12th IFAC International Symposium on
Advanced Control of Chemical Processes
July 14-17, 2024. Toronto, Canada

State estimation of a carbon capture process through POD model reduction and neural network approximation

Siyu Liu [*,**], **Xiao Zhang** [**], **Zhichao Pan** [*], **Xunyuan Yin** [***], **Jinfeng Liu** [****]

[*] *College of Engineering, Zhejiang Normal University, Jinhua 321004, China (siyu.liu@zjnu.edu.cn)*
[**] *School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China (xzhang@jiangnan.edu.cn)*
[***] *School of Chemistry, Chemical Engineering and Biotechnology, Nanyang Technological University, 62 Nanyang Drive, Singapore, 637459 (xunyuan.yin@ntu.edu.sg)*
[****] *Department of Chemical & Materials Engineering, University of Alberta, Edmonton, AB, Canada, T6G 1H9 (jinfeng@ualberta.ca)*

Abstract: This paper presents an efficient approach for state estimation of post-combustion CO_2 capture plants (PCCPs). The approach involves extracting lower-dimensional feature vectors from the high-dimensional operational data of PCCPs and constructing a reduced-order process model through proper orthogonal decomposition (POD). Multi-layer perceptron (MLP) neural network is then constructed and trained to approximate the dynamics of reduced-order process. For state estimation, a reduced-order extended Kalman filtering scheme, grounded in the POD-MLP model, is developed. Our simulations demonstrate that the proposed POD-MLP modeling reduces computational complexity in comparison to the POD-only model when applied to nonlinear systems. Additionally, the proposed algorithm can accurately reconstruct complete state information of PCCPs while markedly improving computational efficiency.

1. INTRODUCTION

In recent years, post-combustion CO_2 capture plants (PCCPs) have gained considerable attention due to their potential in reducing greenhouse gas emissions and mitigating global warming. The carbon capture efficiency and economic cost of PCCPs are highly dependent on the performance of advanced control systems used for managing process operations (Manaf et al., 2019; Liu et al., 2023a).

Real-time information of the key quality variables of the PCCP is essential for the advanced control system to make the most appropriate decisions for safe and efficient process operation. However, measuring all quality variables online through deploying hardware sensors is unrealistic (Liu et al., 2023b). Therefore, it is crucial to exploit state estimation and parameter identification algorithms for PCCPs. Unfortunately, results on state estimation of PCCPs have been limited. Yin et al. (2020) made an initial attempt on estimating the states of the absorber of a PCCP by developing a distributed moving horizon estimation scheme. Wang et al. (2022) presented a robust soft sensor using a neural network and moving horizon estimator to monitor key operating parameters in the carbon capture process. In the context of nonlinear state estimation, there have been some algorithms that have the potential to be leveraged for state estimation of PCCPs (Pan et al., 2022b), e.g., extended Kalman filtering (Liu et al., 2022; Li et al., 2023), moving horizon estimation methods (Pan et al., 2022a), and finite impulse response (FIR) (Pan et al., 2023). Recently, some identification ideas and identification principles have been proposed for determining model parameters, which have the potential for identifying PCCP parameters, such as the multi-innovation identification theory (Xu, 2023; Ding et al., 2024a; Xu et al., 2024b), the hierarchical identification principle (Ding et al., 2024b; Xing et al., 2024; Xu and Ding, 2023; Yang and Ding, 2023), the data filtering methods (Liu et al., 2024; Xu et al., 2024a; Ding et al., 2023) and the separable identification (Xu, 2022). However, most of nonlinear estimation algorithms require accurate first-principles dynamic models of the underlying nonlinear processes.

Data-driven modeling using neural networks has been widely used as an alternative to first-principles modeling for various nonlinear processes. However, one of the major challenges in applying these methods to state estimation of PCCPs is the large computational cost associated with training and testing neural networks on high-dimensional data. To address these limitations associated with state estimation of PCCPs, we propose a solution that combines data-driven modeling using NNs with model reduction techniques. Proper orthogonal decomposition (POD) has been widely adopted in various engineering fields for reducing the dimensionality of high-dimensional data sets

⋆ This work was supported by the National Natural Science Foundation of China (No. 62203187) and the Natural Science Foundation of Jiangsu Province (China, BK20221064).

while preserving dominant patterns or features of the data (Zhang et al., 2023).

Motivated by the observations above, we propose a neural network-based state estimation approach for PCCPs using POD reduced-order models. Specifically, we normalize the data before POD and use the POD approach to obtain a reduced-order model. Then, we train a multi-layer perceptron (MLP) neural network to capture the dominant dynamics of the reduced-order model from the low-dimensional data. We develop a reduced-order extended Kalman filtering algorithm based on the POD-MLP model to estimate the states.

2. MODEL DESCRIPTION

This diagram of the post-combustion capture plant contains the four key components: the absorption and desorption columns, lean-rich heat exchanger (LRHE), and the reboiler. The flue gas, which contains a high concentration of CO_2, is introduced at the bottom of the absorber from the power plant and is then mixed with a lean solvent having low CO_2 levels. 5M Monoethanolamine (MEA) is used as the solvent in this study. The treated flue gas with a reduced amount of CO_2 leaves the absorption column, while the rich solvent with a high concentration of CO_2 is heated via the heat exchanger by exchanging heat with the lean solvent coming from the reboiler. The rich solvent is then fed into the top of the desorption column, where it is heated through contact with the hot vapor from the reboiler. In the desorption column, the CO_2 is stripped from the rich solvent, which is then recycled back to the absorber. The discharged CO_2 gas, with a high concentration of CO_2 (90-99%), is obtained from the desorption column.

The details of the PCCP model are briefly described in the following (Decardi-Nelson et al., 2018):

$$\frac{\partial C_L(i)}{\partial t} = \frac{F_L}{S_c}\frac{\partial C_L(i)}{\partial l} + (N(i)a^I), \tag{1a}$$

$$\frac{\partial C_G(i)}{\partial t} = -\frac{F_G}{S_c}\frac{\partial C_G(i)}{\partial l} - (N(i)a^I), \tag{1b}$$

$$\frac{\partial T_L}{\partial t} = \frac{F_L}{S_c}\frac{\partial T_L}{\partial l} + \frac{(Q_L a^I)}{\sum_{i=1}^{n} C_L(i)C_{p,i}}, \tag{1c}$$

$$\frac{\partial T_G}{\partial t} = -\frac{F_G}{S_c}\frac{\partial T_G}{\partial l} + \frac{(Q_G a^I)}{\sum_{i=1}^{n} C_G(i)C_{p,i}}, \tag{1d}$$

$$\frac{dT_{tu}}{dt} = \frac{\dot{V}_{tu}}{V_{tu}}(T_{tu,in} - T_{tu,out}) + \frac{\dot{Q}}{C_{p_{tu}}\rho_{tu}V_{tu}}, \tag{1e}$$

$$\frac{dT_{sh}}{dt} = \frac{\dot{V}_{sh}}{V_{sh}}(T_{sh,in} - T_{sh,out}) + \frac{\dot{Q}}{C_{p_{sh}}\rho_{sh}V_{sh}}, \tag{1f}$$

$$\rho C_p V\frac{dT_{reb}}{dt} = f_{in}H_{in} - f_V H_{V,out} - f_L H_{L,out} + Q_{reb}. \tag{1g}$$

The dynamic models for the absorption column and desorption column are described by the partial differential equations (1a)–(1d), where $i = CO_2, MEA, H_2O, N_2$, and the subscripts L and G denote liquid and gas phases, respectively. The dependent variables vary with time t and axial position l of the column. It is assumed that each stage in the two columns is well mixed. Their dynamic models are similar except for a few details like the direction of reactions, temperature, and reaction rate constants. The energy balance equations in (1e)–(1f) represent the dynamics of the lean-rich heat exchanger, where $\dot{V}(\mathrm{m}^3/\mathrm{s})$ and $\dot{Q}(\mathrm{kJ/s})$ represent the volumetric flow and heat transfer rate, respectively, the subscripts *tu*, *sh*, *in* and *out* denote the tube-side, shell-side, inlets and outlets of the heat exchanger, respectively. It is assumed that the mass inside the heat exchanger remains constant. Equation (1g) is the energy balance equation of the reboiler, where $T_{reb}(\mathrm{K})$ represents the temperature of the reboiler, the subscripts *in*, *out*, V, and L denote inlet, outlet, vapour and liquid, respectively, and $Q_{reb}(\mathrm{kJ/s})$ is the heat input. The definitions of other variables and parameters of the PCCP can be found in Table 1. Physical property calculations of gas and liquid phases are necessary for the model development, and they are estimated from seven nonlinear algebraic correlations. Details of these calculations are not included in this work but can be found in (Decardi-Nelson et al., 2018).

Table 1. Process variables of each unit of the PCCP.

Notation	Definition	Unit
C_i	Molar concentrations of component i	$\mathrm{mol/m^3}$
S_c	Cross-sectional area of the column	$\mathrm{m^2}$
F	Volumetric flow	$\mathrm{m^3/s}$
N_i	Mass transfer rate	$\mathrm{kmol/m^2s}$
T	Temperature	K
l	Height of the column	m
C_p	Heat capacity	kJ/kmol/K
Q	Heat transfer rate	$\mathrm{kJ/m^2s}$
a^I	Interfacial area	$\mathrm{m^2/m^3}$
V	Volume	$\mathrm{m^3}$
ρ	Average molar density	$\mathrm{kmol/m^3}$
H	Enthalpy	kJ
f	Flow rate	mol/s

The PCCP model consists of partial differential equations for the two columns, and ordinary differential equations for the heat exchanger and reboiler, as well as some algebraic equations for parameter calculations. As the variables in the columns exhibit temporal and spatial distributions, the partial differential equations are discretized using the method outlined in (Decardi-Nelson et al., 2018) to convert them into ordinary differential equations, with the column length divided into five stages. Therefore, the model presented in (1) is expressed as a system of differential-algebraic equations (DAEs):

$$\boldsymbol{x}(k+1) = \boldsymbol{F}(\boldsymbol{x}(k), \boldsymbol{a}(k), \boldsymbol{u}(k)) + \boldsymbol{w}(k), \tag{2a}$$

$$\boldsymbol{G}(\boldsymbol{x}(k), \boldsymbol{a}(k), \boldsymbol{u}(k)) = \boldsymbol{0}, \tag{2b}$$

$$\boldsymbol{y}(k) = \boldsymbol{H}(\boldsymbol{x}(k), \boldsymbol{u}(k)) + \boldsymbol{v}(k), \tag{2c}$$

where $\boldsymbol{x}(k) \in \mathbb{R}^{103}$ is the state vector, and state variables of two columns are discrete at the jth discrete point ($j = 1, 2, \ldots, 5$), $\boldsymbol{a}(k) \in \mathbb{R}^7$ is the algebraic state vector, $\boldsymbol{u}(k) = [F_L, Q_{reb}, F_G] \in \mathbb{R}^3$ denotes the input vector: solvent flow rate in L/s, reboiler heat in kJ/s, and flue gas flow rate $\mathrm{m^3/s}$, and $\boldsymbol{w}(k)$ is the process noise.

3. POD AND ITS APPLICATION TO PCCP

In this section, we employ the proper orthogonal decomposition (POD) to derive a reduced-order model that approximates the dynamics of the PCCP that originally has 103 state variables. This is accomplished by computing the singular value decomposition (SVD) of the data matrix, which yields a set of orthonormal basis vectors that represent the most significant patterns of variability in the data.

For general nonlinear systems described by (2), we obtain a state trajectory by capturing and sampling the system's response to a typical input trajectory at fixed time intervals δ. Then, we sample the resulting state trajectory to construct a matrix of process states from time 0 to N, denoted as:

$$\boldsymbol{\chi} = [\boldsymbol{x}(0)\ \boldsymbol{x}(1)\ \ldots\ \boldsymbol{x}(N)] \in \mathbb{R}^{n\times(N+1)}, \tag{3}$$

where the snapshot matrix $\boldsymbol{\chi}$ is composed of the actual state at each sampling interval, the number of state variables is denoted as n, and the number of sampling intervals is represented by N. To ensure a sufficient number of samples, we require N to be much larger than n.

For PCCP, the magnitudes of different states vary greatly. To ensure that the POD reduction method is not biased towards states with larger magnitudes in PCCP, each state variable x_i (where $i = 1, 2, \ldots, 103$) in the data matrix $\boldsymbol{\chi}$ is normalized using (4) prior to performing SVD decomposition:

$$x_{i,norm} = \frac{x_i - x_{i,min}}{x_{i,max} - x_{i,min}}. \tag{4}$$

This normalization transforms each state variable x_i to $x_{i,norm}$, where $x_{i,min}$ and $x_{i,max}$ are the minimum and maximum values of x_i in the original dataset, respectively. The resulting normalized matrix $\boldsymbol{\chi}_{norm}$ constructed by $x_{i,norm}$ has all states with magnitudes between 0 and 1. SVD is then performed on the normalized matrix $\boldsymbol{\chi}_{norm}$ as follows:

$$\boldsymbol{\chi}_{norm} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}, \tag{5}$$

where $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{V} \in \mathbb{R}^{(N+1)\times(N+1)}$ are orthogonal matrices, and the rectangular matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{n\times(N+1)}$ has non-negative real values on its main diagonal. The diagonal entries σ_i represent the singular values of the $\boldsymbol{\chi}$ matrix, where $i \in 1, 2, \ldots, n$. These values are sorted in descending order on the main diagonal of $\boldsymbol{\Sigma}$.

To construct a reduced-order model, we select a positive integer r that is smaller than the number of states 103, and truncate $\boldsymbol{\Sigma}$ at the rth column and row to form the reduced-order matrix $\boldsymbol{\Sigma}_r \in \mathbb{R}^{r\times r}$ using the first r singular values σ_i. Accordingly, we select the first r columns of $\boldsymbol{U}$ and the first r rows of $\boldsymbol{V}^{\mathrm{T}}$ to form the matrices $\boldsymbol{U}_r$ and $\boldsymbol{V}_r^{\mathrm{T}}$, respectively. Using these matrices, we obtain a reduced-order approximation of normalized process data, given by

$$\boldsymbol{\chi}_{norm} \approx \boldsymbol{U}_r\boldsymbol{\Sigma}_r\boldsymbol{V}_r^{\mathrm{T}}. \tag{6}$$

We define $\boldsymbol{\xi} \in \mathbb{R}^r$ as the state vector of the reduced-order model, and set $\boldsymbol{\xi}(k) := \boldsymbol{U}_r^{\mathrm{T}}\boldsymbol{x}(k)$. Using the truncated SVD matrices, the original nonlinear model in (2) can be expressed as a reduced-order model in state-space form:

$$\boldsymbol{\xi}(k+1) = \boldsymbol{U}_r^{\mathrm{T}}\boldsymbol{F}(\boldsymbol{U}_r\boldsymbol{\xi}(k), \boldsymbol{a}(k), \boldsymbol{u}(k)) + \boldsymbol{U}_r^{\mathrm{T}}\boldsymbol{w}(k), \tag{7a}$$

$$\boldsymbol{G}(\boldsymbol{U}_r\boldsymbol{\xi}(k), \boldsymbol{a}(k), \boldsymbol{u}(k)) = \boldsymbol{0}, \tag{7b}$$

$$\boldsymbol{y}(k) = \boldsymbol{H}(\boldsymbol{U}_r\boldsymbol{\xi}(k), \boldsymbol{u}(k)) + \boldsymbol{v}(k). \tag{7c}$$

The evolution of $\boldsymbol{\xi}(k)$ in the reduced-order model can be used to approximate the actual state trajectory of the original nonlinear process through the mapping $\boldsymbol{x}(k) \approx \boldsymbol{U}_r\boldsymbol{\xi}(k)$.

Remark 1. In this section, we improve the accuracy of the model approximation by normalizing the data prior to applying the POD method to obtain a reduced-order model. The effectiveness of the normalization for POD will also be illustrated through simulations in Section 6.

4. APPROXIMATING REDUCED-ORDER MODEL WITH MLP NETWORKS

The POD technique is commonly applied in linear systems to decrease computational costs by reducing the dimensionality of the problem. However, it does not yield similar advantages for nonlinear systems due to the challenge in explicitly expressing $\boldsymbol{U}_r^{\mathrm{T}}\boldsymbol{F}(\boldsymbol{U}_r\boldsymbol{\xi}, \boldsymbol{a}, \boldsymbol{u})$ in terms of the reduced basis $\boldsymbol{U}_r$. Consequently, evaluating the reduced-order model may require more time than evaluating the original nonlinear function $\boldsymbol{F}$. Different approaches have been adopted to address this issue. For instance, a linear parameter varying model was utilized to approximate the reduced-order model, however, the benefits were found to be insignificant (Zhang and Liu, 2019).

To address this issue, we present a method to speed up the evolution of reduced-order models for nonlinear systems such as (2). The method employs a multi-layer perceptron (MLP) neural network to fit the reduced-order model and hence decrease the computation time. The MLP neural network model of $\boldsymbol{\xi}$ is given by:

$$\boldsymbol{\xi}(k+1) = \boldsymbol{f}_{mlp}(\boldsymbol{\xi}(k), \boldsymbol{u}(k)) + \boldsymbol{w}_r(k), \tag{8}$$

where the vector function $\boldsymbol{f}_{mlp} \in \mathbb{R}^r$ approximates the dynamic behavior of $\boldsymbol{\xi}$ in (7a), and $\boldsymbol{w}_r \in \mathbb{R}^r$ denotes the process noise and model error. Consequently, we do not have to evaluate the vector function $\boldsymbol{F} : \mathbb{R}^{103} \to \mathbb{R}^{103}$ of the full-order model, leading to significant time savings.

MLP models are a type of artificial neural network widely used to approximate complex and nonlinear problems. An MLP typically consist of an input layer, one or more hidden layers, and an output layer. Each layer in an MLP contains multiple fully connected neurons, which are connected to the next layer by weights. In supervised learning, the weights are adjusted to approximate each target value. The number of neurons in the input and output layers is determined by the input and output variables, respectively. The computation time required for MLP output is relatively short, as only a few matrix multiplications, vector additions, and function evaluations are necessary. Given these characteristics, MLP is a suitable choice for approximating the vector function $\boldsymbol{f}_{mlp}$ in equation (2) after POD model reduction. The basic model formulation of MLP is indicated below:

$$\boldsymbol{z}^{(l)} = \sigma_h(\boldsymbol{w}^{(l)}\boldsymbol{z}^{(l-1)} + \boldsymbol{b}^{(l)}), \quad \boldsymbol{y} = \boldsymbol{w}^o\boldsymbol{z}^{(l)} + \boldsymbol{b}^o \tag{9}$$

where $\boldsymbol{z}^{(l)}$ denotes the output vector of the l-th hidden layer, obtained by applying an activation function σ_h to the weighted sum of the input vector $\boldsymbol{z}^{(l-1)}$ and the bias vector $\boldsymbol{b}^{(l)}$. The weight matrix of the l-th hidden layer is

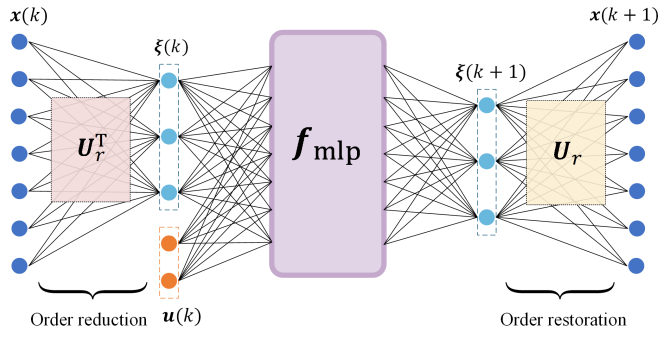

Fig. 1. The structure of the POD-MLP model.

denoted by $\boldsymbol{w}^{(l)}$. The input vector $\boldsymbol{z}^{(0)}$ is the MLP input, and the output vector $\boldsymbol{y}$ is obtained by applying the weight matrix $\boldsymbol{w}^o$ to the output vector of the last hidden layer and adding the bias vector $\boldsymbol{b}^o$. It is worth noting that the MLP output layer is typically linear, while the activation function of the hidden layer can be chosen from various options based on the specific problem being solved.

The reduced-order PCCP model is enhanced by utilizing an MLP network, where the input is $\boldsymbol{\xi}_u := [\boldsymbol{u}^{\mathrm{T}}, \boldsymbol{\xi}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{3+r}$, and the output is $\hat{\boldsymbol{\xi}} \in \mathbb{R}^r$. The input layer has $3+r$ neurons, and the output layer has r neurons. The MLP model is trained to minimize the mean-squared-error (MSE) between the predicted output $\hat{\boldsymbol{\xi}}$ and the actual output $\boldsymbol{\xi}$:

$$L = MSE(\boldsymbol{\xi}, \hat{\boldsymbol{\xi}}). \tag{10}$$

The model structure of the proposed POD-based MLP (POD-MLP) model is shown in Figure 1.

Remark 2. The POD-MLP model not only reduces the order of the system, but also eliminates the need for solving the DAEs in the original PCCP model. This is achieved by extracting the dynamic information of the algebraic state variable $\boldsymbol{a}$ from the data, and excluding $\boldsymbol{a}$ from the model.

5. EKF USING THE POD-MLP MODEL

In this section, we develop an extended Kalman filtering (EKF) based on the POD-MLP model (the POD-MLP-EKF algorithm for short) to estimate the actual process states. The POD-MLP model is summarized in the following:

$$\boldsymbol{\xi}(k+1) = \boldsymbol{f}_{mlp}(\boldsymbol{\xi}(k), \boldsymbol{u}(k)) + \boldsymbol{w}_r(k), \tag{11a}$$
$$\boldsymbol{y}(k) = \boldsymbol{H}(\boldsymbol{U}_r\boldsymbol{\xi}(k), \boldsymbol{u}(k)) + \boldsymbol{v}(k). \tag{11b}$$

Assuming that $\boldsymbol{w}_r(k)$ and $\boldsymbol{v}(k)$ are two mutually uncorrelated Gaussian noise sequences with zero-mean, we further assume that they have covariance matrices $\boldsymbol{Q}_r$ and $\boldsymbol{R}_r$.

Based on the above preparation, the POD-MLP-EKF algorithm is designed in the following two steps:
Step 1: Prediction step:

$$\hat{\boldsymbol{\xi}}(k+1|k) = \boldsymbol{f}_{mlp}(\hat{\boldsymbol{\xi}}(k|k), \boldsymbol{u}(k)), \tag{12a}$$
$$\boldsymbol{P}(k+1|k) = \boldsymbol{A}(k)\boldsymbol{P}(k|k)\boldsymbol{A}^{\mathrm{T}}(k) + \boldsymbol{Q}_r, \tag{12b}$$

where $\hat{\boldsymbol{\xi}}(k+1|k)$ is the prediction of the system state at time $k+1$ based on the current state estimate $\hat{\boldsymbol{\xi}}(k|k)$ and the input $\boldsymbol{u}(k)$, and $\boldsymbol{P}(k+1|k)$ contains a priori error covariance information, incorporating the prediction error and the uncertainty associated with the system dynamics through the covariance matrix $\boldsymbol{Q}_r$. The matrix $\boldsymbol{A}(k) = \frac{\partial \boldsymbol{f}_{mlp}(\boldsymbol{\xi},\boldsymbol{u})}{\partial \boldsymbol{\xi}}|_{\boldsymbol{\xi}=\hat{\boldsymbol{\xi}}(k|k)}$ is the Jacobian matrix of the MLP model with respect to the state vector $\boldsymbol{\xi}$ evaluated at the predicted state estimate $\hat{\boldsymbol{\xi}}(k|k)$.
Step 2: Update step:

$$\boldsymbol{K}(k+1) = \frac{\boldsymbol{P}(k+1|k)\boldsymbol{C}^{\mathrm{T}}(k)}{\boldsymbol{C}(k)\boldsymbol{P}(k+1|k)\boldsymbol{C}^{\mathrm{T}}(k) + \boldsymbol{R}_r}, \tag{13a}$$
$$\hat{\boldsymbol{\xi}}(k+1|k+1) = \hat{\boldsymbol{\xi}}(k+1|k) + \boldsymbol{K}(k+1)[\boldsymbol{y}(k+1) - \boldsymbol{H}(\boldsymbol{U}_r\hat{\boldsymbol{\xi}}(k+1|k), \boldsymbol{u}(k))], \tag{13b}$$
$$\boldsymbol{P}(k+1|k+1) = [\boldsymbol{I} - \boldsymbol{K}(k+1)\boldsymbol{C}(k)]\boldsymbol{P}(k+1|k), \tag{13c}$$

where the correction gain $\boldsymbol{K}(k+1)$ is computed based on the a priori estimation error covariance $\boldsymbol{P}(k+1|k)$, the measurement error covariance $\boldsymbol{R}_r$, and the observation matrix $\boldsymbol{C}(k)$, which maps the predicted state $\hat{\boldsymbol{\xi}}(k+1|k)$ to the measurement space. The state estimate is then updated to $\hat{\boldsymbol{\xi}}(k+1|k+1)$ using the correction gain and the measurement innovation $\boldsymbol{y}(k+1) - \boldsymbol{H}(\boldsymbol{U}_r\hat{\boldsymbol{\xi}}(k+1|k), \boldsymbol{u}(k))$, which represents the difference between the actual measurement and the predicted measurement based on the predicted state. Finally, the a posteriori estimation error covariance matrix $\boldsymbol{P}(k+1|k+1)$ is computed based on the updated state estimate and the correction gain, which reflects the reduced uncertainty in the estimated state after incorporating the measurement information.

Then, we can obtain the state estimates of the actual PCCP states, denoted by $\hat{\boldsymbol{x}}$, by utilizing the reduced-order state estimate $\hat{\boldsymbol{\xi}}(k+1|k+1)$ and the linear mapping $\boldsymbol{U}_r$, as follows:

$$\hat{\boldsymbol{x}}(k+1) = \boldsymbol{U}_r\hat{\boldsymbol{\xi}}(k+1|k+1). \tag{14}$$

6. SIMULATION RESULTS

To perform the model reduction, the input vector $\boldsymbol{u}(k) = [F_L, Q_{reb}, F_G]$ is used to excite the PCCP, which are constrained as 0.48 L/s $\leq F_L \leq$ 0.66 L/s, 0.14 kJ/s $\leq Q_{reb} \leq$ 0.20 kJ/s and 0.8 m^3/s $\leq F_G \leq$ 1.2 m^3/s. The dynamic model of the PCCP is discretized at a sample interval of Δ = 30s. Pseudo-random multi-level signal is used as excitation signal. The system states are sampled over a duration of 100 hours to construct the snapshot matrix $\boldsymbol{\chi}$ for POD reduction. With 12,000 sampling intervals, the requirement $N \gg n$ is satisfied.

Next, we apply the SVD to the matrix $\boldsymbol{\chi}$ which has dimensions 103 by 12000. This results in a unitary matrix U that can be used for coordinate transformation. To evaluate the reduced-order model accuracy, we use the root-mean-square error (RMSE) to denote the model errors, which is defined as follows:

$$\text{RMSE} := \sqrt{\frac{\sum_{j=0}^{N}\sum_{i=1}^{103}(x_{i,norm}(j) - \hat{x}_{i,norm}(j))^2}{N}},$$

where $\hat{x}_{i,norm} := \boldsymbol{U}_r\xi_i$ is the ith approximated state obtained from a reduced order model. To validate the accuracy of a reduced-order model, we should use input trajectories that are different from the ones used in POD model reduction. Additional 600 sampling data are used for validation. Based on the actual state trajectories and reduced-order model state trajectories, the RMSE is calculated for each model. The values of log(RMSE) at different

orders $r = 20, \ldots, 90$ under the POD with normalization and the POD without normalization are shown in Figure 2. It shows that the degree of model mismatch increases with the decrease in the model order for both cases. Moreover, the RMSE value of POD with normalization is smaller than that without normalization. This is because for the POD method without normalization of the data matrix $\boldsymbol{\chi}$, the approximation of the state with small numerical values is not accurate when the model is reduced, and the model approximation error is added to each state in the form of absolute value. While the method that normalizes the data matrix $\boldsymbol{\chi}$ reflects the model approximation error in the form of relative values for each state. Therefore, RMSE values of reduced-order models at different orders obtained from POD with normalization are smaller than those obtained from POD. This can also be further demonstrated from Fig. 3, which shows the trajectories of some of states based on the original model and the reduced model with order 30 using POD with normalization and POD methods respectively. From the figure, it can be seen that for states with large numerical values (x_{17}, x_{101}), both methods provide very good approximation effects, but for states with small numerical values or small fluctuations (x_3, x_{11}, x_{26}, x_{31}), the approximation of POD with normalization is significantly better than that of POD.

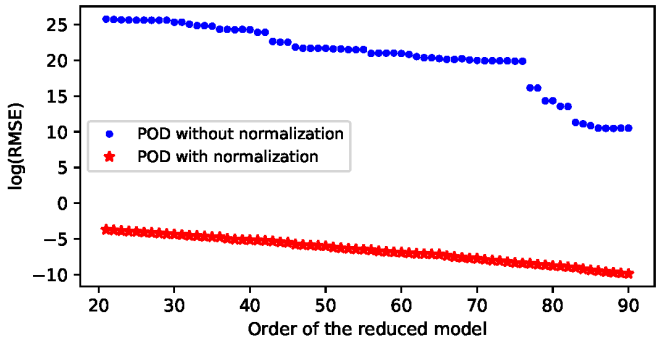

Fig. 2. The values of log(RMSE) at different orders.

Next, we will evaluate the open-loop prediction performance of the POD-MLP model. Using the PRMS input signal, we generate 100,000 samples for each state by utilizing the first-principle model with $\boldsymbol{U}_r$. These data are split into training (70%), validation (20%), and testing (10%) datasets. The developed POD-MLP model consists of 3 hidden layers with 128 neurons, the input of the POD-MLP is the normalized 33-dimensional vector $\boldsymbol{\xi}_u$ and the output is the normalized state vector $\boldsymbol{\xi}$. The activation function for the hidden layers has been selected to be Tanh, and a linear activation function is used in the output layer. Figure 4 demonstrates the testing performance of the POD-MLP model in multi-step open-loop prediction for the actual state trajectory of the PCCP. It can be seen that the fit is accurate.

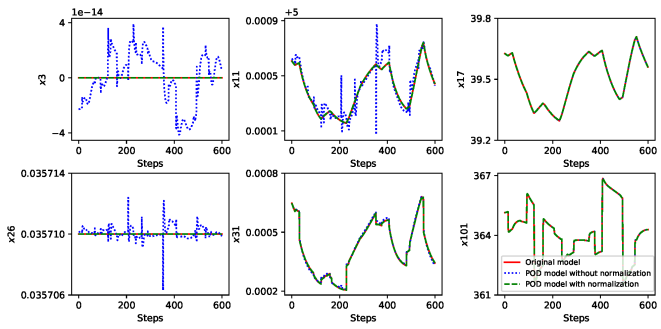

Fig. 3. The trajectories of some states based on the original model and the reduced model using POD with normalization and POD without normalization.

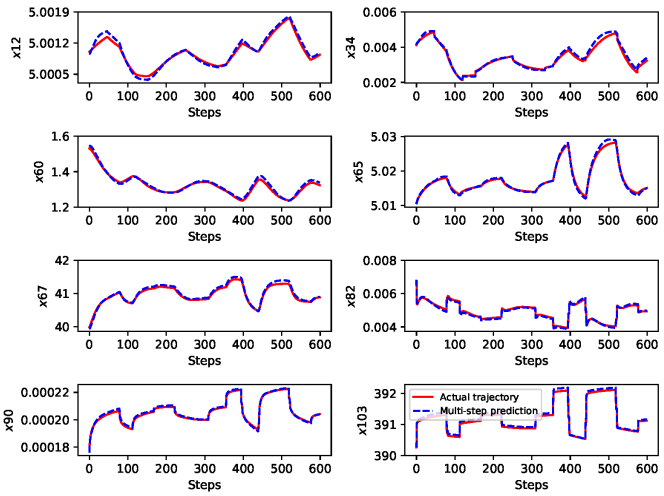

Fig. 4. Multi-step prediction for the PCCP.

In the following analysis, we focus on the state estimation of the PCCP using the POD-MLP-EKF algorithm. The PCCP is subject to process disturbances, and the output measurements are corrupted by random noise. Specifically, each process disturbance sequence associated with the ith state x_i is generated following a normal distribution with zero mean and a standard deviation $0.01x_{i,s}$, where $x_{i,s}$ is the steady-state value of x_i. Random noise is added to each measurement y_i as Gaussian white noise with zero mean and a standard deviation $0.01y_{i,s}$, where $y_{i,s}$ is the value of y_i at steady-state. As a result, the covariance matrices of process noise and measurement noise are $\boldsymbol{Q} = \text{diag}((0.01\boldsymbol{x}_s))^2$ and $\boldsymbol{R} = \text{diag}((0.01\boldsymbol{y}_s))^2$, respectively. The tuning parameters in the POD-MLP-EKF are $\boldsymbol{Q}_r = \boldsymbol{U}_r^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{U}_r$ and $\boldsymbol{R}_r = \boldsymbol{U}_r^{\mathrm{T}}\boldsymbol{R}\boldsymbol{U}_r$. The initial guess for the normalized states is set to $0.5 \times \boldsymbol{1}_{103}$. Figure 5 shows some of the state estimates and the actual states. The proposed estimation scheme provides accurate state estimates.

We compare the computational efficiency of the proposed POD-MLP-EKF algorithm, a centralized EKF design directly based on the PCCP model, and an EKF for POD model, denoted as the POD-EKF. Specifically, we evaluate the average computation time for 600 steps required by the three algorithms, which is 125 s, 127 s, and 6.4 s, respectively. The results demonstrate that the combination of POD reduction and neural network significantly reduces the computing time.

7. CONCLUSIONS

The proposed POD-MLP model can facilitate reductions in the computational cost for model training and implementation of state estimation. The POD-MLP-EKF algorithm provides a promising solution to state estimation of

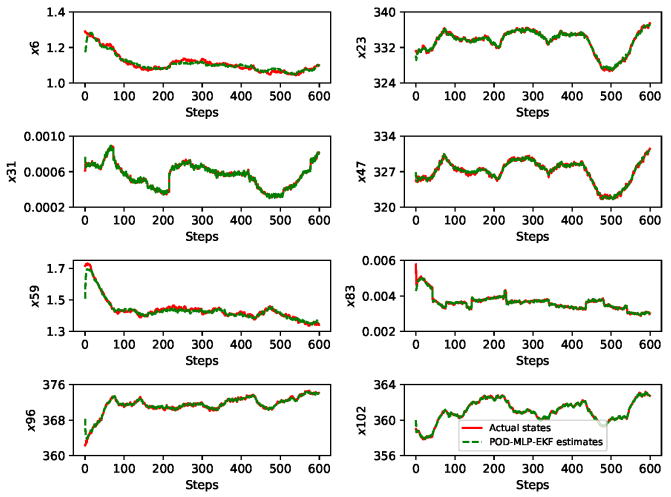

Fig. 5. The actual states and POD-MLP-EKF state estimates for the PCCP.

PCCPs and can potentially be applied to other complex systems with high dimension for efficient state estimation.

REFERENCES

Decardi-Nelson, B., Liu, S., and Liu, J. (2018). Improving flexibility and energy efficiency of post-combustion CO_2 capture plants using economic model predictive control. *Processes*, 6(9), 135.

Ding, F., Xu, L., Zhang, X., and Ma, H. (2024a). Hierarchical gradient-and least-squares-based iterative estimation algorithms for input-nonlinear output-error systems from measurement information by using the over-parameterization. *International Journal of Robust and Nonlinear Control*, 34(2), 1120–1147.

Ding, F., Xu, L., Zhang, X., and Zhou, Y. (2023). Filtered auxiliary model recursive generalized extended parameter estimation methods for box–jenkins systems by means of the filtering identification idea. *International Journal of Robust and Nonlinear Control*, 33(10), 5510–5535.

Ding, F., Xu, L., Zhang, X., Zhou, Y., and Luan, X. (2024b). Recursive identification methods for general stochastic systems with colored noises by using the hierarchical identification principle and the filtering identification idea. *Annual Reviews in Control*, 57, 100942.

Li, X., Law, A.W.K., and Yin, X. (2023). Partition-based distributed extended kalman filter for large-scale nonlinear processes with application to chemical and wastewater treatment processes. *AIChE Journal*, 69(12), e18229.

Liu, S., Wang, Y., Ding, F., Alsaedi, A., and Hayat, T. (2024). Joint iterative state and parameter estimation for bilinear systems with autoregressive noises via the data filtering. *ISA Transactions*. doi: 10.1016/j.isatra.2024.01.035.

Liu, S., Yin, X., and Liu, J. (2023a). Sensor network design for post-combustion CO_2 capture plants: Computational efficiency and robustness. *Journal of Process Control*, 129, 103035.

Liu, S., Yin, X., Pan, Z., and Liu, J. (2023b). A sensitivity-based approach to optimal sensor selection for complex processes. *Chemical Engineering Science*, 278, 118901.

Liu, S., Zhang, X., Xu, L., and Ding, F. (2022). Expectation–maximization algorithm for bilinear systems by using the Rauch–Tung–Striebel smoother. *Automatica*, 142, 110365.

Manaf, N.A., Qadir, A., and Abbas, A. (2019). Efficient energy management of CO_2 capture plant using control-based optimization approach under plant and market uncertainties. *Journal of Process Control*, 74, 2–12.

Pan, Z., Huang, B., and Liu, F. (2022a). Moving horizon estimation for bounded noises based on the set-membership approach. *Journal of Process Control*, 119, 25–33.

Pan, Z., Luan, X., and Liu, F. (2022b). Confidence set-membership state estimation for LPV systems with inexact scheduling variables. *ISA transactions*, 122, 38–48.

Pan, Z., Zhao, S., Huang, B., and Liu, F. (2023). Confidence set-membership FIR filter for discrete time-variant systems. *Automatica*, 157, 111231.

Wang, Q., Zheng, C., Wu, X., and Wang, M. (2022). Robust monitoring of solvent based carbon capture process using deep learning network based moving horizon estimation. *Fuel*, 321, 124071.

Xing, H., Ding, F., Zhang, X., Luan, X., and Yang, E. (2024). Highly-efficient filtered hierarchical identification algorithms for multiple-input multiple-output systems with colored noises. *Systems & Control Letters*, 186, 105762.

Xu, L. (2022). Separable newton recursive estimation method through system responses based on dynamically discrete measurements with increasing data length. *International Journal of Control, Automation and Systems*, 20(2), 432–443.

Xu, L. (2023). Parameter estimation for nonlinear functions related to system responses. *International Journal of Control, Automation and Systems*, 21(6), 1780–1792.

Xu, L. and Ding, F. (2023). Separable synthesis gradient estimation methods and convergence analysis for multivariable systems. *Journal of Computational and Applied Mathematics*, 427, 115104.

Xu, L., Ding, F., Zhang, X., and Zhu, Q. (2024a). Novel parameter estimation method for the systems with colored noises by using the filtering identification idea. *Systems & Control Letters*, 186, 105774.

Xu, L., Xu, H., and Ding, F. (2024b). Adaptive multi-innovation gradient identification algorithms for a controlled autoregressive autoregressive moving average model. *Circuits, Systems, and Signal Processing*, 43. doi:10.1007/s00034-024-02627-z.

Yang, D. and Ding, F. (2023). Multi-innovation gradient-based iterative identification methods for feedback nonlinear systems by using the decomposition technique. *International Journal of Robust and Nonlinear Control*, 33(13), 7755–7773.

Yin, X., Decardi-Nelson, B., and Liu, J. (2020). Distributed monitoring of the absorption column of a post-combustion CO_2 capture plant. *International Journal of Adaptive Control and Signal Processing*, 34(6), 757–776.

Zhang, A. and Liu, J. (2019). Economic mpc of wastewater treatment plants based on model reduction. *Processes*, 7(10), 682.

Zhang, X., Han, M., and Yin, X. (2023). Reduced-order koopman modeling and predictive control of nonlinear processes. *Computers & Chemical Engineering*, 179, 108440.