Neural Computing and Applications (2024) 36:2315–2322
https://doi.org/10.1007/s00521-023-08722-7

S.I.: MACHINE LEARNING AND BIG DATA ANALYTICS FOR IOT SECURITY AND PRIVACY (SPIOT 2022)

Fuzzy identification of nonuniformly sampled nonlinear systems based on forwards recursive input–output clustering

Ranran Liu[1] · Enxing Zheng[2] · Feng Li[2] · Wei Guo[1] · Yifeng Jiang[3]

Received: 24 January 2023 / Accepted: 31 May 2023 / Published online: 23 June 2023

Abstract
Based on forwards recursive input–output data clustering, a recursive least squares (RLS) algorithm is proposed to estimate nonuniformly sampled nonlinear systems. The relationship of linear and nonlinear systems is studied under nonuniform sampling, and a fuzzy model is constructed for the global system. Forwards recursive input–output data clustering based on k-means clustering is used to identify the fuzzy rule number and the antecedent parameters. Based on the membership function, the consequent parameters are identified by an RLS algorithm. A practical application verified the efficiency of the method.

Keywords Nonuniform sampling · Nonlinear systems · System identification · Fuzzy k-mean cluster

1 Introduction

Because of hardware limitations, economic constraints or circumstance requirements, there are many nonuniformly sampled-data systems (NUSDS) in the process industry (such as petroleum, chemical, food, medicine and so on). Furthermore, input and output data may be sampled nonuniformly because of various restricted conditions, such as timing jitter, manual sampling, data loss or transmission delay. Thus, NUSDS is widely used in industrial processes [5, 6, 13]. The NUSDS is a special multirate sampling system that has irregular sampling intervals of input–output data. NUSDS not only reflect the characteristics of data sampling in actual industry but also affect the control performance directly. The identification of NUSDS has become a research hotspot.

Based on nonuniformly sampled-data (NUSD) discrete-time systems, Ding studied how to recover a continuous-time system [2]. Furthermore, Liu deduced lifted state-space models, and an AM-RLS algorithm was studied to identify the NUSDS parameters [7]. Based on these works, Xie presented iterative methods based on the least squares/gradient algorithm [18]. To find the unknown variables in the information vectors, Fan constructed an auxiliary model, and then, a multi-innovation recursive algorithm was presented [4]. The simulation results showed that it can improve the accuracy and convergence rate. However, the computation size is larger. To reduce the computation, Liu decomposed the system, and the subsystems were identified by using a hierarchical least squares algorithm [8].

NUSD linear systems were studied in predecessors' studies. However, there also exist nonlinear features in actual industry. However, modular nonlinear systems were identified under single-rate sampling [1, 12, 14, 15, 19]. Because of the combination of the linear and nonlinear characters, computation is high in these systems. To overcome this problem and to improve accuracy, the system model was separated into two submodels. Then, Ding

✉ Enxing Zheng
zex@jsut.edu.cn

[1] School of Automotive and Traffic Engineering, Jiangsu University of Technology, Changzhou 213001, Jiangsu, China

[2] School of Electrical and Information Engineering, Jiangsu University of Technology, Changzhou, China

[3] Information Center of Jiangsu University of Technology, Changzhou 213001, Jiangsu, China

proposed a modified algorithm to estimate all the parameters [3, 16].

Considering the dual-rate sampling characteristics, the outliers in the actual process and the irregular time delays, Ma proposed an EM algorithm to estimate the input nonlinear systems with dual-rate sampling [11]. Then, Liu studied a class of NUSD nonlinear (NUSDN) systems. Due to the coupling parameters, the computation was huge. An iterative RLS algorithm was proposed to identify the parameters directly [9, 10]. Furthermore, she proposed an RLS algorithm to identify a NUSDN system with unknown nonlinear characteristics based on a multimodel approach [9, 10]. Since the input signal refreshed quickly and the output signal sampled slowly, Wang transformed the nonlinear system into a series of consequent linear models of the fuzzy model by the lifting technique. Then, a fuzzy identification algorithm was proposed based on competitive learning and the recursive gradient descent method [17]. Although these methods can identify systems whose structure is unknown, the number of local models is difficult to determine when there is no other a priori knowledge.

In this paper, the NUSDN system is depicted as a fuzzy model. To estimate the model, a two-phase approach is obtained. First, the inputs and outputs are analysed by using recursive clustering analysis. Then, the model structure is identified. Once the rule number and the antecedent parameters are obtained, we have the global model including the membership function and local linear model. Furthermore, based on an RLS method, the consequent parameters can be obtained. The simulation of an industry process verifies the performance of the method.

2 Problem formulation

The NUSDN system is depicted in Fig. 1.

Here, $u(kT+t_{j-1}), u(t)$ are the inputs of the NUSDN system and the nonlinear module S_c, respectively. Accordingly, $y(kT+T), y(t)$ are the outputs. H_τ is a nonuniform zero-order holder. The sampling interval is $\{\tau_1, \tau_2, \ldots, \tau_m\}$. Then, the inputs are updated at time $kT+t_{j-1}, j=1,2,\ldots,m\,(t_0=0, t_j:=\tau_1+\tau_2+\ldots+\tau_j)$. S_c and S_T are the nonlinear dynamic module and the sampler, respectively, and the frame period is $T:=\tau_1+\tau_2+\ldots+\tau_m=t_m$. Based on the lifting technique, we can obtain the following results:

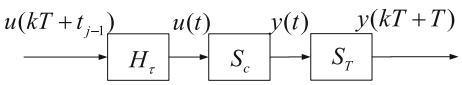

Fig. 1 Construction of the NUSDN system

$$u(t)=\begin{cases} u(kT), & kT\le t<kT+t_1 \\ u(kT+t_1), & kT+t_1\le t<kT+t_2 \\ \vdots & \vdots \\ u(kT+t_{m-1}), & kT+t_{m-1}\le t<kT+T \end{cases} \tag{1}$$

Taking S_c as the state space model of the nonlinear character, it can be described by

$$S_c:=\begin{cases} \dot{\mathbf{x}}(t)=\Gamma(\mathbf{x}(t),u(t)) \\ y(t)=h(t) \end{cases} \tag{2}$$

Here, we consider $\mathbf{x}(t)\in R^n$ to be the state vector. $y(t)\in R, u(t)\in R$ are the outputs and inputs of S_c, respectively.

2.1 Relationship between the linear and nonlinear models

First, we consider P_c as a linear dynamic block

$$P_c:=\begin{cases} \dot{\mathbf{x}}(t)=\mathbf{A}_c\mathbf{x}(t)+\mathbf{B}_c u(t) \\ \overline{y}(t)=\mathbf{C}\mathbf{x}(t)+Du(t) \end{cases}, \tag{3}$$

In Eq. (3), $\mathbf{x}(t)\in\mathbb{R}^n$ has been defined before, and $\overline{y}(t)\in\mathbb{R}, u(t)\in\mathbb{R}$ are the I/O signals of the linear dynamic block. We define $\mathbf{A}_c\in\mathbb{R}^{n\times n}, \mathbf{B}_c\in\mathbb{R}^n, \mathbf{C}\in\mathbb{R}^{1\times n}, D\in\mathbb{R}$ as the parameter matrices. Given discrete P_c, we can obtain the linear characters as follows:

$$\begin{aligned} \overline{y}(kT) &= \frac{\sum_{j=1}^m B_j(z^{-1})}{A(z^{-1})}u(kT+t_{j-1}) \\ &= [1-A(z^{-1})]\overline{y}(kT)+\sum_{j=1}^m B_j(z^{-1})u(kT+t_{j-1}), \end{aligned} \tag{4}$$

where

$$\begin{cases} A(z^{-1})=1+a_1z^{-1}+a_2z^{-2}+\ldots+a_{n_a}z^{-n_a} \\ B_1(z^{-1})=b_{10}+b_{11}z^{-1}+\ldots+b_{1n_b}z^{-n_b} \\ B_j(z^{-1})=b_{j1}z^{-1}+b_{j2}z^{-2}+\ldots+b_{jn_b}z^{-n_b}, j=2,3,\cdots,m \end{cases},$$

Here, z^{-1} is defined as the backwards shift operator; then, it satisfies $z^{-1}y(kT)=y(kT-T)$.

The relationship between (2) and (3) is as follows: The nonlinear systems have multiple equilibria points, and the local linear model (3) is taken to describe the nonlinear characteristics. Furthermore, the combination of the local linear model and the nonlinear weighting functions describes this whole nonlinear system model. Then, we describe the nonlinear system (2) as follows:

$$y(kT)=\sum_{q=1}^{l} f_q(\varphi(kT))g_q(\varphi(kT)) \tag{5}$$

Here, $\varphi(kT) \in R^{n_\varphi}$ is the information vector, and $y(kT) \in R$ is the output of the system. $g_q(\cdot),, q = 1, 2, \ldots, l$ is the qth local model, and $f_q(\cdot), q = 1, 2, \ldots, l$ is the nonlinear weighting function of $g_q(\cdot)$. We assume l is the number of $g_q(\cdot).\varphi(kT)$ is defined as:

$$\begin{aligned}\varphi(kT) := [&-y(kT-T), -y(kT-2T), \ldots,\\ &-y(kT-n_aT),\ u(kT), u(kT-T), \ldots,\\ &u(kT-n_bT), u(kT-T+t_1), \ldots,\\ &u(kT-n_bT+t_1), \ldots, u(kT-T+t_{m-1}),\\ &\ldots, u(kT-n_bT+t_{m-1})]^T \in R^{n_\varphi}, n_\varphi = n_a + mn_b + 1\end{aligned} \tag{6}$$

where n_a is the order of the system output, and n_b is the order of the system input.

2.2 Fuzzy modelling

We consider the system shown in (2). If the input signals are sampled nonuniformly, we can obtain its fuzzy modelling as.

R^q : if $\varphi(kT)$ belongs to F_q (Membership function of F_q is $\mu_q[\varphi(kT)]$), then

$$\begin{aligned}y(kT) &= \frac{\sum_{j=1}^{m} B_j^q(z^{-1})}{A^q(z^{-1})} u(kT + t_{j-1})\\ &= [1 - A^q(z^{-1})]y(kT) + \sum_{j=1}^{m} B_j^q(z^{-1})u(kT + t_{j-1})\\ &= g_q(\varphi(kT))\end{aligned} \tag{7}$$

We set F_q as the fuzzy set of the qth rule and $\mu_q[\varphi(kT)]$ as its membership function; l is the number of fuzzy rules; $\varphi(kT)$ is sampling at time kT, as shown in (6); $A^q(z^{-1})$ and $B_j^q(z^{-1})$ satisfy

$$A^q(z^{-1}) = 1 + a_{q1}z^{-1} + a_{q2}z^{-2} + \ldots + a_{qn_a}z^{-n_a}$$

$$B_1^q(z^{-1}) = b_{q10} + b_{q11}z^{-1} + \ldots + b_{q1n_b}z^{-n_b}$$

$$\begin{aligned}B_j^q(z^{-1}) &= b_{qj1}z^{-1} + b_{qj2}z^{-2} + \ldots + b_{qjn_b}z^{-n_b}, j\\ &= 2, 3, \cdots, m\end{aligned}$$

Then, for (7), the consequent fuzzy model can be described as.

R^q : if $\varphi(kT)$ belongs to F_q, then

$$\begin{aligned}y(kT) = &-a_{q1}y(kT-T) - \cdots - a_{qn_a}y(kT-n_aT)\\ &+ \sum_{j=1}^{m} B_j^q(z^{-1})u(kT + t_{j-1})\end{aligned}$$

Therefore, we can obtain the following form:

$$\begin{aligned}y(kT) = \sum_{q=1}^{l} \mu_q[\varphi(kT)][&-a_{q1}y(kT-T) - \cdots - a_{qn_a}y(kT\\ &- n_aT) + \sum_{j=1}^{m} B_j^q(z^{-1})u(kT + t_{j-1})]\end{aligned} \tag{8}$$

From (8), the following can be seen: (1) Locally, there is a discrete consequent model of each fuzzy rule in every polynomial expression in (8). Each one describes the local dynamic process of a nonlinear system, which is equivalent to the discrete model of a linear system developed near an equilibrium point of the qth rule. (2) Overall, the whole nonlinear model is obtained after the weighted connection of the fuzzy membership function, and the discrete models of local linearity are also connected. Additionally, the whole nonlinear model describes the whole regional dynamic evolution process of the entire system. Therefore, the dynamic change process of the whole nonlinear system is described in (8).

We choose the Gaussian membership function as follows:

$$\mu_q[\varphi(kT)] = \exp\left\{-\frac{[\varphi(kT) - c_q]^2}{\sigma_q^2}\right\}$$

where c_q, σ_q are the centre and width parameters of $\mu_q[\varphi(kT)]$, respectively.

We propose an approach to identify fuzzy systems from three aspects: rule number l, antecedent parameters c_q, σ_q, and consequent parameters, including $[a_{q1}, a_{q2}, \ldots, a_{qn_a}, b_{q10}, \ldots, b_{q1n_b}, \ldots, b_{qj1}, \ldots, b_{qjn_b}]$, where $q = 1, 2, \ldots, l, j = 1, 2, \ldots, m$.

3 Stochastic gradient fuzzy algorithm based on forwards recursive clustering (SGF-FRC)

From the last section, the identification of the NUSDS can be divided into two steps. First, the rule number l and the antecedent parameters c_q, σ_q are estimated. Second, the consequent parameters are identified. In the first step, we can obtain the system structure based on an FRC approach. If the number of clusters l is obtained, the corresponding centres $[c_1, c_2, \ldots, c_l]$ and width parameters $[\sigma_1, \sigma_2, \ldots, \sigma_l]$ of the Gaussian membership function can be determined. Then, consequent parameters are identified based on a recursive least squares algorithm. Thus, the proposed approach can be named the recursive least squares fuzzy algorithm based on forwards recursive clustering (RLS-FRC).

3.1 Structure and antecedent parameter identification

In this section, the rule number of (8) can be obtained by an FRC method. First, based on the k-means clustering method, inputs are partitioned. Second, the output data are also divided based on the clustering results. Then, if the output data of every cluster are out of the acceptable limit, the cluster is clustered again. Thus, the clustering process is recursive. In the process of identification, this clustering method can perfectly meet the demand. If the system is smoother, the clustering is coarser. In contrast, if the nonlinear characteristic of the system is obvious, the clustering is fine. The details of the method are described as follows.

We assume the data to be clustered as $\{u(t), y(t)\}, (t = 1, 2, \ldots, N)$, in which N is the sampling number.$u(t), y(t)$ are the inputs and outputs, respectively. In the NUSDS, we take $[u(kT), u(kT + t_1) \ldots, u(kT + t_{m-1})]$ and $y(kT)$ as $u(t)$ and $y(t)$, respectively. Otherwise, $y_{\min}$ is defined as the minimum of $y(kT)$. We define $y_{\max}$ as the maximum of $y(kT)$, shown as

$$y_{\min} = \min_{k=1,2,\ldots,N} \{y(kT)\}, y_{\max} = \max_{k=1,2,\ldots,N} \{y(kT)\} \tag{9}$$

We suppose a small real value $\varepsilon\,(\varepsilon > 0)$ is the threshold. We initialize the quantity of clusters as L:

$$\frac{y_{\max} - y_{\min}}{L} < \varepsilon \tag{10}$$

Then, $[y_{\min}, y_{\max}]$ is equally divided into L clusters, and every cluster is within the threshold. Therefore, as the min value, L can be calculated as

$$L = \left[\frac{y_{\max} - y_{\min}}{\varepsilon}\right]_+ \tag{11}$$

Here, $[\cdot]_+$ is the rounding function.

Once the initial cluster number L is determined by (11), the input data can be partitioned into the L set by using the k-means clustering algorithm. The result of the L clusters is defined as $u_{c_1}, u_{c_2}, \ldots, u_{c_L}$.

The input data are partitioned based on their similarities, but the corresponding output set may be out of the threshold. Therefore, we need to check whether the outputs for each cluster meet the requirement. For each cluster, if the outputs are out of the threshold, the corresponding input set should be clustered recursively. In contrast, if the outputs are within the threshold, the corresponding input set should not be clustered further. By applying this method recursively, the desired clustering outcome is obtained. The process is specified below.

After k-means clustering, the input data are partitioned into L clusters $u_{c_i} (i = 1, 2, \ldots, L)$. Then, the corresponding outputs are defined as follows:

$$y_{c_i} = \{y(t) | u(t) \in u_{c_i}\} \tag{12}$$

For each $y_{c_i}, (i = 1, 2, \ldots, L)$, we assume that $y_{\min_i}, y_{\max_i}$ are the minimal value and max value, respectively.

$$y_{\min_i} = \min_{y(t) \in y_{c_i}} \{y(t)\}, y_{\max_i} = \max_{y(t) \in y_{c_i}} \{y(t)\} \tag{13}$$

Then, if $y_{\max_i} - y_{\min_i} < \varepsilon$, the outputs are within the threshold. This means that the cluster meets the requirement. Therefore, the corresponding inputs do not need to be further clustered. The outcome clusters are sufficient to represent the system. In contrast, if $y_{\max_i} - y_{\min_i} > \varepsilon$, the outputs are out of the threshold. Then, the cluster cannot meet the requirement. Thus, the corresponding inputs need to be further clustered. The outcome clusters are sufficient to represent the system. The further clustering for u_{c_i} is like the initial clustering. Let L_i be the smallest integer as

$$\frac{y_{\max_i} - y_{\min_i}}{L_i} < \varepsilon \tag{14}$$

Similarly, it can be rewritten as

$$L_i = \left[\frac{y_{\max_i} - y_{\min_i}}{\varepsilon}\right]_+ \tag{15}$$

where L_i is the cluster number for u_{c_i}. Then, based on the proposed clustering process, subclusters of u_{c_i} can be obtained as $u_{c_{i,1}}, u_{c_{i,2}}, \ldots, u_{c_{i,L_i}}$. The obtained subclusters also need to evaluated for further clustering by using the same method described above. Thus, the clustering process is recursive. The subclustering ends if both input–output signals of every cluster can satisfy variation within the threshold.

The proposed forwards FRC method can be summarized as follows:

Step 1 We define the final cluster set, the intermediate clusters and the clusters to be checked as $L_{final}, L_{temp}, L_{check}$ separately. We initialize them as empty.

Step 2 We calculate the initial number of clusters L based on (9)–(11).

Step 3 We partition the inputs based on the k-means cluster, and the input set $L_{\text{check}} = \{u_{c_1}, u_{c_2}, \ldots, u_{c_L}\}$ can be obtained.

Step 4 For each $u_{c_i}, i = 1, 2, \ldots, L$, we determine their outputs by (12). We calculate the minimum and maximum values of the collected output data based on (13). We check the integer $y_{\max_i} - y_{\min_i}$; if $y_{\max_i} - y_{\min_i} < \varepsilon$, we add u_{c_i} into L_{final}; otherwise, we add u_{c_i} into L_{temp}. Then, let $L_{check} = L_{temp}$.

Step 5 We check the output data of each cluster to decide whether the subcluster needs to be further clustered. If $L_{check} \neq empty$, then we obtain outputs and calculate $y_{\min_i}$ and $y_{\max_i}$. If $y_{\max_i} - y_{\min_i} < \varepsilon$, we add u_{c_i} into L_{final}; otherwise, we calculate L_i based on (15). Then, we partition u_{c_i} by using k-means clustering and obtain L_i clusters

$u_{c_{i,1}}, u_{c_{i,2}}, \ldots, u_{c_{i,L_i}}$. We add $\{u_{c_{i,1}}, u_{c_{i,2}}, \ldots, u_{c_{i,L_i}}\}$ into L_{temp}. Let $L_{\text{check}} = L_{\text{temp}}$, and we go back to Step 5.

Let us take the Gaussian function as the membership function. We define $\mathbf{c} = [c_1, c_2, \ldots, c_l]$ as the centre vector and $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_l]$ as the width vector of the Gaussian function. Once clusters are identified, $\mathbf{c}, \boldsymbol{\sigma}$ can be determined. If the outcome is $L_{final} = \{L_1, L_2, \ldots, L_l\}$, in which $L_i = [u_{i,1}(t), u_{i,2}(t), \ldots, u_{i,n_i}(t)], i = 1, 2, \ldots, l$ and l are the *ith* cluster and final quantity of clusters, respectively.

Then,

$$\mathbf{c} = [c_1, c_2, \ldots, c_l], \quad \boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_l] \tag{16}$$

where

$$c_i = (c_{i,1}, c_{i,2}, \ldots, c_{i,n_i}), \quad \sigma_i = (\sigma_{i,1}, \sigma_{i,2}, \ldots, \sigma_{i,n_i}) \tag{17}$$

In (17),

$$c_{i,j} = \frac{1}{n_i}\sum_{j=1}^{n_i} u_{i,j}(t)\,,\ \sigma_{i,j} = \frac{1}{\sqrt{2}}\sqrt{\frac{1}{n_i}\sum_{j=1}^{n_i}\left[u_{i,j}(t) - c_{i,j}\right]^2} \tag{18}$$

3.2 Consequent parameter identification

We define the model parameter $\boldsymbol{\theta}_q$ of the *qth* rule as

$$\boldsymbol{\theta}_q := [a_{q1}, a_{q2}, \ldots, a_{qn_a}, b_{q10}, b_{q11}, \ldots b_{q1n_b}, b_{q21}, \ldots, b_{q2n_b}, \ldots, b_{qm1}, \ldots, b_{qmn_b}]^T \in R^{n_\varphi}, n_\varphi = n_a + mn_b + 1$$

We consider the influence of white noise, Eq. (8) is described by Eq. (19)

$$y(kT) = \sum_{q=1}^{l} \mu_q[\varphi(kT)][\varphi(kT)\boldsymbol{\theta}_q + v(kT)] \tag{19}$$

where $q = \arg\min_{t=1,2,\ldots,l} (\phi(kT) - v_t)^T(\phi(kT) - v_t)$. The system parameter vector $\boldsymbol{\theta}$ is:

$$\boldsymbol{\theta} := [\boldsymbol{\theta}_1^T, \boldsymbol{\theta}_2^T, \ldots, \boldsymbol{\theta}_l^T]^T \in R^{ln_\varphi}$$

By using the FRC algorithm, the cluster number l and the corresponding vectors $\mathbf{c} = [c_1, c_2, \ldots, c_l], \boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_l]$ are determined. Then, let the information vector be:

$$\psi(kT) = [\psi_1^T(kT), \psi_2^T(kT), \ldots, \psi_l^T(kT)]^T \tag{20}$$

where

$$\psi_1(kT) := [-\mu_1(kT)y(kT-T), -\mu_1(kT)y(kT-2T), \ldots, -\mu_1(kT)y(kT-n_aT),\ \mu_1(kT)u(kT), \mu_1(kT)u(kT-T), \ldots, \mu_1(kT)u(kT-n_bT),\ \mu_1(kT)u(kT-T+t_1), \ldots, \mu_1(kT)u(kT-n_bT+t_1), \ldots,\ \mu_1(kT)u(kT-T+t_{m-1}), \ldots, \mu_1(kT)u(kT-n_bT+t_{m-1})]^T \in R^{n_\varphi}, n_\varphi = n_a + mn_b + 1$$

$$\psi_2(kT) := [-\mu_2(kT)y(kT-T), -\mu_2(kT)y(kT-2T), \ldots, -\mu_2(kT)y(kT-n_aT),\ \mu_2(kT)u(kT), \mu_2(kT)u(kT-T), \ldots, \mu_2(kT)u(kT-n_bT),\ \mu_2(kT)u(kT-T+t_1), \ldots, \mu_2(kT)u(kT-n_bT+t_1), \ldots,\ \mu_2(kT)u(kT-T+t_{m-1}), \ldots, \mu_2(kT)u(kT-n_bT+t_{m-1})]^T \in R^{n_\varphi}, n_\varphi = n_a + mn_b + 1$$

$$\psi_l(kT) := [-\mu_l(kT)y(kT-T), -\mu_l(kT)y(kT-2T), \ldots, -\mu_l(kT)y(kT-n_aT),\ \mu_l(kT)u(kT),\ \mu_l(kT)u(kT-T), \ldots, \mu_l(kT)\ u(kT-n_bT),\ \mu_l(kT)u(kT-T+t_1), \ldots, \mu_l(kT)u(kT-n_bT+t_1), \ldots,\ \mu_l(kT)u(kT-T+t_{m-1}), \ldots, \mu_l(kT)u(kT-n_bT+t_{m-1})]^T \in R^{n_\varphi}, n_\varphi = n_a + mn_b + 1$$

Then, Eq. (8) is rewritten by Eq. (21). Obviously, it is like the linear system.

$$y(kT) = \psi^T(kT)\boldsymbol{\theta} + v(kT) \tag{21}$$

Then, an RLS algorithm is proposed for the identification of (21).

$$\hat{\theta}(kT) = \hat{\theta}(kT-T) + \mathbf{L}(kT)[y(kT) - \psi^T(kT)\hat{\theta}(kT-T)] \tag{22}$$

$$\mathbf{L}(kT) = \mathbf{P}(kT-T)\psi(kT)[1 + \psi^T(kT)\mathbf{P}(kT-T)\psi(kT)]^{-1} \tag{23}$$

$$\mathbf{P}(kT) = [I - L(kT)\psi^T(kT)]\mathbf{P}(kT-T) \tag{24}$$

Here, $\wedge$ is the symbol of estimation. $\mathbf{I}, \mathbf{P}(kT)$ are the identity matrix and the covariance matrix, respectively.

We can summarize the RLS-FRC algorithm using the following steps:

Step 1 We determine the rule number of the fuzzy model based on an FRC approach.

Step 2 $\mathbf{c}, \boldsymbol{\sigma}$ can be obtained by using Eqs. (16)–(18). Then, we determine the membership function $\mu_q(kT)$ $(q = 1, 2, \ldots, l)$.

Step 3: We initialize the variables as.$k = 1, \hat{\theta}(0) = \mathbf{1}/p_0,\ \mathbf{P}(0) = p_0\mathbf{1}$,

Here, $\mathbf{1}$ is defined as an identity column vector. p_0 is a large number, and we take it as 10^6.

Step 4 By using the inputs/outputs, Eq. (20), $\psi(kT)$ is determined easily.

Step 5 Based on the recursive least squares, $\hat{\theta}(kT)$ is identified by using Eqs. (22)– (24).

4 Simulation example

Taking the pH neutralization reaction as an example for simulation, it can be described in Fig. 2. HNO_3, NaOH and $NaHCO_3$ are reactants in the process, and the pH value is the product. We define F_a, F_b, F_c as the velocity of the flow of the reactant.

We can describe the character of the pH neutralization reaction as:

$$V\frac{\mathrm{d}w_a}{\mathrm{d}t} = F_aC_a - (F_a + F_b)w_a \tag{25}$$

$$V\frac{\mathrm{d}w_b}{\mathrm{d}t} = F_bC_b - (F_a + F_b)w_b \tag{26}$$

Here, we define C_a as the concentration of HNO_3, C_b as the concentration of NaOH, and V as the reactor volume. w_a, w_b are, respectively, charge invariant and material invariant, defined as:

$$w_a = [\mathrm{H}^+] - [\mathrm{OH}^-] - [\mathrm{HCO}_3^-] - 2[\mathrm{CO}_3^{2-}]$$

$$w_b = [\mathrm{H}_2\mathrm{CO}_3] + [\mathrm{HCO}_3^-] + [\mathrm{CO}_3^{2-}]$$

Let $pK_a = -\log_{10} K_a$, and we define $K_a = 17.6 \times 10^{-6}$ as the ionization constant of HNO_3. Furthermore, we can obtain the static titration curve:

$$w_b + 10^{-\mathrm{pH}} - 10^{\mathrm{pH}-14} - \frac{w_a}{1 + 10^{pK_a - \mathrm{pH}}} = 0 \tag{27}$$

The titration curve is shown in Fig. 3.

Obviously, the characteristics of the pH neutralization process are seriously nonlinear.pH neutralization is a process of strong acidity-weak basicity. It can be described as Eqs. (25), (26) and (27). Let us take F_a and F_b as input vectors $u(kT + t_{j-1})$ and consider the pH value as the output vector y_{pH}.

The sample sets $\{F_b, y_{\mathrm{pH}}\}$ were gained from the mechanism model Eqs. (25), (26) and (27). Introducing a variation $[-515, +515]$ in F_b, we can obtain 180 samples. The corresponding parameters of the process and model in the system are listed in Table 1.

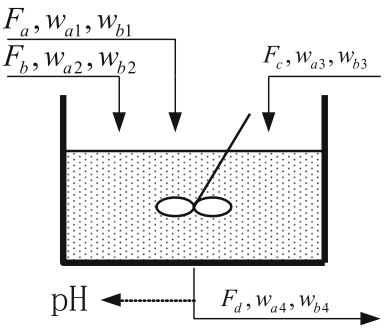

Fig. 2 pH neutralization process

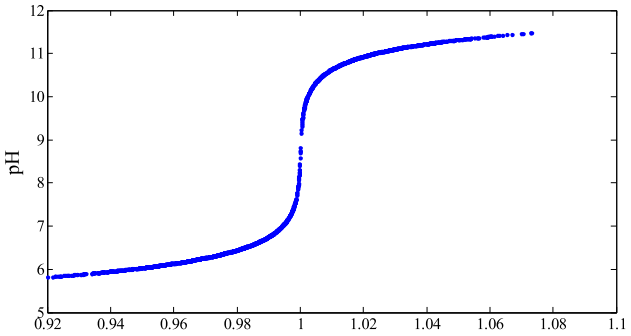

Fig. 3 Titration curve of the process

Table 1 Corresponding parameters

parameter	Value	Parameter	Value
F_a	81 mL/min	$w_a(0)$	0.0435 mol/L
F_b	515 + ζ mL/min	$w_b(0)$	0.0432 mol/L
C_a	0.32 mol/L	V	1000 mL
C_b	0.05 mol/L	T_s	0.5 s
m	2	τ_1	0.5 s
τ_2	1 s	n_a	2
n_b	2		

Obviously, $t_0 = 0s, t_1 = \tau_1 = 0.5s$, $t_2 = \tau_1 + \tau_2 = 1.5s = T$. We collect the NUSDS inputs from F_b. In addition, we collect the sampled data $y(kT)$ from y_{pH}. Then, we can get the input–output sets. Furthermore, the information vector $\phi(kT)$ is obtained.

$$\phi(kT) = [-y(kT - T), -y(kT - 2T), u(kT), u(kT - T), u(kT - 2T), u(kT - T + t_1), u(kT - 2T + t_1)]^T$$

The rule number l and the antecedent parameters $\mathbf{c}, \boldsymbol{\sigma}$ are obtained based on the FRC method. Then, the Gaussian membership function $\mu_q[\varphi(kT], q = 1, 2, \ldots, l$ is constructed, and the information vector $\psi(kT)$ is determined by Eq. (20). After the identification, the outcome is shown the rule number $l = 6$. Then, the system parameter $\boldsymbol{\theta}$ was identified by using the RLS-FRC algorithm. The results of the true value and the prediction are shown in Fig. 4.

To give a fairer measurement of the performance and to eliminate the impact of the randomly generated data. A normalized RMSE (NRMSE) is presented to measure the numerical results.

$$\mathrm{NRMSE} = \frac{\mathrm{RMSE}}{\max_{\{t=1,2,\ldots,\mathrm{num}\}} y_d(t) - \min_{\{t=1,2,\ldots,num\}} y_d(t)} \tag{28}$$

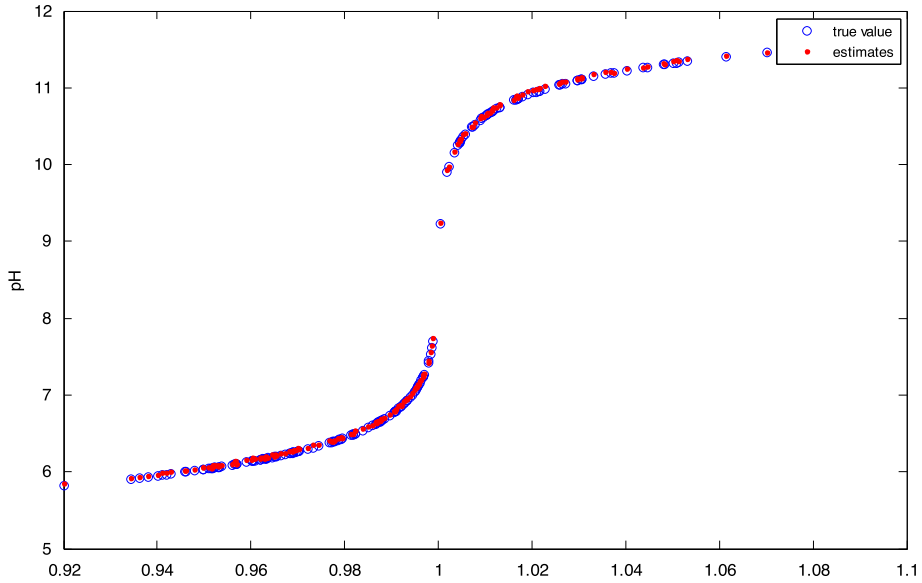

Fig. 4 Results of true value and prediction

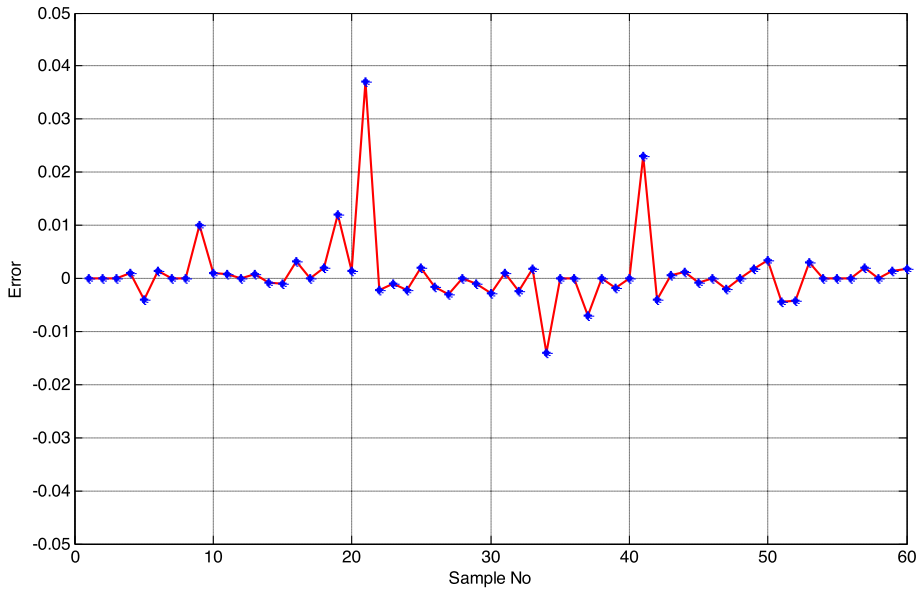

Fig. 5 NRMSE of the RLS-FRC method

Here, num is the scale of data, and $y_d(t), y(t)$ are the true value and the prediction of system outputs, respectively. RMSE is the root mean square error, shown as

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{T} (y_d(t) - y(t))^2}{T}} \tag{29}$$

The range of NRMSE is [0,1]. In system identification, the NRMSE can describe the identification accuracy. If it is smaller, the identification is more accurate. Because NRMSE represents the relative error, it is always used to describe the performance of the algorithm.

The NRMSE of pH identification is 0.1284, while the RMSE is 0.0065. The error curve is shown in Fig. 5.

From the error curve, the prediction result of the RLS-FRC method tracks the true value well. It also depicts the nonlinearity.

5 Conclusions

In this paper, we present an online learning algorithm known as RLS-FRC for calculating the identification of NUSDS. Taking the pH neutralization reaction as an example, the pH value is predicted. The performance of the algorithm is verified.

The whole method is divided into 2 steps. First, the structure and the antecedent parameters of the system are

estimated by using an FRC algorithm. Second, based on an RLS algorithm, consequent parameters can be estimated. The major novelties of RLS-FRC are its new fuzzy modelling for NUSDS. Furthermore, considering the nonlinear character, we can obtain a new model in which the local linear models are combined with nonlinear weighting functions for this special system. The fuzzy rule is determined based on an FRC for I/O sets. This approach meets special clustering requirements. If the system to be identified is smoother, the clustering is coarser. Otherwise, the higher the nonlinearity is, the finer the clustering. Once the cluster rules are determined, the local linear models are identified based on the RLS algorithm. For the subsystems, we used an RLS algorithm, and other methods for linear system identification can also be used. The proposed algorithm is not only applied to NUSDN systems as mentioned before but can also be expanded to other nonlinear systems with aperiodic sampling, such as dual-rate, multirate sampling systems. The algorithm can be further applied to NUSDN systems with MISO, SIMO and MIMO. However, the performance of the presented method requires further analysis and verification.

Acknowledgements Our research was partly financed by the National Natural Science Foundation of China (Grand No. 62003150, 62003151), the Fundamental Science (Natural Science) Research Project of Jiangsu Higher Education Institutions (Grand No. 21KJB120002), Jiangsu Province Policy Guidance Program (Foresighted Joint Research of Industry-University-Research Institute) (2018130), the Open Project of Changzhou Key Laboratory of Industrial Internet Data Intelligence Technology (Nos. CM20190201), the "Blue Project" of Jiangsu Province and Zhongwu Young Innovative Talents.

Data availability All data included in this study are available upon request by contact with the corresponding author.

Declarations

Conflict of interest We declare that we have no conflict of interest.

References

1. Crama P, Schoukens J (2004) Hammerstein–Wiener system estimator initialization. Automatica 40:1543–1550
2. Ding F, Qin L, Chen TW (2009) Reconstruction of continuous-time systems from their non-uniformly sampled discrete-time systems. Automatica 45:324–332
3. Ding F (2013) Hierarchical multi-innovation stochastic gradient algorithm for Hammerstein nonlinear system modeling. Appl Math Model 37:1694–1704
4. Fan Y, Liu X (2022) Auxiliary model-based multi-innovation recursive identification algorithms for an input nonlinear controlled autoregressive moving average system with variable-gain nonlinearity. Int J Adapt Control Signal Process 36(3):521–540
5. Kordestani M, Dehghani M, Moshiri B, Saif M (2020) A new fusion estimation method for multi-rate multi-sensor systems with missing measurements. IEEE Access 8:47522–47532
6. Li BH, Jesús GM, Eusebio V (2022) A dynamic mode decomposition technique for the analysis of non–uniformly sampled flow data. J Comput Phys 468:1–17
7. Liu YJ, Xie L, Ding F (2009) An auxiliary model based recursive least squares parameter estimation algorithm for non-uniformly sampled multirate systems. Proc Inst Mech Eng Part I J Syst Control Eng 223:445–454
8. Liu Y, Ding F, Shi Y (2012) Least squares estimation for a class of non-uniformly sampled systems based on the hierarchical identification principle. Circuits Syst Signal Process 31(6):1985–2000
9. Liu RR, Pan TH, Chen S, Li ZM (2017) Identification of non-uniformly sampled Wiener systems with dead-zone non-linearities. Math Comput Model Dyn Syst 23(6):595–612
10. Liu RR, Pan TH, Li ZM (2017) Multi-model recursive identification for nonlinear systems with non-uniformly sampling. Clust Comput 20(1):25–32
11. Ma JX, Chen J, Xiong WL, Ding F (2018) Expectation maximization estimation algorithm for Hammerstein models with non-Gaussian noise and random time delay from dual-rate sampled-data. Digit Signal Process 73:135–144
12. Milanese M, Novara C, Pivano L (2005) Structured SM identification of vehicle vertical dynamics. Math Comput Model Dyn Syst 11:195–207
13. Shen YX, Wang ZD, Shen B, Alsaadi FE (2020) H$\propto$ state estimation for multi-rate artificial neural networks with integral measurements: a switched system approach. Inf Sci 539:434–446
14. Shiki SB, Hansen C, Silva SD (2023) Practical applications for nonlinear system identification using discrete-time Volterra series. J Braz Soc Mech Sci Eng 45(2):87
15. Wills A, Schon TB, Ljung L, Ninness B (2013) Identification of Hammerstein–Wiener models. Automatica 49:70–81
16. Wang XH, Ding F (2016) Modelling and multi-innovation parameter identification for Hammerstein nonlinear state space systems using the filtering technique. Math Comput Model Dyn Syst 22:113–140
17. Wang HW, Xie LR (2021) Fuzzy modeling of non-uniformly sampling nonlinear systems based on clustering method and convergence analysis. J Syst Sci Complex 34:502–519
18. Xie L, Yang H (2011) Gradient-based iterative identification for nonuniform sampling output error systems. J Vib Control 17(3):471–478
19. Zand B, Ghaderi P, Amini F (2023) Structural system identification via synchronization technique and fuzzy logic. Math Comput Simul 203:174–188