLMS is improved by one order of magnitude.

Table 4 Comparison of identification performance based on QKLMS and other algorithms

Algorithm	MSE_test
BP-MLP[18]	-1.7×10^{-2}
LS-SVM[18]	-1.6×10^{-3}
FNN[8]	2.976×10^{-4}
GAP-RBF[9]	4.956×10^{-7}
MGAP-RBF[9]	4.078×10^{-7}
ESN[10]	$1.105\ 3\times10^{-7}$
NARX-ELM[11]	$4.160\ 2\times10^{-11}$
NARX-KELM[11]	$2.043\ 6\times10^{-13}$
KLMS	$1.850\ 9\times10^{-13}$
QKLMS	$1.232\ 8\times10^{-14}$

4 Conclusion

In this paper, based on the input and output data of the unknown nonlinear CSTR process, a novel identification method with the QKLMS algorithm is proposed. The online vector quantization method is used to quantize the input in the feature space and control the size of the kernel function structure.

Compared with existing feed-forward neural network method and ESN network as well as other kernel learning methods, under the same conditions, experimental results show that the identification method based on QKLMS algorithm can achieve good results, and improve the accuracy of identification effectively. Therefore, our research provides a new idea for the complex nonlinear process that is difficult to obtain accurate mathematical models.

References

[1] Li T, Wei F K, Wang Y C, et al. Recovery of long-term storage of nitrosated flocculent sludge activity. Environmental Science, 2018, 39(9): 4310-4316.

[2] Torres-Mancera P, Ancheyta J, Martínez J. Deactivation of a hydrotreating catalyst in a bench-scale continuous stirred tank reactor at different operating conditions. Fuel, 2018, 234: 326-334.

[3] Zhao L L, Feng J, Zhao L X, et al. Research progress on optimization of full-mix anaerobic fermentation reactor. China Biogas, 2018, 36(4): 33-38.

[4] Kothare M V. Process control: modeling, design and simulation. IEEE Control Systems Magazine, 2007, 26(6): 106-108.

[5] Ding X Q, Yang X L, Yang H, Nonlinear predictive con-

troller design for CSTR process. Control Engineering of China, 2009, 16(2): 145-147.
[6] Li D J. Adaptive neural network control of continuous stirred tank reactor. CIESC Journal, 2013, 64(12): 4674-4680.
[7] Beyhan S, Alci M. Mixed structured RBF network for direct inverse control of nonlinear systems. In: Proceedings of 2009 Fifth IEEE International Conference on Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control, Famagusta, IEEE, 2009: 1-4.
[8] Chen F, Jing Z L, Yao X D. Fast parameter learning algorithm for fuzzy neural networks. Control Theory and Applications, 2002, 19(4): 583-587.
[9] Kamalabady A S, Salahshoor K. A new on-line identification approach for affine modeling of nonlinear processes using an adaptive neural network. In: Proceedings of 16th Mediterranean Conference on Control and Automation, Ajaccio, France, IEEE, 2008: 735-740.
[10] Li X H, Li J. Identification of continuous stirred tank reactor based on echo state network. Information and control, 2014, 43(2): 223-228.
[11] Li J, Shi Q. Model Identification of Continuous Stirring Reactor Based on KELM. Control Engineering, 2017, (10): 171-177.
[12] Rosipal R J, Trejo L J. Kernel partial least squares regression in reproducing kernel hilbert space. Journal of Machine Learning Research, 2002, 2: 97-123.
[13] Li H L, Qian Z H, Tian H L. Research on indoor localization algorithmbased on kernel principal component analysis. Journal on Communications, 2017, 38(1): 158-167.
[14] Engel Y, Mannor S, Meir R. The kernel recursive least-squares algorithm. IEEE Transactions on Signal Processing, 2004, 52(8): 2275-2285.
[15] Li J, Wang Q L. Short-term traffic flow online forecasting based on kernel adaptive filter. Journal of Measurement Science & Instrumentation, 2018, 9(4): 326-334.
[16] Chen B, Zhao S, Zhu P, et al. Quantized kernel least mean square algorithm. IEEE Transactions on Neural Networks and Learning Systems, 2012, 23(1): 22-32.
[17] Lightbody G, Irwin G W. Nonlinear control structures based on embedded neural system models. IEEE transactions on Neural Networks, 1997, 8(3): 553-567.
[18] Li H S, Zhong Z Y, Zhang Y L. Predictive Control Based on Least Squares Support Vector Machine. Computing Technology and Automation, 2009, 28(1): 32-36.

基于量化核最小均方算法的连续搅拌反应釜模型辨识

李 军[1,2,3], 李香月[1]

(1. 兰州交通大学 自动化与电气工程学院, 甘肃 兰州 730070;
2. 甘肃省物流及运输装备信息化工程技术研究中心, 甘肃 兰州 730070;
3. 甘肃省物流与运输装备行业技术中心, 甘肃 兰州 730070)

摘 要: 连续搅拌反应釜(Continuous stirred tank reactor, CSTR)是典型的化工过程之一, 本文针对其强非线性特性, 提出了一种量化核最小均方(Quantized kernel least mean square, QKLMS)算法。该算法基于一种简单在线矢量量化技术替代稀疏化准则, 可以对输入空间进行压缩, 从而抑制核学习算法中径向基函数(Radial basis function, RBF)结构的增长。为验证该算法的有效性, 将其应用于CSTR过程的模型辨识中, 构建冷却剂流量与生成物浓度之间的非线性映射关系。此外, 将所提算法与最小二乘支持向量机(Least square support vector machine, LS-SVM)、回声状态网络(Echo state network, ESN)以及核极限学习机(Extreme learning machine with kernels, KELM)等算法进行比较。实验结果表明, 在同等条件下, 本文所提算法具有更高的辨识精度和更好的在线学习能力。

关键词: 核学习算法; 量化核最小均方; 连续搅拌反应釜; 系统辨识

引用格式: LI Jun, LI Xiang-yue. Model identification of continuous stirred tank reactor based on QKLMS algorithm. Journal of Measurement Science and Instrumentation, 2020, 11(4): 382-387. [doi: 10.3969/j.issn.1674-8042.2020.04.010]