

# 系统辨识与物理信息神经网络融合的非线性智能建模方法

张艺楠, 牛宏<sup>†</sup>

(辽宁石油化工大学理学院, 辽宁抚顺 113001)

**摘要:** 针对一类非线性动态系统, 提出一种带有遗忘因子的递推最小二乘法与物理信息神经网络融合的非线性智能建模方法, 实现了非线性系统的动态建模. 采用带有遗忘因子的递推最小二乘法对低阶线性模型的未知参数进行辨识, 通过逐步更新模型参数, 有效应对非线性系统中的动态变化以及参数跳变的情况, 并根据数据信息逐步修正模型, 提高精度. 在获得低阶线性模型的基础上, 引入物理信息神经网络对未建模动态未知增量进行估计. 将机理模型的部分先验知识作为约束条件嵌入物理信息神经网络中, 使神经网络能够快速收敛到最优解, 提高建模精度. 该方法弥补了实际工业过程中数据样本不足或数据损坏的建模需求. 最后, 通过数值仿真对比实验验证了所提方法的有效性.

**关键词:** 物理信息神经网络; 非线性系统; 未建模动态; 递推最小二乘法; 深度学习; 复杂工业过程

**中图分类号:** TP319 **文献标志码:** A

**DOI:** 10.13195/j.kzyjc.2025.1148

**引用格式:** 张艺楠, 牛宏. 系统辨识与物理信息神经网络融合的非线性智能建模方法 [J]. 控制与决策.

## A nonlinear intelligent modeling method combining system identification and physics-informed neural network

ZHANG Yi-nan, NIU Hong<sup>†</sup>

(College of Sciences, Liaoning Petrochemical University, Fushun 113001, China)

**Abstract:** A nonlinear intelligent modeling method is proposed for a class of nonlinear dynamic systems, integrating the recursive least squares method with a forgetting factor and a physical information neural network. This approach facilitates the dynamic modeling of nonlinear systems. The recursive least squares method with a forgetting factor is employed to identify the unknown parameters of a low-order linear model. By incrementally updating the model parameters, the method effectively addresses dynamic changes and parameter jumps within the nonlinear system, gradually refining the model based on data information to enhance accuracy. Following the establishment of the low-order linear model, a physical information neural network is utilized to estimate the unmodeled dynamic unknown increment. Prior knowledge of the mechanism model is incorporated as constraint conditions within the physical information neural network, allowing for rapid convergence to the optimal solution and improved modeling accuracy. This method addresses the challenges posed by insufficient data samples or data corruption in actual industrial processes. The effectiveness of the proposed method is ultimately validated through numerical simulations and comparative experiments.

**Keywords:** physics-informed neural network; nonlinear systems; unmodeled dynamics; recursive least squares method; deep learning; complex industrial processes

## 0 引言

复杂系统往往呈现强非线性、机理不清等特性, 其精确数学模型的构建一直是一项极具挑战性的难题. 常用的最小二乘法通过最小化估计误差的平方和来确定模型参数. 极大似然估计法在给定一组观测数据的情况下估计模型参数<sup>[1-4]</sup>. 文<sup>[5]</sup>采用自回

归滑动平均模型来表征系统的非线性特性, 建立系统的动态关系, 但易导致模型欠拟合或过拟合. 文<sup>[6]</sup>提出核递推最小二乘法, 提升了系统的抗干扰能力. 这些传统的系统辨识建模方法通常需要依赖于一些特定的假设, 如线性、高斯分布等, 但在实际应用中, 系统往往受到非线性、非高斯分布等不可控因素的

收稿日期: 2025-11-05; 录用日期: 2026-03-18.

基金项目: 国家自然科学基金 (61603168).

责任编辑: 李忠奎.

<sup>†</sup>通信作者. E-mail: 13942393197@163.com.

影响,这就导致辨识效果不佳.同时,传统建模方法依赖于大量数据,数据的数量及质量直接影响建模精度.

文<sup>[7]</sup>系统阐述了机理建模、数据驱动建模及混合建模三大建模方法,明确指出融合系统辨识与深度学习的混合建模框架在捕捉工业过程动态特性中具备显著优势.近年来,随着计算机科学与技术的持续革新,系统辨识在原有辨识算法的基础上与机器学习、深度学习等人工智能方法相融合,使系统辨识能够应对的辨识对象范围更大,辨识精度更高.文<sup>[8]</sup>首次将神经网络用于系统辨识,建立了能够较准确描述非线性系统动态行为的智能模型.目前,BP(back-propagation, BP)神经网络、长短时记忆网络(long short memory, LSTM)以及循环神经网络(recurrent neural network, RNN)等广泛应用于系统辨识<sup>[9-12]</sup>.文<sup>[13]</sup>提出一种交替辨识算法,并在电熔镁炉功率预报过程中验证了其有效性.文<sup>[14]</sup>则针对电熔镁炉冶炼过程,将系统辨识与深度学习相结合,实现了用电需量的高精度预报.文<sup>[15]</sup>提出了利用带有遗忘因子的递推最小二乘法与随机配置网络相结合的建模方法.文<sup>[16]</sup>设计了一种在线鲁棒自适应系统辨识方法,提高了监控精度.文<sup>[17]</sup>利用 FFRLS 算法在线精确估计锂电池电路模型中的时变参数.文<sup>[18]</sup>利用 RNN 辨识非线性系统动态行为,但传统 RNN 在处理慢动态系统时易出现梯度爆炸或梯度消失问题,导致辨识精度下降.文<sup>[19]</sup>融合系统辨识与 LSTM 的数字孪生建模法,结合云边协同自校正控制策略,突破了传统建模方法局限.

物理信息神经网络(physics-informed neural network, PINN)将物理定律作为约束条件嵌入到神经网络的损失函数中,通过适当惩罚神经网络的损失函数,加强了模型对物理定律的遵循,从而提高求解微分方程的准确率和效率.同时,由于物理定律的约束,PINN在较少数据或者数据有所缺失的情况下同样可以获得良好结果.文<sup>[20]</sup>提出了基于 PINN 的深度稀疏回归网络,采用隐式控制方程来校正神经网络.文<sup>[21]</sup>通过引入放缩系数将动力学模型嵌入到神经网络的损失函数中,保证了网络输出与力学原理的一致性.文<sup>[22]</sup>提出了一种基于 PINN 的参数辨识算法,大幅减少数据量的需求.文<sup>[23-25]</sup>将非线性系统拆分为低阶线性部分与未建模动态未知增量部分,为复杂非线性系统的精确建模提供了新途径.

本文融合 FFRLS 与 PINN 进行智能建模.首先采集系统的输入输出数据,通过 FFRLS 算法辨识低阶模型的未知参数,再利用 PINN 对非线性系统的

未建模动态未知增量进行估计,将均方误差(mean squared error, MSE)作为先验知识嵌入 PINN 的损失函数中,使模型不仅能够符合数据特征,又能确保建模过程与先验知识规律的一致性,从而进一步提高建模精度.通过数值仿真对比实验,验证了所提方法的有效性.

## 1 问题描述

一类复杂工业过程的动态模型可以描述为

$$y(k) = f[\varphi(k-d)]. \quad (1)$$

其中  $\varphi(k-d) = [y(k-1), \dots, y(k-n_A), u(k-d), \dots, u(k-d-n_B)]^T$  为数据向量;  $f(\cdot)$  为非线性函数;  $y(k)$ 、 $u(k)$  为输出和输入;  $d \geq 1$  为时滞;  $n_A$ 、 $n_B$  为模型阶次.

将系统(1)在原点 Taylor 展开,可得

$$\begin{aligned} a_i &= \left. \frac{\partial f[\varphi(k-d)]}{\partial y(k-i)} \right|_{y=0, u=0} \quad i = 1, \dots, n_A \\ b_j &= \left. \frac{\partial f[\varphi(k-d)]}{\partial u(k-j)} \right|_{y=0, u=0} \quad j = 1, \dots, n_B. \end{aligned} \quad (2)$$

与文<sup>[13]</sup>类似,设  $d = 1$ ,则系统(1)可表示为

$$\begin{aligned} A(z^{-1})y(k+1) &= B(z^{-1})u(k) + \\ &v[\varphi(k-1)] + \Delta v[\varphi(k)] \end{aligned} \quad (3)$$

其中

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + \dots + a_{n_A} z^{-n_A} \\ B(z^{-1}) &= b_0 + b_1 z^{-1} + \dots + b_{n_B} z^{-n_B} \end{aligned} \quad (4)$$

$v[\varphi(k)]$  为高阶非线性函数,称为未建模动态,具体表现为实际系统中未被低阶线性模型捕捉的高阶非线性效应、随机扰动、时变特性等动态成分,  $\Delta v[\varphi(k)]$  为其未知增量.

将上式等价表示为

$$\begin{aligned} y(k+1) &= -a_1 y(k) - a_2 y(k-1) - \dots - b_0 u(k) + \\ &b_1 u(k-1) + \dots + b_{n_B} u(k-n_B) + \\ &v[\varphi(k-1)] + \Delta v[\varphi(k)] = \\ &[y(k), \dots, y(k-n_A+1), u(k-1), \dots, \\ &u(k-n_B)] \cdot [-a_1, \dots, -a_{n_A}, b_1, \dots, b_{n_B}] + \\ &v[\varphi(k-1)] + \Delta v[\varphi(k)] = \\ &\varphi^T(k)\theta + v[\varphi(k-1)] + \Delta v[\varphi(k)] \end{aligned} \quad (5)$$

其中,  $\varphi^T(k)\theta$  为系统低阶线性部分,  $\varphi(k)$  为数据向量,  $\theta$  为待估参数向量,即

$$\begin{aligned} \varphi(k) &= [y(k), \dots, y(k-n_A), u(k), \dots, u(k-n_B)]^T \\ \theta &= [-a_1, -a_2, \dots, -a_{n_A}, b_0, \dots, b_{n_B}]^T \end{aligned}$$

## 2 低阶线性模型构建

在具有强非线性特征的系统建模问题中,传统递推最小二乘法无法精确捕捉其非线性关系,旧数据会淹没新数据的特征,参数突变、噪声干扰等情况

也会导致参数震荡. 因此, 本文引入 FFRLS 算法对系统低阶线性模型的未知参数进行辨识. 在辨识过程中, 遗忘因子会在赋予新数据更高权重的同时利用指数衰减旧数据的权重, 迅速降低旧数据对模型参数辨识的影响, 使模型能够更快速的收敛到新的参数.

由式 (5) 可知, 系统低阶线性部分的参数辨识方程为

$$\bar{y}(k+1) = \varphi^T(k)\theta. \quad (6)$$

其中,  $\bar{y}(k+1)$  为低阶模型的输出.

式 (6) 对应的参数估计方程为

$$\hat{y}(k+1) = \varphi^T(k)\hat{\theta}(k). \quad (7)$$

其中  $\hat{\theta}(k) = [-\hat{a}_1, -\hat{a}_2, \dots, -\hat{a}_{n_A}, \hat{b}_0, \dots, \hat{b}_{n_B}]^T$ ,  $\hat{y}(k+1)$  为 FFRLS 算法辨识得到的模型输出,  $\hat{\theta}(k)$  为  $k$  时刻对  $\theta$  的估计值, 采用如下 FFRLS 算法进行估计

$$\begin{cases} \hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[y(k) - \varphi^T(k-1)\hat{\theta}(k-1)] \\ K(k) = \frac{P(k-1)\varphi(k)}{\lambda + \varphi^T(k)P(k-1)\varphi(k)} \\ P(k) = \frac{1}{\lambda}[I - K(k)\varphi^T(k)]P(k-1) \end{cases} \quad (8)$$

其中, 遗忘因子  $0.9 < \lambda < 1$ ;  $K(k)$  为预测误差的加权矩阵;  $P(k)$  为可逆的协方差矩阵;  $I$  为与  $P(k)$  同维的单位矩阵;  $\hat{\theta}(0)$  与  $P(0)$  为参数的初值, 设置初值为

$$\begin{cases} \hat{\theta}(0) = \varepsilon \\ P(0) = \alpha I \end{cases}$$

其中,  $\alpha$  为充分大的正实数 ( $10^4 \sim 10^{10}$ ),  $\varepsilon$  为零向量或充分小的正实向量.

### 3 PINN 非线性辨识方法

传统神经网络在没有先验知识的情况下, 只能看作是黑箱模型中的一种经典类型, 并且传统的神经网络训练往往需要大量数据, 这使得传统神经网络建模的性能严重依赖于训练数据的数量以及质量. 而 PINN 是一种将神经网络与先验知识相结合的网络架构, 它利用了深度神经网络作为函数逼近器的能力, 在无需预先假设或线性化处理系统的情况下应对非线性问题. 由于 PINN 中损失函数中的先验知识是可视的, 通过调整这种损失函数及其权重, 能够实现在少量样本数据或数据缺失的条件下更好的应对非线性系统建模.

由于  $\Delta v[\varphi(k)]$  未知, 需对  $k$  时刻的  $\Delta v[\varphi(k)]$

进行估计. 利用 FFRLS 算法获得参数的估计值  $\hat{\theta}(k)$ , 从而得到采用 PINN 估计  $\Delta v[\varphi(k)]$  的导师信号, 再采用 PINN 对  $\Delta v[\varphi(k)]$  进行估计, 得到其估计值  $\Delta \hat{v}[\varphi(k)]$ . 则  $k$  时刻 PINN 导师信号为

$$\Delta \bar{v}[\varphi(k)] = y(k+1) - \varphi^T(k)\theta - v[\varphi(k-1)]. \quad (9)$$

由式 (5) 可知, 带有未建模动态未知增量的参数辨识方程为

$$\Delta v[\varphi(k)] = y(k+1) - \varphi^T(k)\theta - v[\varphi(k-1)]. \quad (10)$$

式 (11) 对应的参数估计方程为

$$\Delta \hat{v}[\varphi(k)] = \hat{y}(k+1) - \varphi^T(k)\hat{\theta} - \hat{v}[\varphi(k-1)]. \quad (11)$$

利用数据向量  $\varphi(k)$  与导师信号  $\Delta \bar{v}[\varphi(k)]$  对 PINN 进行训练与校正, 进而得到其下一时刻的估计值  $\Delta \hat{v}[\varphi(k+1)]$ .

#### 3.1 网络架构设计

PINN 网络架构如图 1 所示

在图 1 中, PINN 网络主要包含前向传播和反向传播两个部分, 神经元分层排列, 由输入层、隐藏层和输出层组成, 网络各层的参数由权重和偏置构成.

1) 输入层的输入为  $\varphi(k) = [y(k-1), \dots, y(k-n_A), u(k-1), \dots, u(k-1-n_B)]^T$ , 对  $\varphi(k)$  进行标准化处理

$$\varphi_{scaled}(k) = \frac{\varphi(k) - \mu_\varphi}{\sigma_\varphi}. \quad (12)$$

其中,  $\varphi_{scaled}(k)$  为标准化数据向量,  $\mu_\varphi$  为均值向量,  $\sigma_\varphi$  为标量标准差

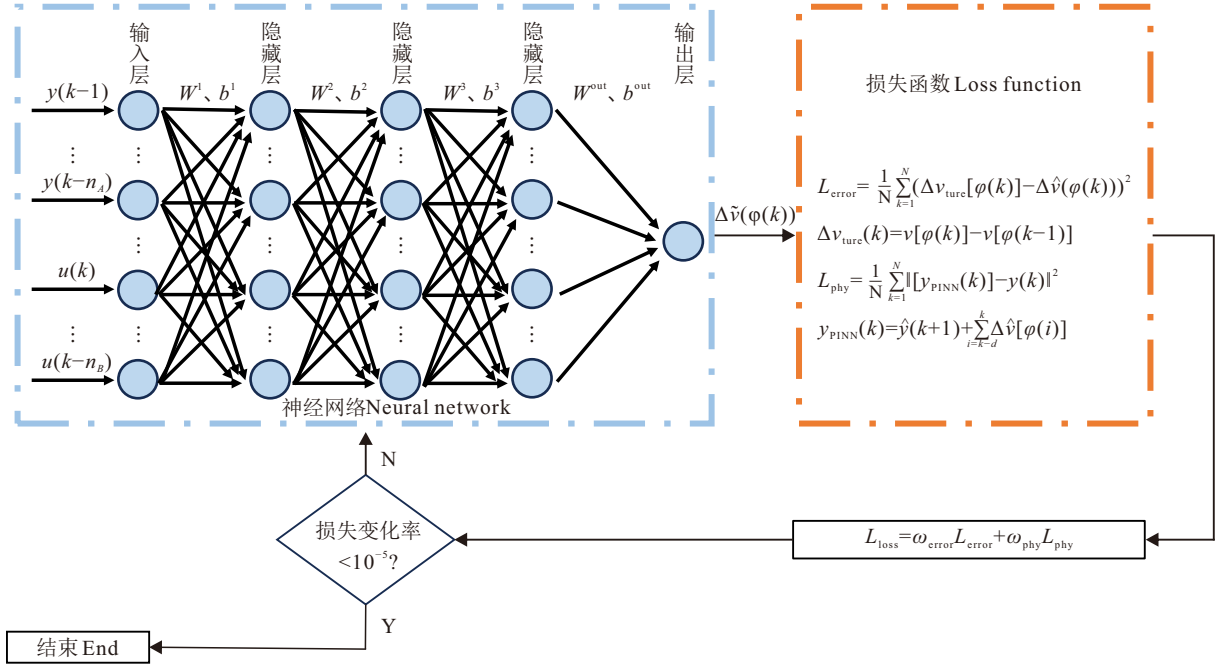
$$\begin{cases} \mu_\varphi = [\underbrace{\mu_y, \mu_y, \dots, \mu_y}_{n_A+1}, \underbrace{\mu_u, \mu_u, \dots, \mu_u}_{n_B+1}]^T \\ \sigma_\varphi = \sqrt{(n_A+1)\sigma_y^2 + (n_B+1)\sigma_u^2} \end{cases} \quad (13)$$

其中,  $\mu_y$  为输出序列  $y(k)$  的均值,  $\mu_u$  为输入序列  $u(k)$  的均值,  $\sigma_y^2$  为输出序列  $y(k)$  的方差,  $\sigma_u^2$  为输入序列  $u(k)$  的方差.

$$\begin{cases} \mu_y = \frac{1}{N} \sum_{n=1}^N y(n) \\ \mu_u = \frac{1}{N} \sum_{n=1}^N u(n) \\ \sigma_y^2 = \frac{1}{N} \sum_{n=1}^N (y(n) - \mu_y)^2 \\ \sigma_u^2 = \frac{1}{N} \sum_{n=1}^N (u(n) - \mu_u)^2 \end{cases} \quad (14)$$

2) 隐藏层选择 3 层全连接层, 每层 100 个节点, 采用双曲正切 (tanh) 激活函数.

第 1 隐藏层的链接权值更新方式如下

图1 PINN 估计  $\Delta v[\varphi(k)]$  的结构

$$\begin{cases} Z^1(k) = W^1 \varphi_{scaled}(k) + b^1 \\ H^1(k) = \tanh(Z^1(k)) \\ b^1 \in \mathbb{R}^{100} \end{cases} \quad (15)$$

$W^1$  为输入层到隐藏层的链接权重矩阵

$$W^1 = \begin{bmatrix} w_{1,1}^1 & \cdots & w_{1,n_A+n_B+1}^1 \\ \vdots & \ddots & \vdots \\ w_{100,1}^1 & \cdots & w_{100,n_A+n_B+1}^1 \end{bmatrix}. \quad (16)$$

$b^1$  为偏置向量

$$b^1 = [b_1^1, \dots, b_{100}^1]^T. \quad (17)$$

$Z^1(k)$  为第 1 隐藏层的加权输入矩阵

$$Z^1(k) = [z_1^1(k), \dots, z_i^1(k), \dots, z_{100}^1(k)]^T. \quad (18)$$

其中,  $z_i^1(k)$  为第 1 隐藏层的第  $i$  个神经元加权输入

$$z_i^1(k) = \sum_{j=1}^{n_A+n_B+1} w_{i,j}^1 \varphi_{scaled_j}(k) + b_i^1, \quad i = 1, \dots, 100, j = 1, \dots, n_A + n_B + 1. \quad (19)$$

其中,  $w_{i,j}^1$  表示输入层第  $j$  个神经元与第 1 隐藏层的第  $i$  个神经元之间的链接权重.  $H^1(k)$  为第 1 隐藏层的神经元激活输出矩阵

$$H^1(k) = [h_1^1(k), \dots, h_i^1(k), \dots, h_{100}^1(k)]^T. \quad (20)$$

其中,  $h_i^1(k)$  为第 1 隐藏层的第  $i$  个神经元激活输出

$$h_i^1(k) = \tanh(z_i^1(k)) = \tanh\left(\sum_{j=1}^{n_A+n_B+1} w_{i,j}^1 \varphi_{scaled_j}(k)\right), i = 1, \dots, 100. \quad (21)$$

$H_d^1(k)$  为第 1 隐藏层激活输出矩阵的丢弃层

$$H_d^1(k) = Dropout(H^1(k), p = 0.2) = [h_{d,1}^1(k), \dots, h_{d,100}^1(k)]^T. \quad (22)$$

其中,  $h_{d,i}^1(k)$  为第 1 隐藏层的第  $i$  个神经元激活输出的丢弃输出

$$h_{d,i}^1(k) = Dropout(h_i^1(k), p = 0.2), \quad i = 1, \dots, 100. \quad (23)$$

其中,  $p$  为失活概率.

第 2、3 隐藏层的链接权值更新方式如下

$$\begin{cases} Z^l(k) = W^l H_d^{l-1}(k) + b^l \\ H^l(k) = \tanh(Z^l(k)) \\ b^l \in \mathbb{R}^{100} \end{cases}, l = 2, 3 \quad (24)$$

$W^l = [W^2, W^3]$  为第  $l-1$  隐藏层到第  $l$  隐藏层之间的链接权重矩阵

$$W^l = \begin{bmatrix} w_{1,1}^l & \cdots & w_{1,100}^l \\ \vdots & \ddots & \vdots \\ w_{100,1}^l & \cdots & w_{100,100}^l \end{bmatrix}, l = 2, 3 \quad (25)$$

$b^l$  为偏置向量

$$b^l = [b_1^l, \dots, b_{100}^l]^T, l = 2, 3 \quad (26)$$

$Z^l(k)$  为第  $l$  隐藏层的神经元加权输入矩阵

$$Z^l(k) = [z_1^l(k), \dots, z_i^l(k), \dots, z_{100}^l(k)]^T, \quad l = 2, 3, i = 1, \dots, 100. \quad (27)$$

其中,  $z_i^l(k)$  为第  $l$  隐藏层第  $i$  个神经元输入

$$z_i^l(k) = \sum_{j=1}^{100} w_{i,j}^l h_{d,i}^{l-1}, l = 2, 3, i = 1, \dots, 100. \quad (28)$$

$H^l(k)$  为第  $l$  隐藏层神经元的激活输出矩阵

$$H^l(k) = [h_1^l(k), \dots, h_i^l(k), \dots, h_{100}^l(k)]^T, \quad (29)$$

$$l = 2, 3, i = 1, \dots, 100.$$

其中,  $h_i^l(k)$  为第  $l$  隐藏层的第  $i$  个神经元激活输出

$$h_i^l(k) = \tanh(z_i^l(k)), \quad l = 2, 3, i = 1, \dots, 100. \quad (30)$$

$H_d^l(k)$  为第  $l$  隐藏层激活输出矩阵的丢弃层

$$H_d^l(k) = [h_{d,1}^l(k), \dots, h_{d,100}^l(k)]^T, \quad l = 2, 3. \quad (31)$$

其中,  $h_{d,i}^l(k)$  为第  $l$  隐藏层第  $i$  个神经元激活输出的丢弃输出

$$h_{d,i}^l(k) = \text{Dropout}(h_i^l(k), p = 0.2), \quad (32)$$

$$l = 2, 3, i = 1, \dots, 100.$$

3) 输出层选择 1 个节点估计未建模动态未知增量. 更新方式如下

$$\Delta \hat{v}[\varphi(k)] = W^{\text{out}} H_d^3(k) + b^{\text{out}}. \quad (33)$$

其中,  $W^{\text{out}}$  为第 3 隐藏层到输出层的链接权重矩阵

$$W^{\text{out}} = [w_1^{\text{out}}, \dots, w_{100}^{\text{out}}]^T. \quad (34)$$

$b^{\text{out}}$  为偏置向量

$$b^{\text{out}} = [b_1^{\text{out}}, \dots, b_{100}^{\text{out}}]^T. \quad (35)$$

则式 (34) 可表示为

$$\Delta \hat{v}[\varphi(k)] = \sum_{i=1}^{100} W_i^{\text{out}} h_{d,i}^3 + b_i^{\text{out}}. \quad (36)$$

### 3.2 损失函数设计

在利用 PINN 求解方程时, 首先构建前向传播神经网络逼近方程 (5) 的隐藏解, 得到  $k$  时刻  $\Delta \hat{v}[\varphi(k)]$  的值, 然后利用 MSE 构建物理约束条件, 并将该约束条件转化为损失函数项. 网络损失函数在加入物理约束项后的总损失函数可以表示为

$$L_{\text{loss}} = \omega_{\text{error}} L_{\text{error}} + \omega_{\text{phy}} L_{\text{phy}}. \quad (37)$$

数据拟合损失

$$\begin{cases} L_{\text{error}} = \frac{1}{N} \sum_{k=1}^N (\Delta v_{\text{true}}(k) - \Delta \hat{v}[\varphi(k)])^2 \\ \Delta v_{\text{true}}[\varphi(k)] = v[\varphi(k)] - v[\varphi(k-1)] \end{cases} \quad (38)$$

物理约束损失

$$\begin{cases} L_{\text{phy}} = \frac{1}{N} \sum_{k=1}^N (y_{\text{PINN}}(k) - y(k))^2 \\ y_{\text{PINN}}(k) = \hat{y}(k+1) + \sum_{i=k-d}^k \Delta \hat{v}[\varphi(i)] \end{cases} \quad (39)$$

其中,  $L_{\text{loss}}$  为前向传播神经网络的损失函数;  $L_{\text{error}}$  为估计值与真实值之间的误差;  $\omega_{\text{error}}$  为  $L_{\text{error}}$  的权重;  $L_{\text{phy}}$  为物理约束条件, 选择 MSE 作为物理约束条件函数;  $\omega_{\text{phy}}$  为  $L_{\text{phy}}$  的权重.

### 3.3 训练算法与梯度计算

#### 3.3.1 总损失对权重 $W^l$ 的梯度

$$\frac{\partial L_{\text{loss}}}{\partial W^{\text{out}}} = \omega_{\text{error}} \frac{\partial L_{\text{error}}}{\partial W^{\text{out}}} + \omega_{\text{phy}} \frac{\partial L_{\text{phy}}}{\partial W^{\text{out}}}. \quad (40)$$

其中

$$\begin{cases} \frac{\partial L_{\text{error}}}{\partial W^{\text{out}}} = \frac{2}{N} \sum_{k=1}^N (\Delta \hat{v}[\varphi(k)] - \Delta v_{\text{true}}[\varphi(k)]) \cdot (H_d^3(k))^T \\ \frac{\partial L_{\text{phy}}}{\partial W^{\text{out}}} = \frac{2}{N} \sum_{k=1}^N (y_{\text{PINN}}(k) - y(k)) \cdot (H_d^3(k))^T \end{cases} \quad (41)$$

令  $\Delta \hat{v}[\varphi(k)] - \Delta v_{\text{true}}[\varphi(k)] = \varepsilon_k$ ,  $y_{\text{PINN}}(k) - y(k) = \delta_k$ , 则式 (42) 为

$$\begin{cases} \frac{\partial L_{\text{error}}}{\partial W^{\text{out}}} = \frac{2}{N} \sum_{k=1}^N \varepsilon_k \cdot (H_d^3(k))^T \\ \frac{\partial L_{\text{phy}}}{\partial W^{\text{out}}} = \frac{2}{N} \sum_{k=1}^N \delta_k \cdot (H_d^3(k))^T \end{cases} \quad (42)$$

则总损失梯度为

$$\begin{aligned} \frac{\partial L_{\text{loss}}}{\partial W^{\text{out}}} &= \omega_{\text{error}} \cdot \frac{2}{N} \sum_{k=1}^N \varepsilon_k \cdot (H_d^3(k))^T + \\ &\omega_{\text{phy}} \cdot \frac{2}{N} \sum_{k=1}^N \delta_k \cdot (H_d^3(k))^T. \end{aligned} \quad (43)$$

#### 3.3.2 隐藏层反向传播

1) 第 3 隐藏层 ( $l = 3$ ) 更新方式如下

输出层对第 3 层隐藏层的梯度

$$\frac{\partial L_{\text{loss}}}{\partial H_d^3(k)} = (W^{\text{out}})^T (\omega_{\text{error}} \frac{2}{N} \varepsilon_k + \omega_{\text{phy}} \frac{2}{N} \delta_k). \quad (44)$$

通过激活函数的梯度

$$\frac{\partial L_{\text{loss}}}{\partial Z^3(k)} = \frac{\partial L_{\text{loss}}}{\partial H_d^3(k)} \odot (1 - \tanh^2(Z^3(k))). \quad (45)$$

第 3 隐藏层权重矩阵和偏置矩阵的梯度

$$\begin{cases} \frac{\partial L_{\text{loss}}}{\partial W^3} = \frac{\partial L_{\text{loss}}}{\partial Z^3(k)} \cdot (H_d^2(k))^T \\ \frac{\partial L_{\text{loss}}}{\partial b^3} = \frac{\partial L_{\text{loss}}}{\partial Z^3(k)} \end{cases} \quad (46)$$

2) 第 2 隐藏层 ( $l = 2$ ) 更新方式如下

第 3 隐藏层对第 2 隐藏层的梯度

$$\frac{\partial L_{\text{loss}}}{\partial H_d^2(k)} = (W^3)^T \frac{\partial L_{\text{loss}}}{\partial Z^3(k)}. \quad (47)$$

通过激活函数的梯度

$$\frac{\partial L_{\text{loss}}}{\partial Z^2(k)} = \frac{\partial L_{\text{loss}}}{\partial H_d^2(k)} \odot (1 - \tanh^2(Z^2(k))). \quad (48)$$

第 2 隐藏层权重矩阵和偏置矩阵的梯度

$$\begin{cases} \frac{\partial L_{loss}}{\partial W^2} = \frac{\partial L_{loss}}{\partial Z^2(k)} \cdot (H_d^1(k))^T \\ \frac{\partial L_{loss}}{\partial b^2} = \frac{\partial L_{loss}}{\partial Z^2(k)} \end{cases} \quad (49)$$

3) 第1隐藏层( $l=1$ )更新方式如下

第2隐藏层对第1隐藏层的梯度

$$\frac{\partial L_{loss}}{\partial H_d^1(k)} = (W^2)^T \frac{\partial L_{loss}}{\partial Z^2(k)}. \quad (50)$$

通过激活函数的梯度

$$\frac{\partial L_{loss}}{\partial Z^1(k)} = \frac{\partial L_{loss}}{\partial H_d^1(k)} \odot (1 - \tanh^2(Z^1(k))). \quad (51)$$

第1隐藏层权重矩阵和偏置矩阵的梯度

$$\begin{cases} \frac{\partial L_{loss}}{\partial W^1} = \frac{\partial L_{loss}}{\partial Z^1(k)} \cdot \varphi(k)^T \\ \frac{\partial L_{loss}}{\partial b^1} = \frac{\partial L_{loss}}{\partial Z^1(k)} \end{cases} \quad (52)$$

### 3.3.3 Adma 优化器

采用 Adma 优化器进行梯度下降算法, 利用反向传播得到的梯度来调整网络参数.

通过反向传播得到第 $l$ 层隐藏层的权重  $W^l$  的梯度

$$g_l = \frac{\partial L_{loss}}{\partial W^l}. \quad (53)$$

其中,  $g_l$  为第 $l$ 隐藏层权重  $W^l$  的梯度.

一阶矩估计  $m_l$ 、二阶矩估计  $v_l$  更新方式为

$$\begin{cases} m_l = \beta_1 \cdot m_{l-1} + (1 - \beta_1) \cdot g_l \\ v_l = \beta_2 \cdot v_{l-1} + (1 - \beta_2) \cdot g_l^2 \end{cases} \quad (54)$$

其中,  $\beta_1$  为一阶矩衰减率;  $\beta_2$  为二阶矩衰减率.

偏差校正  $\hat{m}_l$ 、 $\hat{v}_l$  更新方式为

$$\hat{m}_l = \frac{m_l}{1 - \beta_1^t}, \quad \hat{v}_l = \frac{v_l}{1 - \beta_2^t}. \quad (55)$$

其中,  $1 - \beta_1^t$  为 $t$ 时刻的一阶矩衰减率;  $1 - \beta_2^t$  为 $t$ 时刻的二阶矩衰减率. 参数  $W^l$  更新方式为

$$W^l \leftarrow W^l - \eta_l \cdot \frac{\hat{m}_l}{\sqrt{\hat{v}_l + \varepsilon}}. \quad (56)$$

其中,  $\varepsilon$  为无限小的正实数, 称为平滑项. 本文选定初始学习率为  $\eta_0 = 1$ , 并用指数衰减:  $\eta_l = \eta_0 \cdot 0.995^l$ ; 初始一阶矩 (动量) 为  $m_0 = 0$ ; 一阶矩衰减率为  $\beta_1 = 0.9$ ; 二阶矩 (自适应学习率) 为  $v_0 = 0$ ; 二阶矩衰减率为,  $\beta_2 = 0.99$ ; 平滑项为  $\varepsilon = 10^{-8}$ .

### 3.4 PINN 关键结构参数分析

为确定最优网络结构, 与文<sup>[26]</sup>类似, 分别选取隐藏层为2、3、4层, 节点数为50、100、150个的组合进行对比实验, 实验结果如表1所示

由表1可知, 2层隐藏层150个节点组合的

表1 不同网络结构性能评价

隐藏层层数	节点数	MAE	训练时间(s)
2	50	0.0070	44.7552
2	100	0.0051	43.4357
2	150	0.0004	42.2802
3	50	0.0041	41.6148
3	100	0.0011	42.0716
3	150	0.0010	42.9306
4	50	0.0001	43.2273
4	100	0.0001	43.3831
4	150	0.0012	45.5292

MAE 值虽低, 但其 MAE 波动幅度达 12%, 泛化性较差, 且 2 层网络的容量相对有限, 易出现拟合不充分问题; 4 层隐藏层结果训练精度最佳, 但网络复杂度冗余, 存在过拟合风险, 且训练时间相较于 3 层隐藏层 100 个节点增加 12%, 实时建模效率过低; 而 3 层隐藏层 100 个节点的训练结果不仅其 MAE 值满足建模精度要求, 同时训练时间也符合工业场景的实时性需求, 能够在拟合能力、泛化性能与计算效率之间实现平衡, 故本文最终选择 3 层隐藏层, 100 个节点的网络结构.

### 3.5 动态触发式交替辨识策略

采用交替辨识策略, 所提算法设定基础交替周期为 5 次, 即每完成 5 次 FFRLS 参数辨识, 触发一次 PINN 训练, 实时计算 FFRLS 的参数辨识误差和 PINN 损失函数变化率, 以调整交替周期, 实现自适应优化. 若  $e(k) > 10^{-3}$  或  $\frac{\partial L_{loss}}{\partial W^{out}} > 10^{-4}$ , 则将交替周期缩短至 3 次 FFRLS 迭代, 提升 PINN 补偿频率, 快速修正线性模型偏差; 若  $e(k) < 10^{-4}$  或  $\frac{\partial L_{loss}}{\partial W^{out}} < 10^{-5}$ , 则将交替周期延长至 8 次 FFRLS 迭代, 避免 PINN 过度训练, 提高整体计算效率.

基于 FFRLS-PINN 的交替辨识算法步骤如下:

step 1: 初始化 FFRLS, 设置初值  $\theta(0) = [0, 0, 0, 0]$ 、 $P(0) = 10I$  以及遗忘因子  $\lambda$ , 累计迭代次数 0, 输入初始数据, 初始化 PINN, 搭建 3 层隐藏层 100 个节点的网络结构, 初始化网络权重与偏置; 初始化交替周期为 5.

step 2: 固定时间步  $k$ , 执行单次 FFRLS 迭代. 计算增益矩阵、参数辨识误差, 更新参数向量、协方差矩阵; 累计 FFRLS 迭代次数+1, 存储当前参数估计值  $\hat{\theta}(k)$ .

step 3: 判断累计迭代此时是否满足交替周期要求, 若满足, 则执行 1500 轮 epoch 训练, 输出未建模动态位置增量估计值  $\Delta \hat{v}[\varphi(k)]$ .

step 4: 根据参数辨识误差和 PINN 损失函数变

化率更新交替周期.

step 5: 融合 FFRLS 的估计值与 PINN 的估计值得到系统整体输出估计值, 并计算系统整体估计误差  $y_{error} = y(k) - \hat{y}(k)$ .

step 6: 若  $e(k) < 10^{-3}$ ,  $\frac{\partial L_{loss}}{\partial W^{out}} < 10^{-5}$  且  $y_{error} < 10^{-5}$ , 迭代终止; 否则令  $k = k + 1$ , 返回 Step2, 继续迭代.

### 4 数值仿真实验

为验证所提算法的有效性, 采用如下非线性系统模型进行仿真实验

$$y(k) = 0.8y(k-1) - 0.3y(k-2) + 1.2u(k) + 0.2u(k-1) + \sin[u(k) + u(k-1) + y(k-1) + y(k-2)] - \frac{u(k) + u(k-1) + y(k-1) + y(k-2)}{1 + u(k)^2 + u(k-1)^2 + y(k-1)^2 + y(k-2)^2} + \xi(k)$$

系统高阶非线性项为

$$v[\varphi(k)] = \sin[u(k) + u(k-1) + y(k-1) + y(k-2)] - \frac{u(k) + u(k-1) + y(k-1) + y(k-2)}{1 + u(k)^2 + u(k-1)^2 + y(k-1)^2 + y(k-2)^2}$$

数据向量为  $\varphi(k) = [y(k), y(k-1), u(k), u(k-1)]^T$ ,  $\xi(k)$  为白噪声.

设置初值为  $\hat{\theta}(0) = [0, 0, 0, 0]^T$ ,  $y(0) = 0, u(0) = 0, v[\varphi(k)] = 0$ , 白噪声  $\xi(k) = 10^{-4}$ .

为分析不同遗忘因子对估计结果的影响, 分别选取  $\lambda = 1.0, 0.995, 0.99, 0.95, 0.9$  进行对比实验, 当  $\lambda = 1.0$  时, 该算法退化为传统 RLS 算法. 以 MAE 值作为指标验证算法有效性, 实验结果如图 2-3 及表 2 所示

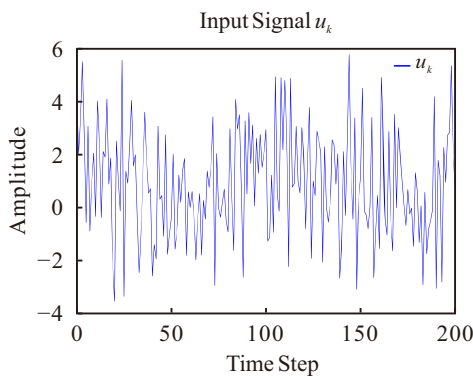


图2 输入信号  $u(k)$

图 2 为输入信号  $u(k)$ , 实验结果表明, 当  $\lambda = 1.000$  时, 算法 MAE 为 0.0445, 旧数据的持续干扰导致算法无法快速适配系统动态特性; 当

$\lambda = 0.995$  时, 整体 MAE 降至 0.0492, 实现了新数据动态跟踪能力与旧数据信息利用效率的最优平衡, 能快速响应系统动态变化; 当  $\lambda = 0.990$  时, 辨识精度较  $\lambda = 0.995$  时略有下降; 当  $\lambda = 0.950$  与  $\lambda = 0.900$  时, 遗忘速度过快, 其 MAE 进一步升高, 辨识误差显著增大. 综上, 选取  $\lambda = 0.995$  为最优遗忘因子进行仿真实验.



图3 系统辨识误差

图 3 为 FFRLS 算法对低阶线性模型的参数估计误差, 误差在 0 附近小幅震荡, 表明辨识结果稳定性较好, 验证了该算法的有效性.

表2 不同  $\lambda$  值 FFRLS 算法性能评价

遗忘因子 $\lambda$	整体MAE	稳定后MAE
1.000	0.0505	0.0445
0.995	0.0492	0.0397
0.990	0.0526	0.0442
0.950	0.0810	0.0582
0.900	0.1083	0.0784

为体现本文算法优势, 分别采用 LSTM、SCNs、PC-BP、PINN 进行仿真对比实验, 实验结果如图 4-5 及表 3 所示

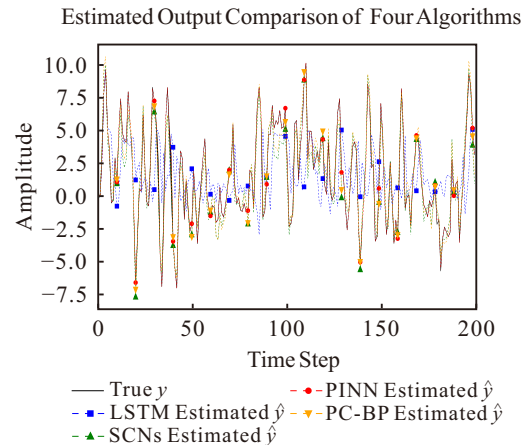


图4 采用四种算法估计结果  $y$

从图 4 可以看出, LSTM 算法虽能够保证多数时刻估计方向的准确性, 但由于未建模动态未知增量

未在训练数据中出现,其泛化能力会急剧下降,对超出训练数据范围的动态难以准确预测,在对未建模动态未知增量进行估计时,能力稍显逊色;SCNs算法的估计值与真实值的变化趋势基本一致,但拟合效果稍显不足;PC-BP算法的估计方向与真实输出高度一致,但由于模型需要对物理约束进行线性化处理,对系统动态变化的精准捕捉能力稍显逊色;PINN对系统整体的估计结果具有较高的精度,其估计值与真实值的拟合程度好。

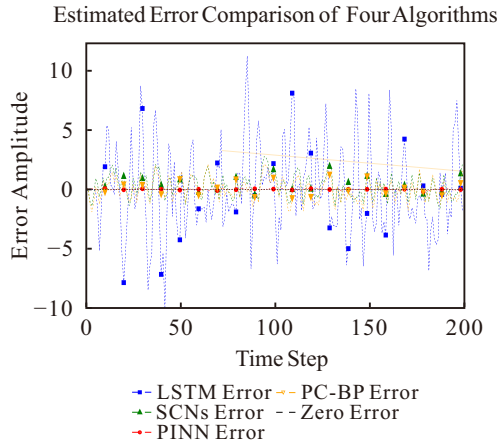


图5 四种算法的估计误差结果对比

图5为四种算法的误差对比图,从图中可以看出PINN的辨识结果误差平稳,无明显震荡;LSTM虽能捕捉部分系统特征,但估计稳定性略显逊色;SCNs的误差则剧烈波动,且拟合精度不是很高;PC-BP的估计误差虽无极端异常值,但波动幅度大于PINN,在估计精度上稍显不足.该结果表明PINN的有效性。

表3 各算法估计 $y(k)$ 的误差对比

算法	RMSE	MAE	$R^2$
PINN	0.0442	0.5079	0.999
LSTM	3.7138	2.9256	-0.071
SCNs	0.9589	0.7734	0.929
PC-BP	0.8931	0.7410	0.938

表3通过三项核心误差指标的量化评估,呈现了四种算法的性能差异,PINN的综合性能最优,与真实值的拟合精度优异且稳定性强,SCNs与PC-BP的表现相近,二者具备一定的建模能力,但仍然存在优化空间。

为进一步说明本文所提方法在小样本场景下的优势,设置了分别为100、200、500、1000的4组训练数据量梯度,测试数据量固定为500样本,各训练数据样本下未建模动态未知增量的MSE与MAE如表4所示

表4 不同数据量下PINN算法性能评价

训练数据量	MSE	MAE
100	0.0038	0.0611
200	0.0030	0.0244
500	0.0028	0.0500
1000	0.0025	0.0613

由表4可知,当训练数据量从100样本提升至200样本时,MSE与MAE出现一定程度的上升,随着训练数据量的增加,拟合精度明显改善,而当数据量达到500样本后,MSE与MAE变化幅度显著减小,表明PINN在小样本场景下已具备有效建模能力,模型已具备充分学习系统非线性动态特性的能力。

## 5 结论

本文提出了一种FFRLS与PINN融合的智能建模方法.对低阶线性模型的未知参数采用FFRLS算法进行估计,通过引入遗忘因子,有效抑制了参数估计过程中的震荡问题;通过PINN对未建模动态未知增量进行估计,从而构建了两融合的智能估计模型.通过仿真对比实验,结果表明了所提算法能够有效提高对非线性系统的建模精度。

## 参考文献 (References)

- [1] 高峰涛. 基于极大似然准则自适应EKF的感应电机转速辨识方法[D]. 西安: 西安理工大学, 2019. (Gao F T. Sensorless control of induction machines based on an adaptive extended kalman filter with maximum likelihood criterion[D]. Xi'an: Xi'an University of Technology, 2019.)
- [2] Ding F. Least squares parameter estimation and multi-innovation least squares methods for linear fitting problems from noisy data[J]. *Journal of Computational and Applied Mathematics*, 2023, 426: 115107.
- [3] Ding Y, Guo C, Luo Y, et al. Research on PID self-tuning control based on recursive least squares parameter identification for the fast steering mirror-based optoelectronic tracking system[J]. *Sensors and Actuators A: Physical*, 2025, 386: 116323.
- [4] Aslam M S. Maximum likelihood least squares identification method for active noise control systems with autoregressive moving average noise[J]. *Automatica*, 2016, 69: 1-11.
- [5] Zhang T, Lu Z R, Liu J K, et al. Parameter identification of nonlinear systems with time-delay from time-domain data[J]. *Nonlinear Dynamics*, 2021, 104(4): 4045-4061.
- [6] 潘雅璞, 谢莉, 杨慧中. 基于KRLS的非均匀采样非线性系统辨识[J]. 控制与决策, 2021, 36(12): 3049-3055. (Pan Y P, Xie L, Yang H Z. Identification of non-uniformly sampled nonlinear systems based on KRLS[J]. *Control and Decision*, 2021, 36(12): 3049-3055.)

- [7] 乔俊飞, 黄卫民, 丁海旭, 等. 复杂工业过程特征建模方法及应用研究[J]. *控制与决策*, 2023, 38(8): 2063-2078.  
(Qiao J F, Huang W M, Ding H X, et al. Research on feature modeling method for complex industrial process and its application[J]. *Control and Decision*, 2023, 38(8): 2063-2078.)
- [8] Narendra K S, Parthasarathy K. Identification and control of dynamical systems using neural networks[J]. *IEEE Transactions on Neural Networks*, 1990, 1(1): 4-27.
- [9] Yu W, Gonzalez J, Li X O. Fast training of deep LSTM networks with guaranteed stability for nonlinear system modeling[J]. *Neurocomputing*, 2021, 422: 85-94.
- [10] Qiao J F, Wang L Y. Nonlinear system modeling and application based on restricted Boltzmann machine and improved BP neural network[J]. *Applied Intelligence*, 2021, 51(1): 37-50.
- [11] 梁相龙, 姚建勇. 基于神经网络的机电伺服系统非线性控制[J]. *控制与决策*, 2023, 38(4): 1008-1014.  
(Liang X L, Yao J Y. Nonlinear control of mechatronic servo system based on neural network[J]. *Control and Decision*, 2023, 38(4): 1008-1014.)
- [12] Wang Y H, Xu Z Q, Fu Y, et al. A novel optimized control strategy based on BP neural network for dynamic heating of sorption energy storage system[J]. *Journal of Energy Storage*, 2026, 141: 119180.
- [13] 张亚军, 柴天佑, 杨杰. 一类非线性离散时间动态系统的交替辨识算法及应用[J]. *自动化学报*, 2017, 43(1): 101-113.  
(Zhang Y J, Chai T Y, Yang J. Alternating identification algorithm and its application for a class of nonlinear discrete-time dynamic systems[J]. *Acta Automatica Sinica*, 2017, 43(1): 101-113.)
- [14] Chai T Y, Zhang J W, Yang T. Demand forecasting of the fused magnesite smelting process with system identification and deep learning[J]. *IEEE Transactions on Industrial Informatics*, 2021, 17(12): 8387-8396.
- [15] 陶金梅, 牛宏, 张亚军, 等. 基于随机配置网络的非线性系统智能建模方法[J]. *控制与决策*, 2022, 37(10): 2559-2564.  
(Tao J M, Niu H, Zhang Y J, et al. An intelligent modeling method for nonlinear systems based on random Configuration networks[J]. *Control and Decision*, 2022, 37(10): 2559-2564.)
- [16] 李景志, 刘志全. 无持续激励条件下船舶在线鲁棒自适应系统辨识[J]. *控制与决策*, 2025, 40(10): 2950-2958.  
(Li J Z, Liu Z Q. Online robust adaptive system identification of ships without persistency of excitation[J]. *Control and Decision*, 2025, 40(10): 2950-2958.)
- [17] Hao X Y, Wang S L, Fan Y C, et al. An improved forgetting factor recursive least square and unscented particle filtering algorithm for accurate lithium-ion battery state of charge estimation[J]. *Journal of Energy Storage*, 2023, 59: 106478.
- [18] Perrusquía A, Yu W. Identification and optimal control of nonlinear systems using recurrent neural networks and reinforcement learning: An overview[J]. *Neurocomputing*, 2021, 438: 145-154.
- [19] 柴天佑, 程思宇, 李平, 等. 端边云协同的复杂工业过程运行控制智能系统[J]. *控制与决策*, 2023, 38(8): 2051-2062.  
(Chai T Y, Cheng S Y, Li P, et al. Intelligent system for operational control of complex industrial process based on end-edge-cloud collaboration[J]. *Control and Decision*, 2023, 38(8): 2051-2062.)
- [20] Zhao S Y, Cheng C M, Lin M M, et al. Physics-informed deep sparse regression network for nonlinear dynamical system identification[J]. *Journal of Sound and Vibration*, 2025, 595: 118796.
- [21] Song N N, Wang C D, Peng H J, et al. A study of mechanism-data hybrid-driven method for multibody system via physics-informed neural network[J]. *Acta Mechanica Sinica*, 2025, 41(3): 524159.
- [22] 王静远. 基于物理信息神经网络的系统辨识与预测控制[D]. 南京: 南京信息工程大学, 2023.  
(Wang J Y. System identification and predictive control based on physics-informed neural networks[D]. Nanjing: Nanjing University of Information Science and Technology, 2023.)
- [23] 张亚军, 魏萃, 柴天佑, 等. 未建模动态增量补偿驱动的非线性 PID 控制及应用[J]. *自动化学报*, 2020, 46(6): 1145-1153.  
(Zhang Y J, Wei C, Chai T Y, et al. Un-modeled Dynamics Increment Compensation Driven Nonlinear PID Control and Its Application[J]. *Acta Automatica Sinica*, 2020, 46(6): 1145-1153.)
- [24] 牛宏, 陶金梅, 张亚军. 一种新的数据驱动的非线性自适应切换控制方法[J]. *自动化学报*, 2020, 46(11): 2359-2366.  
(Niu H, Tao J M, Zhang Y J. A new nonlinear adaptive switching control method based on data driven[J]. *Acta Automatica Sinica*, 2020, 46(11): 2359-2366.)
- [25] Chai T Y, Zhang Y J, Wang H, et al. Data-based virtual unmodeled dynamics driven multivariable nonlinear adaptive switching control[J]. *IEEE Transactions on Neural Networks*, 2011, 22(12): 2154-2172.
- [26] Sheela K G, Deepa S N. Review on methods to fix number of hidden neurons in neural networks[J]. *Mathematical Problems in Engineering*, 2013, 2013: 1-11.

## 作者简介

张艺楠 (1998-), 女, 硕士, 主要研究方向为非线性系统智能建模方法, E-mail: z35758170@163.com;

牛宏 (1979-), 女, 副教授, 主要研究方向为非线性系统稳定性分析, 未建模动态的参数估计, 深度学习等, E-mail: 13942393197@163.com.