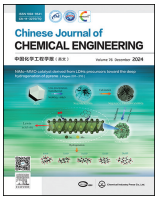
Chinese Journal of Chemical Engineering 76 (2024) 251–263

Contents lists available at ScienceDirect

Chinese Journal of Chemical Engineering

journal homepage: www.elsevier.com/locate/CJChE



Full Length Article

A fuzzy compensation-Koopman model predictive control design for pressure regulation in proten exchange membrane electrolyzer

Haokun Xiong , Lei Xie *, Cheng Hu , Hongye Su

State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China

ARTICLE INFO

Article history:
Received 7 November 2023
Received in revised form
4 September 2024
Accepted 23 September 2024
Available online 30 September 2024

Keywords:
Hydrogen production
PEM electrolyzer
Nonlinear control
Model predictive control
Koopman operator
Fuzzy logic system

ABSTRACT

Proton exchange membrane (PEM) electrolyzer have attracted increasing attention from the industrial and researchers in recent years due to its excellent hydrogen production performance. Developing accurate models to predict their performance is crucial for promoting and accelerating the design and optimization of electrolysis systems. This work developed a Koopman model predictive control (MPC) method incorporating fuzzy compensation for regulating the anode and cathode pressures in a PEM electrolyzer. A PEM electrolyzer is then built to study pressure control and provide experimental data for the identification of the Koopman linear predictor. The identified linear predictors are used to design the Koopman MPC. In addition, the developed fuzzy compensator can effectively solve the Koopman MPC model mismatch problem. The effectiveness of the proposed method is verified through the hydrogen production process in PEM simulation.

1. Introduction

Hydrogen plays an important role in developing an environmentally friendly energy system in the future of the world. Compared to non-renewable fossil fuels, hydrogen is used as a source of energy or fuel and may be the ideal energy carrier [1] due to its high energy content by mass without emitting pollutants or greenhouse gases. Traditionally, hydrogen can be produced from various sources including oil, natural gas, and coal. However, these methods of hydrogen production are highly polluting. In today's global low-carbon development, hydrogen production from water electrolysis is the most promising way. Hydrogen production by water electrolysis can use electrical or thermal energy generated from nuclear or renewable sources (*i.e.* wind and solar) with no carbon emissions during the process. There are three main types of water electrolysis hydrogen production technologies, depending on the type of electrolyzer: Alkaline water electrolysis (AWE), proton exchange membrane (PEM), and solid oxide electrolysis cells (SOEC) [2].

The poor quality of the hydrogen produced and the need for additional treatment are the drawbacks of AWE hydrogen production. The expensive electrode and electrolyte materials as well as the prolonged start-up and shutdown times are SOEC's drawbacks.

Although the PEM technique has many advantages such as compact system design, higher hydrogen purity, faster transient response, higher current density, and higher voltage efficiency. However, the main disadvantages of PEM electrolyzer are the higher cost of components and lower durability compared to the alkaline electrolyzer [3], which severely limits the use of PEM electrolysis. Moreover, the high-pressure operation of the electrolyzer facilitates the delivery of hydrogen to the end user and reduces the energy consumed by further compression and storage of hydrogen. However, according to Fick's law of diffusion, gas permeability increases significantly with increasing pressure [4]. Correa *et al.* [5], Han *et al.* [6], and Koponen *et al.* [3] investigated the effect of the operating pressure on the electrolyzer performance from aspects including energy consumption, activation overpotential, efficiency, *etc.* The drawn conclusion shows that hydrogen production could be maximized by controlling the stack hydrogen outlet pressure as close to the storage pressure as possible, while the specific energy consumption of the PEM water electrolyzer increased at higher hydrogen outlet pressure. In summary, PEM water electrolysis needs to be carried out at appropriate operating pressures in many practical situations.

A suitable dynamic model is required to control the PEM electrolyzer system to achieve optimal performance. Regarding PEM electrolyzer modeling state of the art, the number of publications is considerably lower when compared to fuel cell modeling. In 2006,

* Corresponding author.
E-mail address: leix@iipc.zju.edu.cn (L. Xie).

https://doi.org/10.1016/j.cjche.2024.09.004

the first dynamic PEM electrolysis model was introduced by Görgün [7]. The model was developed in Simulink and considered the PEM electrolyzer to be composed of five ancillaries: anode, cathode, membrane, voltage, and hydrogen storage. Abdin *et al.* [8] presented an enhanced and detailed model and tested it against published experimental results. A large number of operating parameters and empirical constants involved in PEM electrolyzer modeling can be found in these works mentioned above, in addition to the values for various properties in Refs. [9–13]. Only a few of the research focuses on applying the control strategy on PEM electrolyzer to optimize the efficiency, *e.g.*, Tabanjat *et al.* [14] designed a fuzzy inference system coupled with PI controller, which can control water temperature and maximize hydrogen production of PEM electrolyzer. Fortunately, a large number of available publications on fuel cell control provide useful knowledge about the control design of electrolyzers due to certain structural similarities between electrolyzers and fuel cells. Pukrushpan *et al.* [15] presented a dynamic fuel cell system model suitable for control studies and gave the analysis of system observability. Via Pukrushpan's model, Kunusch *et al.* [16] and Matraji *et al.* [17] both achieved second-order sliding mode controllers for the pressure regulation problem taking into account several load conditions, uncertainties, and disturbances, which inspired modeling and control design in our study. Majumdar *et al.* [18] review control-oriented PEM electrolyzer models and discussed different controllers used in literature. Flamm *et al.* [19] derived nonlinear and piecewise-affine models of the PEM electrolyzer from experimental data, and presented the results of online experiments for the real-time model predictive control (MPC) of an electrolyzer.

The main source of the difficulty in pressure control of PEM electrolyzer is high nonlinearity and conflicting control objectives. The approaches we mention above, such as PID and sliding mode control, have disadvantages in dealing with constraints. Unlike the traditional method, MPC is an advanced technique that is used in diverse industrial areas to control multi-input multi-output (MIMO) systems subject to constraints. To overcome the computational complexity of nonlinear MPC [20], industrial approaches for MPC problems are often based on linear approximations around equilibria, making them only effective locally and leading to open-loop unstable during transitions. To address this, two conventional approaches are available for solving tracking problems across a wide range of operating conditions: gain-scheduled MPC (GSMPC) and linear parameter-varying MPC (LPV MPC) [21]. A straightforward method for obtaining an LPV model is the online successive linearization at the current states based on a first-principles model [22], and this technique will serve as a point of reference in our study. However, these approaches require prior physical knowledge of the nonlinear system, which is difficult to determine. Alternatively, we might use data-driven strategies like conventional machine learning and deep learning to build a plant model in forms other than complex mathematical equations. Although these methods can uncover the "black-box" mapping from inputs to outputs, the derived models are not amenable to linear MPC as they are nonlinear and may not be easily invertible. Fortunately, the Koopman operator (an infinite-dimensional operator) theory offers a data-driven approach that yields globally linear and explicit control-oriented models. In Ref. [23], the author utilized a vector of scalar functions (observables) to lift the nonlinear dynamics into a higher but finite dimensional space (finite subspace of observables) where its evolution is approximately linear. By using input-output data, a process called extended dynamic mode decomposition (EDMD) [24] is employed to compute the approximation of the Koopman operator, which can approximately describe the evolution of observables along trajectories of the nonlinear dynamical system. If we assume the outputs of the plant lie within the finite subspace of observables, the nonlinear dynamical system has linear state-space representation (a linear predictor) in the lifted space. Thus, a linear MPC can be designed for this state-space model. Much research on Koopman-based MPC currently focuses on power systems [25], chemical engineering [26], and so on.

Developing accurate analytical models for electrolysis systems and reliable controllers based on such models is extremely difficult. The system exhibits unmodeled nonlinearity even when linear predictors are applied based on Koopman operators. Thus, applying fuzzy control strategies for such a system seems appropriate since intelligent controllers with fuzzy logic systems (FLSs) utilize human experience more naturally than the conventional mathematical approach [27]. Recently, tremendous advanced adaptive fuzzy control schemes have been developed for nonlinear systems, as in Refs. [28,29]. Tabanjat *et al.* [14] used a 2D FLC to determine the appropriate water temperature set-point for the PI controller. FLC was also used by Cano *et al.* [30] to generate the power input reference for both an electrolyzer and a fuel cell in a hybrid renewable energy system.

In this work, a fuzzy compensation-Koopman MPC (FC-KMPC) method is developed to regulate the anode and cathode pressure of PEM electrolyzer. Before implementing the proposed controller, we develop a PEM electrolyzer state-space model which serves as a virtual process to generate data for constructing Koopman linear predictor. Then, the EDMD method is used to identify the linear predictor and the accuracy of the identification model is validated. Then a fuzzy logic system is built to estimate and compensate for the current plant-model mismatch. Finally, the results of the proposed controller are compared with the nominal Koopman MPC, neural network model predictive control (NNMPC) [31] and successive linearization MPC (SLMPC). The proposed FC-KMPC can effectively compensate for the mismatch and achieve zero steady-state offset.

2. Preliminaries

Consider a discrete time-invariant nonlinear model defined as follows:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_k) \\ \boldsymbol{y}_k &= \boldsymbol{h}(\boldsymbol{x}_k) \end{aligned} \tag{1}$$

where $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ is the system state vector, $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$ is the system output or observation vector, and $\boldsymbol{f}_{\mathrm{d}} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$ and $\boldsymbol{h} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_y}$ are vector-valued nonlinear functions.

The Koopman operator is an operator, $\mathcal{K} : \mathcal{F} \to \mathcal{F}$, that acts on *every* scalar-valued functions (often referred to as observables) $g : \mathbb{R}^{n_x} \to \mathbb{R}$ belonging to the space of observables $\mathcal{F}$ as follows:

$$\mathcal{K}g(\boldsymbol{x}_k) = g(\boldsymbol{x}_{k+1}) = g(\boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_k)) \tag{2}$$

When $\mathcal{F}$ is a vector space, $\mathcal{K}$ is a linear operator.

The eigenvalues, λ_i, and eigenfunctions, ϕ_i, of $\mathcal{K}$ are defined as follows:

$$\mathcal{K}\phi_i(\boldsymbol{x}_k) = \lambda_i \phi_i(\boldsymbol{x}_k), i = 1, 2, \cdots \tag{3}$$

Koopman eigenfunctions define a nonlinear change of coordinates in which the system becomes linear.

Now, consider a vector-valued observable $\boldsymbol{g} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_g}$, *i.e.*, $g_j \in \mathcal{F}$, $j = 1, \cdots, n_g$. If all the elements of observable $\boldsymbol{g}$ and system output $\boldsymbol{h}$ lie within $\mathrm{span}\{\phi_i\}_{i=1}^{n_\phi}$ where n_ϕ could be finite or infinite, the set of linearly independent eigenfunctions $\{\phi_i\}_{i=1}^{n_\phi}$ is a subset of a function basis for $\mathcal{F}$, then $\boldsymbol{g}$ and $\boldsymbol{h}$ can be expanded in terms of these eigenfunctions:

$$\begin{aligned} \boldsymbol{g}(\boldsymbol{x}_k) &= \sum_{i=1}^{n_\phi} \phi_i(\boldsymbol{x}_k)\phi_i^g = \boldsymbol{\Phi}^g\boldsymbol{\phi}(\boldsymbol{x}_k), \\ \boldsymbol{g}(\boldsymbol{x}_{k+1}) &= \sum_{i=1}^{n_\phi} \lambda_i\phi_i(\boldsymbol{x}_k)\phi_i^g = \boldsymbol{\Phi}^g\boldsymbol{\Lambda}\boldsymbol{\phi}(\boldsymbol{x}_k) \end{aligned} \tag{4}$$

$$\begin{aligned} \boldsymbol{h}(\boldsymbol{x}_k) &= \sum_{i=1}^{n_\phi} \phi_i(\boldsymbol{x}_k)\phi_i^h = \boldsymbol{\Phi}^h\boldsymbol{\phi}(\boldsymbol{x}_k), \\ \boldsymbol{h}(\boldsymbol{x}_{k+1}) &= \sum_{i=1}^{n_\phi} \lambda_i\phi_i(\boldsymbol{x}_k)\phi_i^h = \boldsymbol{\Phi}^h\boldsymbol{\Lambda}\boldsymbol{\phi}(\boldsymbol{x}_k) \end{aligned} \tag{5}$$

where

$$\boldsymbol{\Phi}^g = \boldsymbol{\Phi}^g(\boldsymbol{g}) = \begin{pmatrix} \phi_1^g & \phi_2^g & \cdots & \phi_{n_\phi}^g \end{pmatrix} \in \mathbb{C}^{n_g\times n_\phi}$$

$$\boldsymbol{\Phi}^h = \boldsymbol{\Phi}^h(\boldsymbol{h}) = \begin{pmatrix} \phi_1^h & \phi_2^h & \cdots & \phi_{n_\phi}^h \end{pmatrix} \in \mathbb{C}^{n_y\times n_\phi}$$

$$\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_{n_\phi}) \in \mathbb{C}^{n_\phi\times n_\phi}$$

$$\boldsymbol{\phi}(\boldsymbol{x}_k) = \begin{pmatrix} \phi_1(\boldsymbol{x}_k)^{\mathrm{T}} & \phi_2(\boldsymbol{x}_k)^{\mathrm{T}} & \cdots & \phi_{n_\phi}(\boldsymbol{x}_k)^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} \in \mathbb{C}^{n_\phi},$$

where $\phi_i^g \in \mathbb{C}^{n_g}$ are the Koopman modes belonging to $\boldsymbol{g}$, and $\phi_i^h \in \mathbb{C}^{n_y}$ are the output Koopman modes belonging to $\boldsymbol{h}$, and the triplets $(\lambda_i, \phi_i, \phi_i^g)$, $i = 1,2,\cdots,n_\phi$, are referred as Koopman tuples [32]. This expansion is referred to as Koopman mode decomposition [33]. If matrix $\boldsymbol{\Phi}^g$ is invertible ($n_\phi = n_g$), then we have the action of $\mathcal{K}$ on the particular observable $\boldsymbol{g}$, not the full action of the Koopman operator on the entire observable space, $\mathcal{F}$.

$$\begin{aligned} &\begin{pmatrix} (\mathcal{K}g_1)(\boldsymbol{x}_k) \\ \vdots \\ \left(\mathcal{K}g_{n_g}\right)(\boldsymbol{x}_k) \end{pmatrix} = \boldsymbol{g}(\boldsymbol{x}_{k+1}) = \boldsymbol{A}\boldsymbol{g}(\boldsymbol{x}_k) \\ &\boldsymbol{h}(\boldsymbol{x}_k) = \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x}_k) \end{aligned} \tag{6}$$

where $\boldsymbol{A} = \boldsymbol{\Phi}^g\boldsymbol{\Lambda}(\boldsymbol{\Phi}^g)^{-1}$ and $\boldsymbol{C} = \boldsymbol{\Phi}^h(\boldsymbol{\Phi}^g)^{-1}$ are constant matrices independent of $\boldsymbol{x}_k$, since $\boldsymbol{\Phi}^g$ is the function of only the chosen $\boldsymbol{g}$ and $\boldsymbol{\Phi}^h$ is the function of only $\boldsymbol{h}$.

Following Kutz *et al.* [34], suppose we have a collection of data $(\boldsymbol{x}_{(k)}, \boldsymbol{x}'_{(k)})$, $k = 1,\cdots,L$ satisfying $\boldsymbol{x}'_{(k)} = \boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_{(k)})$, the EDMD computes the best-fit linear operator $\boldsymbol{A} \in \mathbb{R}^{n_g\times n_g}$ that approximates (the transpose of) the infinite-dimensional Koopman operator $\mathcal{K}$:

$$\boldsymbol{A} = \arg\min_{\hat{\boldsymbol{A}}} \sum_{k=1}^{L} \left\| \boldsymbol{g}\left(\boldsymbol{x}'_{(k)}\right) - \hat{\boldsymbol{A}}\boldsymbol{g}\left(\boldsymbol{x}_{(k)}\right) \right\|_2^2 \tag{7}$$

$$\boldsymbol{C} = \arg\min_{\hat{\boldsymbol{C}}} \sum_{k=1}^{L} \left\| \boldsymbol{h}\left(\boldsymbol{x}_{(k)}\right) - \hat{\boldsymbol{C}}\boldsymbol{g}\left(\boldsymbol{x}_{(k)}\right) \right\|_2^2 \tag{8}$$

3. Koopman Operator and EDMD for Controlled Systems

Korda *et al.* [23] present a practical way of generalizing the Koopman operator to a controlled system. Consider a discrete-time nonlinear controlled dynamical system as follows:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_k, \boldsymbol{u}_k) \\ \boldsymbol{y}_k &= \boldsymbol{h}(\boldsymbol{x}_k) \end{aligned} \tag{9}$$

where $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$, $\boldsymbol{u}_k \in \mathcal{U} \subset \mathbb{R}^{n_u}$, $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$. Then an uncontrolled dynamical system evolving on the extended state-space is defined by

$$\boldsymbol{\chi}_{k+1} = \boldsymbol{F}_{\mathrm{d}}(\boldsymbol{\chi}_k) := \left(\boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_k, \boldsymbol{u}_k), \{\boldsymbol{u}_i\}_{i=k+1}^{\infty}\right) \tag{10}$$

where the infinite-dimensional object $\boldsymbol{\chi}_k = (\boldsymbol{x}_k, \{\boldsymbol{u}_i\}_{i=k}^{\infty}) \in \mathbb{R}^{n_x}\times \ell(\mathcal{U})$ is the extended state, $\ell(\mathcal{U})$ is the space of all sequences $\{\boldsymbol{u}_i\}_{i=k}^{\infty}$. The Koopman operator $\mathcal{K} : \mathcal{F} \to \mathcal{F}$ associated to Eq. (10) is defined by

$$\mathcal{K}g(\boldsymbol{\chi}_k) = g(\boldsymbol{F}_{\mathrm{d}}(\boldsymbol{\chi}_k)) \tag{11}$$

for each $g : \mathbb{R}^{n_x}\times\ell(\mathcal{U}) \to \mathbb{R}$ belonging to the space of observables $\mathcal{F}$. The EDMD algorithm assumes that a collection of data $(\boldsymbol{\chi}_{(k)}, \boldsymbol{\chi}'_{(k)})$, $k = 1,\cdots,L$ satisfying $\boldsymbol{\chi}'_{(k)} = \boldsymbol{F}_{\mathrm{d}}(\boldsymbol{\chi}_{(k)})$ is available and seeks a matrix

$$\boldsymbol{A} = \arg\min_{\hat{\boldsymbol{A}}} \sum_{k=1}^{L} \left\| \boldsymbol{g}\left(\boldsymbol{\chi}'_{(k)}\right) - \hat{\boldsymbol{A}}\boldsymbol{g}\left(\boldsymbol{\chi}_{(k)}\right) \right\|_2^2 \tag{12}$$

where $\boldsymbol{g}$ is a vector of observables $g_j : \mathbb{R}^{n_x}\times\ell(\mathcal{U}) \to \mathbb{R}$, $j = 1,\cdots,n_g$. In order to evaluate the $\boldsymbol{g}(\boldsymbol{\chi}_{(k)})$ in a finite time, g_j's are chosen in a special way as follows:

$$\boldsymbol{g}(\boldsymbol{\chi}_k) = \begin{pmatrix} \boldsymbol{\Psi}(\boldsymbol{x}_k) \\ \boldsymbol{u}_k \end{pmatrix} \tag{13}$$

where the vector of lifting functions $\boldsymbol{\Psi} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_\psi}$, $n_\psi + n_u = n_g$, is a nonlinear mapping. Since there is no need to predict future values of the control sequence, the last n_u components of $\boldsymbol{g}\left(\boldsymbol{\chi}'_{(k)}\right) - \hat{\boldsymbol{A}}\boldsymbol{g}\left(\boldsymbol{\chi}_{(k)}\right)$ in Eq. (12) can be removed. Denoting $\boldsymbol{M}$ the first n_ψ rows of $\boldsymbol{A} \in \mathbb{R}^{n_g\times n_g}$ and decomposing this matrix such that $\boldsymbol{M} = (\boldsymbol{A} \quad \boldsymbol{B})$ with $\boldsymbol{A} \in \mathbb{R}^{n_\psi\times n_\psi}$, $\boldsymbol{B} \in \mathbb{R}^{n_\psi\times n_u}$, leads to the problem

$$\begin{aligned} \boldsymbol{M} &= \arg\min_{\hat{\boldsymbol{M}}} \sum_{k=1}^{L} \left\| \boldsymbol{\Psi}\left(\boldsymbol{x}'_{(k)}\right) - \widehat{\boldsymbol{M}}\boldsymbol{g}\left(\boldsymbol{\chi}_{(k)}\right) \right\|_2^2 \\ &= \arg\min_{\hat{\boldsymbol{M}}} \sum_{k=1}^{L} \left\| \boldsymbol{\Psi}\left(\boldsymbol{x}'_{(k)}\right) - \hat{\boldsymbol{A}}\boldsymbol{\Psi}\left(\boldsymbol{x}_{(k)}\right) - \hat{\boldsymbol{B}}\boldsymbol{u}_{(k)} \right\|_2^2 \end{aligned} \tag{14}$$

In a similar way we can obtain the matrix $\boldsymbol{C}$ by solving the problem

$$\boldsymbol{C} = \arg\min_{\hat{\boldsymbol{C}}} \sum_{k=1}^{L} \left\| \boldsymbol{h}\left(\boldsymbol{x}_{(k)}\right) - \hat{\boldsymbol{C}}\boldsymbol{\Psi}\left(\boldsymbol{x}_{(k)}\right) \right\|_2^2 \tag{15}$$

This leads to the predictor for the controlled system (9)

$$\begin{aligned} \boldsymbol{z}_{k+1} &= \boldsymbol{A}\boldsymbol{z}_k + \boldsymbol{B}\boldsymbol{u}_k \\ \hat{\boldsymbol{y}}_k &= \boldsymbol{C}\boldsymbol{z}_k \end{aligned} \tag{16}$$

where $\boldsymbol{z} \in \mathbb{R}^{n_\psi}$, $\hat{\boldsymbol{y}}_k$ is the prediction of output $\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k)$. The initial condition of the predictor is $\boldsymbol{z}_0 = \boldsymbol{\Psi}(\boldsymbol{x}_0)$.

The approach can be extended to controlled systems including disturbances of the form

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k) \\ \boldsymbol{y}_k &= \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k) \end{aligned} \tag{17}$$

where $\boldsymbol{w}_k \in \mathbb{R}^{n_w}$ is the disturbances. Then define the extended state and the extended system as follows:

$$\boldsymbol{\chi}_k = (\boldsymbol{x}_k, \{\boldsymbol{u}_i\}_{i=k}^{\infty}, \{\boldsymbol{w}_i\}_{i=k}^{\infty}) \tag{18}$$

$$\begin{aligned}&\boldsymbol{\chi}_{k+1}=\boldsymbol{F}_{\mathrm{d}}(\boldsymbol{\chi}_k)\\&:=\left(\boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_k,\boldsymbol{u}_k,\boldsymbol{w}_k),\{\boldsymbol{u}_i\}_{i=k+1}^{\infty},\{\boldsymbol{w}_i\}_{i=k+1}^{\infty}\right)\end{aligned}\tag{19}$$

The vector of observables is

$$\boldsymbol{g}(\boldsymbol{\chi}_k)=\begin{pmatrix}\boldsymbol{\Psi}(\boldsymbol{x}_k)^{\mathrm{T}} & \boldsymbol{u}_k^{\mathrm{T}} & \boldsymbol{w}_k^{\mathrm{T}}\end{pmatrix}^{\mathrm{T}}\tag{20}$$

The definition of Koopman operator is the same as Eq. (11). Assume that a collection of data $(\boldsymbol{x}_{(k)},\boldsymbol{x}'_{(k)},\boldsymbol{u}_{(k)},\boldsymbol{w}_{(k)})$, $k=1,\cdots,L$ satisfying $\boldsymbol{x}'_{(k)}=\boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_{(k)},\boldsymbol{u}_{(k)},\boldsymbol{w}_{(k)})$, $\boldsymbol{y}_{(k)}=\boldsymbol{h}(\boldsymbol{x}_{(k)},\boldsymbol{u}_{(k)},\boldsymbol{w}_{(k)})$ is given. In a similar way, matrices $\boldsymbol{M}=(\boldsymbol{A}\quad\boldsymbol{B}\quad\boldsymbol{D})$ and $\boldsymbol{M}_y=(\boldsymbol{C}\quad\boldsymbol{F}\quad\boldsymbol{G})$ are obtained by solving least-squares problems:

$$\min_{\hat{M}}\sum_{k=1}^{L}\left\|\boldsymbol{\Psi}\left(\boldsymbol{x}'_{(k)}\right)-\widehat{\boldsymbol{A}}\boldsymbol{\Psi}\left(\boldsymbol{x}_{(k)}\right)-\widehat{\boldsymbol{B}}\boldsymbol{u}_{(k)}-\widehat{\boldsymbol{D}}\boldsymbol{w}_{(k)}\right\|_2^2\tag{21}$$

$$\min_{\hat{M}_y}\sum_{k=1}^{L}\left\|\boldsymbol{y}_{(k)}-\widehat{\boldsymbol{C}}\boldsymbol{\Psi}\left(\boldsymbol{x}_{(k)}\right)-\widehat{\boldsymbol{F}}\boldsymbol{u}_{(k)}-\widehat{\boldsymbol{G}}\boldsymbol{w}_{(k)}\right\|_2^2\tag{22}$$

Then the goal of constructing a linear predictor for system (17) is achieved:

$$\begin{aligned}\boldsymbol{z}_{k+1}&=\boldsymbol{A}\boldsymbol{z}_k+\boldsymbol{B}\boldsymbol{u}_k+\boldsymbol{D}\boldsymbol{w}_k\\\widehat{\boldsymbol{y}}_k&=\boldsymbol{C}\boldsymbol{z}_k+\boldsymbol{F}\boldsymbol{u}_k+\boldsymbol{G}\boldsymbol{w}_k\end{aligned}\tag{23}$$

with the same nitial condition $\boldsymbol{z}_0=\boldsymbol{\Psi}(\boldsymbol{x}_0)$.

Note that the Koopman operators with different definitions associated to systems Eq. (10) and Eq. (18) respectively all can fully describe the dynamical systems provided that the space of observables $\mathcal{F}$ contains the components of the state x_i, *i.e.*, the mappings $\boldsymbol{x}\to x_i$ belong to $\mathcal{F}$, $i=1,\cdots,n_x$. Hence, the lifting functions ψ_i also contain the state itself. An additional benefit of setting $\psi_i(\boldsymbol{x})=x_i$, $i=1,\cdots,n_x$, is the convenience when we impose linear constraints on the state for the Koopman MPC.

4. Control Framework

4.1. Koopman MPC design

Using the linear predictor, Koopman MPC solves at each time instant k of the closed-loop operation the optimization problem

$$\begin{aligned}\min_{\boldsymbol{u}_{k+i|k},\boldsymbol{y}_{k+i|k}}\quad&J\left(\left\{\boldsymbol{u}_{k+i|k}\right\}_{i=0}^{N},\left\{\boldsymbol{y}_{k+i|k}\right\}_{i=0}^{N}\right)\\\text{s.t.}\quad&\boldsymbol{z}_{k+i+1|k}=\boldsymbol{A}\boldsymbol{z}_{k+i|k}+\boldsymbol{B}\boldsymbol{u}_{k+i|k}+\boldsymbol{D}\boldsymbol{w}_{k+i|k},\\&i=0,\cdots,N-1\\&\boldsymbol{y}_{k+i|k}=\boldsymbol{C}\boldsymbol{z}_{k+i|k}+\boldsymbol{F}\boldsymbol{u}_{k+i|k}+\boldsymbol{G}\boldsymbol{w}_{k+i|k},\\&i=0,\cdots,N\\&\boldsymbol{E}_i\boldsymbol{z}_{k+i|k}+\boldsymbol{L}_i\boldsymbol{u}_{k+i|k}+\boldsymbol{M}_i\boldsymbol{y}_{k+i|k}\le\boldsymbol{b}_i,\\&i=0,\cdots,N\\&\boldsymbol{z}_{k|k}=\boldsymbol{z}_k,\boldsymbol{w}_{k|k}=\boldsymbol{w}_k\end{aligned}\tag{24}$$

where N is the prediction horizon, $\boldsymbol{z}_{k+i|k}$, $\boldsymbol{y}_{k+i|k}$ and $\boldsymbol{u}_{k+i|k}$ denote lifted state, output, and input vectors at time step $k+i$ that are predicted at time step k (*i.e.* predictions of their values i steps ahead).

Stack the predicted sequences into vectors $\boldsymbol{U}_k\in\mathbb{R}^{n_u(N+1)}$, $\boldsymbol{Z}_k\in\mathbb{R}^{n_z(N+1)}$, $\boldsymbol{Y}_k\in\mathbb{R}^{n_y(N+1)}$, $\boldsymbol{W}_k\in\mathbb{R}^{n_w(N+1)}$.

$$\begin{aligned}\boldsymbol{U}_k&=\begin{pmatrix}\boldsymbol{u}_{k|k}^{\mathrm{T}} & \boldsymbol{u}_{k+1|k}^{\mathrm{T}} & \cdots & \boldsymbol{u}_{k+N|k}^{\mathrm{T}}\end{pmatrix}^{\mathrm{T}},\\\boldsymbol{Z}_k&=\begin{pmatrix}\boldsymbol{z}_{k|k}^{\mathrm{T}} & \boldsymbol{z}_{k+1|k}^{\mathrm{T}} & \cdots & \boldsymbol{z}_{k+N|k}^{\mathrm{T}}\end{pmatrix}^{\mathrm{T}},\\\boldsymbol{Y}_k&=\begin{pmatrix}\boldsymbol{y}_{k|k}^{\mathrm{T}} & \boldsymbol{y}_{k+1|k}^{\mathrm{T}} & \cdots & \boldsymbol{y}_{k+N|k}^{\mathrm{T}}\end{pmatrix}^{\mathrm{T}},\\\boldsymbol{W}_k&=\begin{pmatrix}\boldsymbol{w}_{k|k}^{\mathrm{T}} & \boldsymbol{w}_{k+1|k}^{\mathrm{T}} & \cdots & \boldsymbol{w}_{k+N|k}^{\mathrm{T}}\end{pmatrix}^{\mathrm{T}}\end{aligned}$$

Notice that measurements of the future disturbance are not available, we assume that $\boldsymbol{w}_{k|k}=\boldsymbol{w}_{k+1|k}=\cdots=\boldsymbol{w}_{k+N|k}$.

For compact notation, the original MPC optimization problem is equivalent to

$$\begin{aligned}\min_{\boldsymbol{u}_{k+i|k},\boldsymbol{y}_{k+i|k}}\quad&J\left(\left\{\boldsymbol{u}_{k+i|k}\right\}_{i=0}^{N},\left\{\boldsymbol{y}_{k+i|k}\right\}_{i=0}^{N}\right)\\\text{s.t.}\quad&\boldsymbol{Z}_k=\boldsymbol{\Psi}_A\boldsymbol{z}_{k|k}+\boldsymbol{\Psi}_B\boldsymbol{U}_k+\boldsymbol{\Psi}_D\boldsymbol{W}_k\\&\boldsymbol{Y}_k=\overline{\boldsymbol{C}}\boldsymbol{Z}_k+\overline{\boldsymbol{F}}\boldsymbol{U}_k+\overline{\boldsymbol{G}}\boldsymbol{W}_k\\&\boldsymbol{E}\boldsymbol{Z}_k+\boldsymbol{L}\boldsymbol{U}_k\le\boldsymbol{b}\\&\boldsymbol{z}_{k|k}=\boldsymbol{z}_k,\boldsymbol{w}_{k|k}=\boldsymbol{w}_k\end{aligned}\tag{25}$$

If the control objective is to track a given reference $\boldsymbol{y}^{\mathrm{r}}_{k+i|k}$, then the convex quadratic cost function is given by

$$\begin{aligned}&J\left(\left\{\boldsymbol{u}_{k+i|k}\right\}_{i=0}^{N},\left\{\boldsymbol{y}_{k+i|k}\right\}_{i=0}^{N}\right)\\&=\sum_{i=0}^{N}\left\|\boldsymbol{y}_{k+i|k}-\boldsymbol{y}^{\mathrm{r}}_{k+i|k}\right\|_{\boldsymbol{\Lambda}_i}^2+\sum_{i=0}^{N}\left\|\boldsymbol{u}_{k+i|k}\right\|_{\boldsymbol{R}_i}^2\\&=\left\|\boldsymbol{Y}_k-\boldsymbol{Y}^{\mathrm{r}}_k\right\|_{\boldsymbol{\Lambda}}^2+\left\|\boldsymbol{U}_k\right\|_{\boldsymbol{R}}^2\end{aligned}\tag{26}$$

where

$$\begin{aligned}&\boldsymbol{\Lambda}=\mathrm{diag}(\boldsymbol{\Lambda}_0,\cdots,\boldsymbol{\Lambda}_N),\boldsymbol{R}=\mathrm{diag}(\boldsymbol{R}_0,\cdots,\boldsymbol{R}_N),\\&\boldsymbol{Y}^{\mathrm{r}}_k=\begin{pmatrix}\boldsymbol{y}^{\mathrm{r}}_{k|k}{}^{\mathrm{T}} & \boldsymbol{y}^{\mathrm{r}}_{k+1|k}{}^{\mathrm{T}} & \cdots & \boldsymbol{y}^{\mathrm{r}}_{k+N|k}{}^{\mathrm{T}}\end{pmatrix}^{\mathrm{T}}\end{aligned}$$

So, the "sparse-form" MPC problem can be transformed in the so-called "dense-form" MPC

$$\begin{aligned}\min_{\boldsymbol{U}_k}\quad&\left\|\boldsymbol{U}_k\right\|_{\boldsymbol{H}}^2+\boldsymbol{h}^{\mathrm{T}}\boldsymbol{U}_k\\\text{s.t.}\quad&(\boldsymbol{E}\boldsymbol{\Psi}_B+\boldsymbol{L})\boldsymbol{U}_k\le\boldsymbol{b}-\boldsymbol{E}\left(\boldsymbol{\Psi}_A\boldsymbol{z}_{k|k}+\boldsymbol{\Psi}_D\boldsymbol{W}_k\right)\\&\boldsymbol{z}_{k|k}=\boldsymbol{z}_k,\boldsymbol{w}_{k|k}=\boldsymbol{w}_k\end{aligned}\tag{27}$$

where $\boldsymbol{H}$ is positive (semi-)definite, and since the constraints are linear, this is a convex optimization problem that has a unique solution $\boldsymbol{U}_k^*$. Then only the first n_u elements of the $\boldsymbol{U}_k^*$ are inputs to the system. The explicit expressions for the matrices of problems (25) and (27) are given in Appendix A.

4.2. Compensation strategy based on fuzzy logic

To overcome the plant-model mismatch due to the nonlinearity of the process, the finite-dimensional approximation to the Koopman operator, *etc*. We designed a fuzzy logic system (FLS) to compensate for the influence of the mismatch on the controlled variables $\boldsymbol{y}_k$. The principle is to observe the tracking error of the actual output and the steady-state situation to correct the linear predictor.

Assume that the tracking error $\boldsymbol{e}_k=\boldsymbol{y}_k-\boldsymbol{y}^{\mathrm{r}}_k$ is measured. If $\boldsymbol{y}_k$ and $\boldsymbol{y}^{\mathrm{r}}_k$ converge to constants in steady state, *i.e.* $\boldsymbol{y}_k\to\boldsymbol{y}_\infty$, $\boldsymbol{y}^{\mathrm{r}}_k\to\boldsymbol{y}^{\mathrm{r}}_\infty$ as $k\to\infty$, we require a compensator to determine whether the state of the process at current time k is more like a steady or transient state,

and the compensator is also expected to compute a proper parameter $\boldsymbol{d}_k$ and modify the output formula of the linear predictor Eq. (23) to the following format:

$$\widehat{\boldsymbol{y}}_k = \boldsymbol{C}\boldsymbol{z}_k + \boldsymbol{F}\boldsymbol{u}_k + \boldsymbol{G}\boldsymbol{w}_k + \boldsymbol{d}_k \tag{28}$$

or rewrite Eq. (26) as

$$\sum_{i=0}^{N}\left\|\boldsymbol{y}_{k+i|k} + \boldsymbol{d}_k - \boldsymbol{y}^{\mathrm{r}}_{k+i|k}\right\|^2_{\Lambda_i} + \sum_{i=0}^{N}\left\|\boldsymbol{u}_{k+i|k}\right\|^2_{\boldsymbol{R}_i} \tag{29}$$

Obviously, if the process achieves a steady state, the parameter shall satisfy $\boldsymbol{d}_k = -\boldsymbol{e}_\infty = \boldsymbol{y}^{\mathrm{r}}_\infty - \boldsymbol{y}_\infty$, otherwise the parameter shall satisfy $\boldsymbol{d}_k = \boldsymbol{0}$. Considering the possibility of frequent changes in the operating conditions, it is necessary to provide an update method for the parameter as follows:

$$\boldsymbol{d}_k = s_k\boldsymbol{d}_{k-1} - \alpha_k\boldsymbol{e}_k \tag{30}$$

where $s_k \in \mathcal{S} = [0, 1]$, $\alpha_k \in \mathcal{A} = [0, 0.1]$, $s_k \approx 1$ and stepsize $\alpha_k > 0$ means being steady, $s_k = \alpha_k = 0$ means being transient. In this study, s_k, α_k are chosen as the outputs of the FLS.

Given that the FLS is required to determine the current state by analyzing the magnitude of the input value, we choose the norm of derivates such as

$$n_k = \left\| \begin{pmatrix} \sqrt{\boldsymbol{R}_0}\,\dfrac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} \\ \sqrt{\Lambda_0}\,\dfrac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t} \\ \sqrt{\Lambda_0}\,\dfrac{\mathrm{d}\boldsymbol{e}}{\mathrm{d}t} \end{pmatrix} \right\|_2 \in \mathcal{N} = [0, 1] \tag{31}$$

to be the input of the FLS. When n_k exceeds tolerance, the process is considered to be transient.

Additionally, if the FLS is in compensation mode, the tolerance shall be greater to avoid oscillation due to model modification. Thus the mode m_k (1 for compensation and 0 for idle) is introduced:

$$m_k = \begin{cases} 1, & s_{k-1} > 0.5 \\ 0, & s_{k-1} \le 0.5 \end{cases} \in \mathcal{M} = [0, 1] \tag{32}$$

Hence the FLS has inputs n_k, m_k and outputs s_k, α_k. The membership functions for these variables are given in 1.

In Fig. 1(a), a point $n_k \in \mathcal{N}$ has three membership values: $\mu_{\mathrm{AZ}}(n_k)$, $\mu_{\mathrm{PS}}(n_k)$ and $\mu_{\mathrm{PM}}(n_k)$. $\mu_{\mathrm{AZ}}(\cdot)$, $\mu_{\mathrm{PS}}(\cdot)$, and $\mu_{\mathrm{PM}}(\cdot)$ represent the membership functions for the norm of derivatives being "approximately zero", "positive small", and "positive medium", respectively. In order to briefly summarize the fuzzy logic rule base, simplified statements are used, for instance, "n_k is $\mathcal{N}_{\mathrm{AZ}}$" means "$n_k = 0$ with a certain grade $\mu_{\mathrm{AZ}}(n_k)$" or "n_k belongs to $\mathcal{N}$ with a membership value $\mu_{\mathrm{AZ}}(n_k)$" *etc.* Also, similar notation for m_k, s_k and α_k is employed. Then the fuzzy logic rule base for mismatch compensating application is written as

$R^{(1)}$: IF n_k is $\mathcal{N}_{\mathrm{AZ}}$
THEN s_k is $\mathcal{S}_{\mathrm{steady}}$, α_k is $\mathcal{A}_{\mathrm{PM}}(0.2)$;

$R^{(2)}$: IF n_k is $\mathcal{N}_{\mathrm{PS}}$ AND m_k is $\mathcal{M}_{\mathrm{on}}$
THEN s_k is $\mathcal{S}_{\mathrm{steady}}$, α_k is $\mathcal{A}_{\mathrm{PS}}$ (0.2);

$R^{(3)}$: IF n_k is $\mathcal{N}_{\mathrm{PS}}$ AND m_k is $\mathcal{M}_{\mathrm{off}}$
THEN s_k is $\mathcal{S}_{\mathrm{transient}}$, α_k is $\mathcal{A}_{\mathrm{AZ}}$ (1)

$R^{(4)}$: IF n_k is $\mathcal{N}_{\mathrm{PM}}$
THEN s_k is $\mathcal{S}_{\mathrm{transient}}$, α_k is $\mathcal{A}_{\mathrm{AZ}}$ (1) (33)

Note that the rule weights are applied to the rules. The weight must be a number between 0 and 1, and is generally left as 1. Then, in the defuzzification module, the weighted average formula [35] can be used to obtain the realistic outputs s_k and α_k.

The structure of the system with the FC-KMPC is presented in Fig. 2.

5. Model for PEM Electrolyzer

In this section we present the dynamic model of PEM electrolyzer with a state-space presentation, five ancillary dynamics of PEM electrolyzer, and finally the control objective.

5.1. Dynamic model

In this study, the state space equations of the electrolyzer can be expressed as follows.

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}) \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}) \end{aligned} \tag{34}$$

where

$$\boldsymbol{x} = \begin{pmatrix} N_{\mathrm{O_2}} \\ N^{\mathrm{an}}_{\mathrm{H_2O}} \\ N_{\mathrm{H_2}} \\ N^{\mathrm{cat}}_{\mathrm{H_2O}} \\ P_{\mathrm{b}} \end{pmatrix}, \boldsymbol{u} = \begin{pmatrix} \dot{N}^{\mathrm{an,in}}_{\mathrm{H_2O}} \\ P_{\mathrm{an,out}} \\ P_{\mathrm{cat,out}} \\ I_{\mathrm{el}} \end{pmatrix}, \boldsymbol{y} = \begin{pmatrix} P_{\mathrm{an}} \\ P_{\mathrm{cat}} \\ V_{\mathrm{el}} \\ \dot{N}^{\mathrm{out}}_{\mathrm{H_2}} \end{pmatrix}, \boldsymbol{w} = T_{\mathrm{el}}$$

For the sake of convenience and simplicity, it is assumed that the measure of the outputs $\boldsymbol{y}$, the states $\boldsymbol{x}$, the control inputs $\boldsymbol{u}$ and disturbances $\boldsymbol{w}$ are available at current time instant. To obtain the discrete-time dynamics $\boldsymbol{x}_{k+1} = \boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k)$, we discrete the scaled dynamics using the Runge-Kutta four method with discretization period T_{s}.

5.2. Anode ancillary

We use the notations $N_{\mathrm{O_2}}, N^{\mathrm{an}}_{\mathrm{H_2O}}$ (mol) to denote the numbers of moles of oxygen and water at the anode. The material balance equations for the anode ancillary are given by

$$\begin{aligned} \dot{N}_{\mathrm{O_2}} &= \dot{N}^{\mathrm{in}}_{\mathrm{O_2}} - \dot{N}^{\mathrm{out}}_{\mathrm{O_2}} + \dot{N}^{\mathrm{gn}}_{\mathrm{O_2}} \\ \dot{N}^{\mathrm{an}}_{\mathrm{H_2O}} &= \dot{N}^{\mathrm{an,in}}_{\mathrm{H_2O}} - \dot{N}^{\mathrm{an,out}}_{\mathrm{H_2O}} - \dot{N}^{\mathrm{mem}}_{\mathrm{H_2O}} - \dot{N}^{\mathrm{cons}}_{\mathrm{H_2O}} \end{aligned} \tag{35}$$

where $\dot{N}^{\mathrm{in}}_{\mathrm{O_2}}, \dot{N}^{\mathrm{an,in}}_{\mathrm{H_2O}}, \dot{N}^{\mathrm{out}}_{\mathrm{O_2}}, \dot{N}^{\mathrm{an,out}}_{\mathrm{H_2O}}$ (mol·s^{-1}) are anode inlet and outlet molar flow rates of oxygen and water respectively. One should note that $\dot{N}^{\mathrm{in}}_{\mathrm{O_2}}$ is zero because only input is water. $\dot{N}^{\mathrm{mem}}_{\mathrm{H_2O}}$ (mol·s^{-1}) is the molar flow rate of membrane water (water from anode to cathode through the membrane), $\dot{N}^{\mathrm{cons}}_{\mathrm{H_2O}}$ (mol·s^{-1}) is the molar flow rate of water consumed at the anode.

The rate of oxygen generated is

$$\dot{N}^{\mathrm{gn}}_{\mathrm{O_2}} = \frac{nI_{\mathrm{cell}}}{4F}\eta_F \tag{36}$$

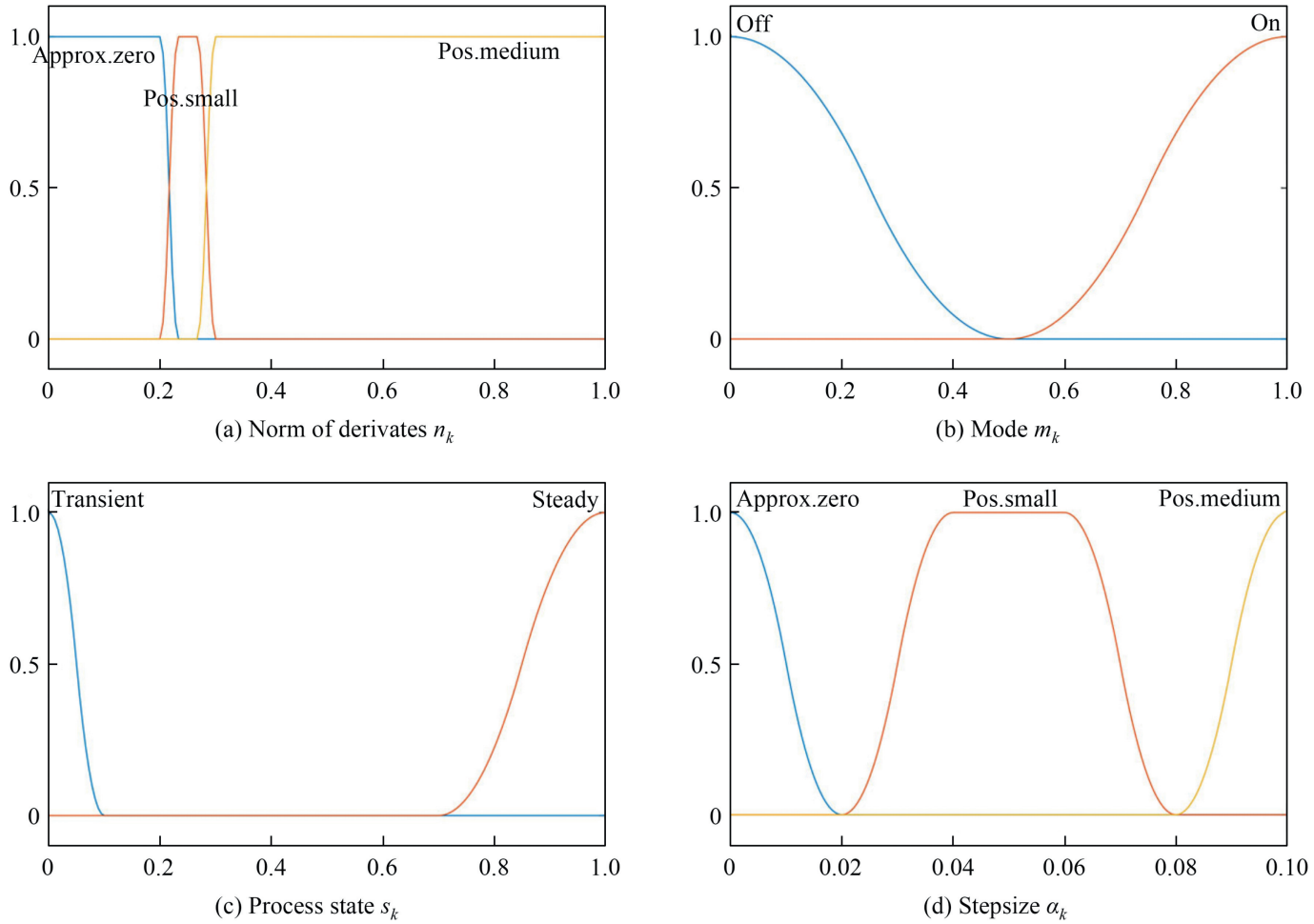


Fig. 1. Membership functions for the FLS inputs and outputs.

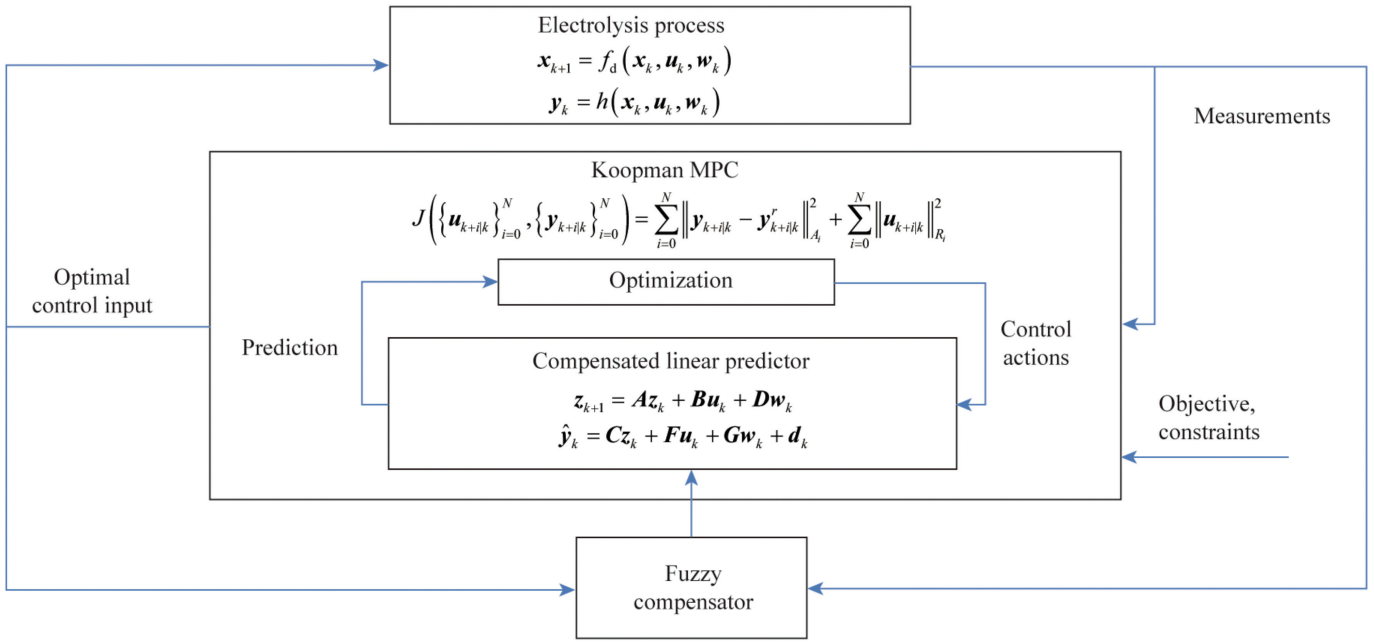


Fig. 2. Structure of the system with the FC-KMPC.

where n is the number of the electrolyzer cells, I_{cell} (A) is the cell current, $F = 96485$ C·mol^{-1} is the Faraday constant, $\eta_F = 0.99$ is Faraday efficiency.

The current through the cell is assumed to be uniformly distributed

$$I_{cell} = iA \tag{37}$$

where i (A·m^{-2}) is the current density and $A = 5 \times 10^{-3}$ m^2 is the active area of the membrane electrode assembly (MEA).

$$p_{O_2} = \frac{N_{O_2} RT_{el}}{V_{an}} \tag{38}$$

$$p^{an}_{H_2O} = \frac{N^{an}_{H_2O} RT_{el}}{V_{an}} \tag{39}$$

where $R = 8.3144$ J·mol^{-1}·K^{-1} is the universal gas constant, T_{el} (K) is the electrolyzer cell temperature, $V_{an} = 0.05$ m^3 is the anode volume, and the total anode pressure P_{an} (Pa) is

$$P_{an} = p_{O_2} + p^{an}_{H_2O} \tag{40}$$

The oxygen mole fraction at the anode outlet is as follows:

$$y_{O_2} = \frac{N_{O_2}}{N_{O_2} + N^{an}_{H_2O}} = \frac{p_{O_2}}{p_{O_2} + p^{an}_{H_2O}} = \frac{p_{O_2}}{P_{an}} \tag{41}$$

If the pressure difference across the nozzle is small, the anode total outlet flow rate $W_{an,out}$ (kg·s^{-1}) is given as follows:

$$W_{an,out} = k_{an,out}(P_{an} - P_{an,out}) \tag{42}$$

where $k_{an,out}$ (kg·Pa^{-1}·s^{-1}) is the nozzle outlet flow constant of the anode side, $P_{an,out}$ (Pa) is the anode out pressure.

The anode outlet mass flow rates of oxygen and water are given as follows

$$W^{out}_{O_2} = M_{O_2} \dot{N}^{out}_{O_2} = x_{O_2} W_{an,out} \tag{43}$$

$$W^{an,out}_{H_2O} = M_{H_2O} \dot{N}^{an,out}_{H_2O} = (1 - x_{O_2}) W_{an,out} \tag{44}$$

where M_{O_2} and M_{H_2O} (kg·mol^{-1}) are the molar mass of oxygen and water, respectively. The oxygen mass fraction x_{O_2} is calculated from

$$x_{O_2} = \frac{y_{O_2} M_{O_2}}{y_{O_2} M_{O_2} + (1 - y_{O_2}) M_{H_2O}} \tag{45}$$

5.3. Cathode ancillary

Similarly, the notations $N_{H_2}, N^{cat}_{H_2O}$ (mol) denote the numbers of moles of hydrogen and water at the cathode. The material balance equations for the cathode ancillary are given by

$$\begin{aligned} \dot{N}_{H_2} &= \dot{N}^{in}_{H_2} - \dot{N}^{out}_{H_2} + \dot{N}^{gn}_{H_2} \\ \dot{N}^{cat}_{H_2O} &= \dot{N}^{cat,in}_{H_2O} - \dot{N}^{cat,out}_{H_2O} + \dot{N}^{mem}_{H_2O} \end{aligned} \tag{46}$$

where $\dot{N}^{in}_{H_2}, \dot{N}^{cat,in}_{H_2O}, \dot{N}^{out}_{H_2}, \dot{N}^{cat,out}_{H_2O}$ (mol·s^{-1}) are cathode inlet and outlet molar flow rates of hydrogen and water respectively. $\dot{N}^{in}_{H_2}, \dot{N}^{cat,in}_{H_2O}$ are equal to zero since there is no inflows.

The rate of hydrogen generated is

$$\dot{N}^{gn}_{H_2} = \frac{nI_{cell}}{2F} \eta_F \tag{47}$$

The partial pressures of the species (hydrogen and water) at the cathode are

$$p_{H_2} = \frac{N_{H_2} RT_{el}}{V_{cat}} \tag{48}$$

$$p^{cat}_{H_2O} = \frac{N^{cat}_{H_2O} RT_{el}}{V_{cat}} \tag{49}$$

where $V_{cat} = 0.05$ m^3 is the cathode volume, and the total cathode pressure P_{cat} (Pa) is

$$P_{cat} = p_{H_2} + p^{cat}_{H_2O} \tag{50}$$

The hydrogen mole fraction at the cathode outlet is as follows:

$$y_{H_2} = \frac{N_{H_2}}{N_{H_2} + N^{cat}_{H_2O}} = \frac{p_{H_2}}{p_{H_2} + p^{cat}_{H_2O}} = \frac{p_{H_2}}{P_{cat}} \tag{51}$$

The cathode total outlet flow rate $W_{cat,out}$ (kg·s^{-1}) is

$$W_{cat,out} = k_{cat,out}(P_{cat} - P_{cat,out}) \tag{52}$$

where $k_{cat,out}$ (kg·Pa^{-1}·s^{-1}) is the cathode outlet flow constant, $P_{an,out}$ (Pa) is the anode out pressure.

The cathode outlet mass flow rates of hydrogen and water are given as follows

$$W^{out}_{H_2} = M_{H_2} \dot{N}^{out}_{H_2} = x_{H_2} W_{cat,out} \tag{53}$$

$$W^{cat,out}_{H_2O} = M_{H_2O} \dot{N}^{cat,out}_{H_2O} = (1 - x_{H_2}) W_{cat,out} \tag{54}$$

where M_{H_2} is the molar mass of hydrogen. The hydrogen mass fracton x_{H_2} is calculated from

$$x_{H_2} = \frac{y_{H_2} M_{H_2}}{y_{H_2} M_{H_2} + (1 - y_{H_2}) M_{H_2O}} \tag{55}$$

5.4. Membrane ancillary

The material balance equation for the membrane ancillary is given by

$$\dot{N}^{mem}_{H_2O} = \dot{N}^{eod}_{H_2O} + \dot{N}^{diff}_{H_2O} - \dot{N}^{pe}_{H_2O} \tag{56}$$

In practice, electro-osmotic drag is the dominant mechanism. This water transportation can be expressed as

$$\dot{N}^{eod}_{H_2O} = n_d \frac{nI_{cell}}{F} \tag{57}$$

where n_d is the electro-osmotic drag coefficient which is given as

$$n_d = 0.0029\lambda^2_{mem} + 0.05\lambda_{mem} - 3.4 \times 10^{-19} \tag{58}$$

where λ_{mem} is the arithmetic mean of both membrane content λ's for the anode and the cathode which are calculated by their own water activities.

Water diffusion through the membrane is given from Fick's first law of diffusion as follows:

$$\dot{N}_{H_2O}^{diff} = \frac{D_w An\left(C_{H_2O}^{mem,cat} - C_{H_2O}^{mem,an}\right)}{\delta_{mem}} \tag{59}$$

where D_w ($m^2 \cdot s^{-1}$) is the water diffusion coefficient, $\delta_{mem} = 2\times 10^{-4}$ m is the membrane thickness, and $C_{H_2O}^{mem,an}, C_{H_2O}^{mem,cat}$ ($mol \cdot m^{-3}$) are the water concentration at the anode and cathode surface of the membrane respectively.

The rate of water transport from the cathode to the anode across the membrane depends on the pressure gradient and this phenomenon can be evaluated by Darcy's law:

$$\dot{N}_{H_2O}^{pe} = \frac{K_{Darcy} An \rho_{H_2O}(P_{cat} - P_{an})}{\delta_{mem}\mu_{H_2O} M_{H_2O}} \tag{60}$$

where $K_{Darcy} = 1.58 \times 10^{-18}$ m^2 is the membrane permeability to water and $\mu_{H_2O} = 1.1 \times 10^{-3}$ $kg \cdot m^{-1} \cdot s^{-1}$ is the viscosity of water.

5.5. Voltage ancillary

The cell voltage in a PEM electrolyzer can be expressed as

$$V_{cell} = V_{oc} + V_{act} + V_{ohm} \tag{61}$$

The open-circuit voltage can be derived from the Nernst equation

$$V_{oc} = E_0 + \frac{RT_{el}}{2F}\ln\left(p_{H_2}\sqrt{p_{O_2}}\right) \tag{62}$$

where E_0 is the reversible potential.

The reversible potential can be expressed as

$$\begin{aligned} E_0 &= \frac{\Delta G}{2F} = \frac{\Delta H}{2F} - \frac{\Delta S}{2F}T_{el} \\ &= 1.229 - 0.0009(T_{el} - 298.15) \end{aligned} \tag{63}$$

where ΔG ($J \cdot mol^{-1}$) is the Gibbs free energy, $\Delta S = 163.1$ $J \cdot mol^{-1} \cdot K^{-1}$ is the entropy chance, $\Delta H = 2.8584 \times 10^5$ $J \cdot mol^{-1}$ is the enthalpy change.

The activation over-potential of the anode and cathode can be obtained by

$$V_{act} = \frac{RT_{el}}{2\alpha_{an}F}\sinh^{-1}\left(\frac{i}{2i_{0,an}}\right) + \frac{RT_{el}}{2\alpha_{cat}F}\sinh^{-1}\left(\frac{i}{2i_{0,cat}}\right) \tag{64}$$

where the charge transfer coefficients are taken as $\alpha_{an} = 0.5$ and $\alpha_{cat} = 0.5$, the exchange current density can be accepted as $i_{0,an} = 2 \times 10^{-3}$ $A \cdot m^{-2}$ and $i_{0,cat} = 2 \times 10^3$ $A \cdot m^{-2}$.

The ohmic over-potential is calculated by

$$V_{ohm} = iR_{ohm} = i\frac{\delta_{mem}}{\sigma_{mem}} \tag{65}$$

where R_{ohm} is the membrane resistance, σ_{mem} is the conductivity of the membrane which is calculated as follows:

$$\sigma_{mem} = (0.00514\lambda_{mem} - 0.00326)\exp\left(1268\left(\frac{1}{303} - \frac{1}{T_{el}}\right)\right) \tag{66}$$

In our study, we assume the PEM electrolyzer cells is connected in series, the operating voltage and current of the electrolyzer can be expressed as

$$V_{el} = nV_{cell} \tag{67}$$

$$I_{el} = I_{cell} \tag{68}$$

The series configuration is useful when higher voltage is required.

5.6. Storage ancillary

Produced hydrogen by electrolyzer is stored in bottle. Constant H_2 flow fills up the bottle until its pressure reaches up the electrolyzer cathode pressure. The dynamics of the storage is obtained as follows:

$$\dot{P}_b = z\frac{\dot{N}_{H_2O}^{cat,out} R T_b}{V_b} \tag{69}$$

where z is the compressibility factor of the hydrogen, which is a function of temperature and pressure [36].

5.7. Control objective

The control problem addressed in this study is the regulation of the anode and cathode pressures such that:

$$\begin{aligned} P_{an} &= P_{an}^r \\ P_{cat} &= P_{cat}^r \end{aligned} \tag{70}$$

Additionally, the following process constraints for variables are applied to the optimal control problem:

$$\begin{aligned} &P_{an,out} \le P_{an} \\ &P_b \le P_{cat,out} \le P_{cat} \\ &0 \le \dot{N}_{H_2O}^{an,in} \le 0.5 \\ &0 \le P_{an,out} \le 3 \times 10^5 \\ &0 \le P_{cat,out} \le 3 \times 10^5 \\ &0 \le I_{el} \le 50 \\ &323.15 \le T_{el} \le 353.15 \end{aligned} \tag{71}$$

6. Simulation Results

6.1. Open-loop prediction

Aim to employ EDMD scheme to identify the linear predictor (23) for system (34), we discretize the system with period $T_s = 0.2$ and simulate $N_{traj} = 1000$ trajectories over $N_{sim} = 100$ sampling periods. The state, input, output, and disturbance data are collected. The control and disturbance input trajectories are randomly generated according to the physical constraints (71).

In order to increase the simulation robustness, the elements of $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{u}$, and $\boldsymbol{w}$ are scaled to $[-1, 1]$, then the lifting functions are chosen as

$$\boldsymbol{\Psi}(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{x} \\ \left\|\boldsymbol{x} - \boldsymbol{c}_1\right\|_2^2 \ln\left\|\boldsymbol{x} - \boldsymbol{c}_1\right\|_2 \\ \vdots \\ \left\|\boldsymbol{x} - \boldsymbol{c}_{N_{rbf}}\right\|_2^2 \ln\left\|\boldsymbol{x} - \boldsymbol{c}_{N_{rbf}}\right\|_2 \end{pmatrix} \tag{72}$$

where the parameters of the radial basis function (RBF) $\boldsymbol{c}_i \in [-1, 1]^{n_x}$, $i = 1, \cdots, N_{rbf}$, are generated randomly. Here $N_{rbf} = 60$. The procedure is performed in EDMD, and the linear predictor is obtained.

For the sake of model validation, randomly generated input trajectories in Fig. 3 and the disturbance trajectory in Fig. 4 are applied to the PEM electrolyzer. Then the output trajectories from system (34), the linear predictor (23), the predictor based on the neural network, and the predictor based on local linearization of the dynamics at the initial condition $\boldsymbol{x}_0$ are compared in Fig. 5. We include more details about constructing the neural network and implementation of NNMPC in Appendix B.

For a sequence of predicted output $\{\widehat{\boldsymbol{y}}_k\}_{k=1}^{N_{\text{sim}}}$ and sequence of actual output $\{\boldsymbol{y}_k\}_{k=1}^{N_{\text{sim}}}$, the relative root mean square error (RRMSE) defined as

$$\text{RRMSE} = \sqrt{\frac{\sum_{k=1}^{N_{\text{sim}}} \|\widehat{\boldsymbol{y}}_k - \boldsymbol{y}_k\|_2^2}{\sum_{k=1}^{N_{\text{sim}}} \|\boldsymbol{y}_k\|_2^2}} \tag{73}$$

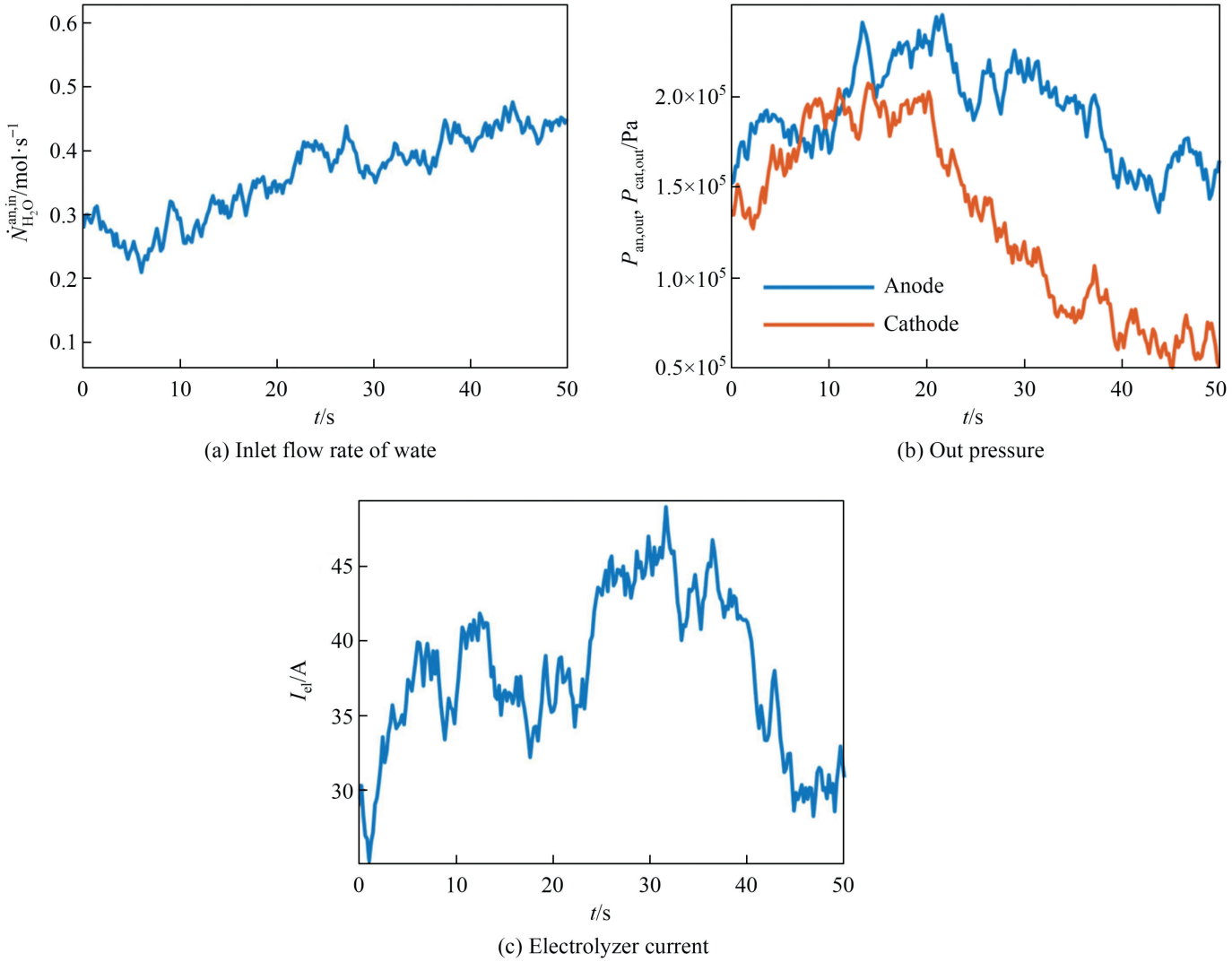
is used to evaluate the estimation accuracy. The linear predictor (23) and the predictor based on the neural network obtains an RRMSE of 0.0847 and 78.3716, respectively.

Although it is observed that the prediction of the lifting-based Koopman predictor properly follows the process dynamics, a plant-model mismatch between the trajectories still exists.

6.2. Closed-loop control

Based on the linear predictor derived from EDMD, we design a Koopman MPC incorporating fuzzy logic for reference tracking of the PEM electrolyzer under the randomly generated disturbance as shown in Fig. 6, Fig. 7 presents the closed-loop trajectories of the output variables induced by the nominal Koopman MPC, NNMPC (see Appendix B), SLMPC, and proposed FC-KMPC schemes.

It is observed that the output trajectories induced by the nominal Koopman MPC sometimes could not reach the reference value, while the proposed controller and SLMPC (requires linearization model at run time which can be unrealistic) can always regulate the anode pressure and cathode pressure to the set-points, since the fuzzy strategy can compensate for the influence of the plant-model mismatch effectively.



(a) Inlet flow rate of wate

(b) Out pressure

(c) Electrolyzer current

Fig. 3. Input trajectories for model validation.

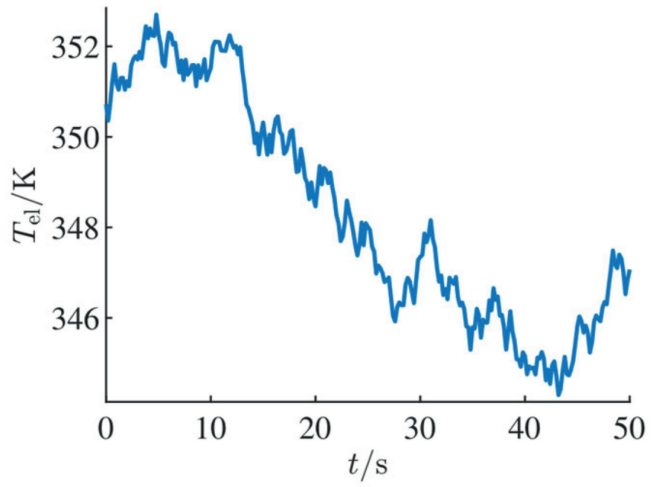


Fig. 4. Disturbance (electrolyzer temperature) trajectory for model validation.

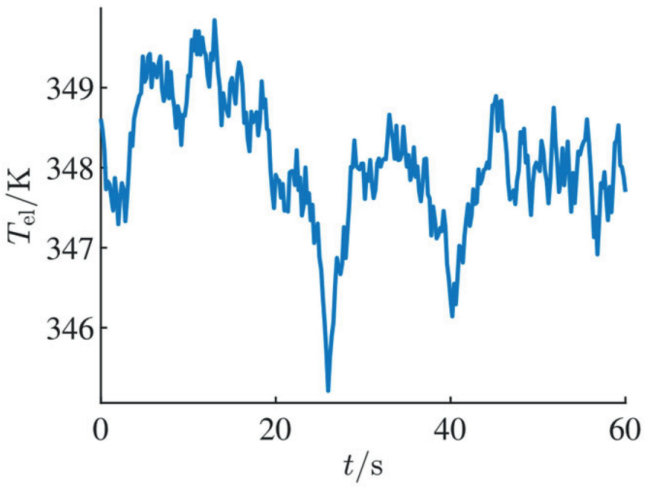
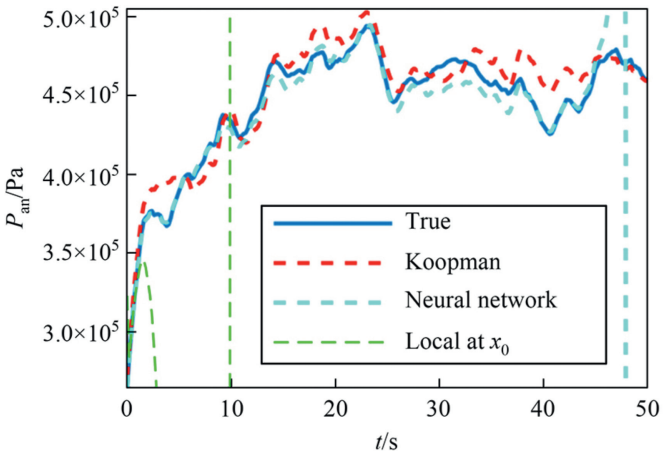
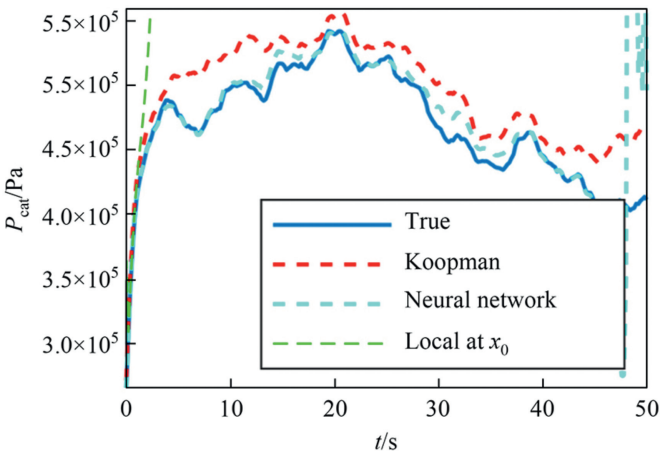


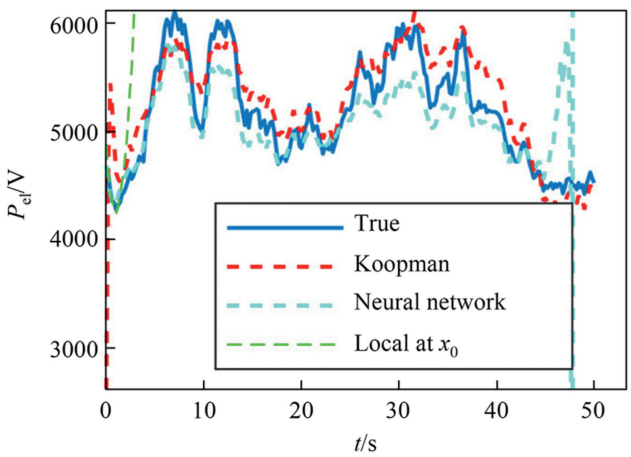
Fig. 6. Disturbance (electrolyzer temperature) trajectory of closed-loop control.



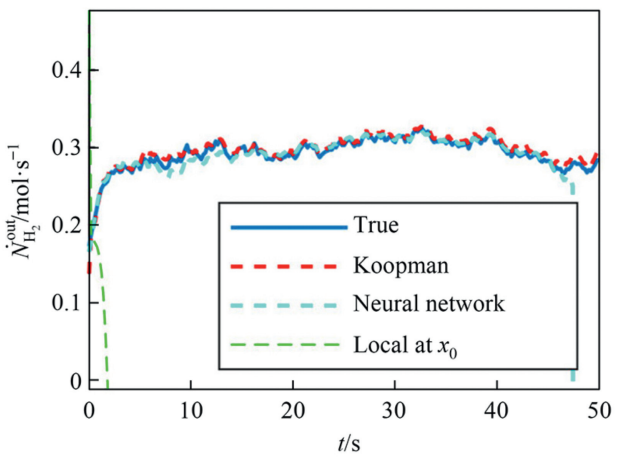
(a) Anode pressure



(b) Cathode pressure



(c) Operating voltage



(d) Outlet flow rate of hydrogen

Fig. 5. Output trajectories of the true dynamics and predictors.

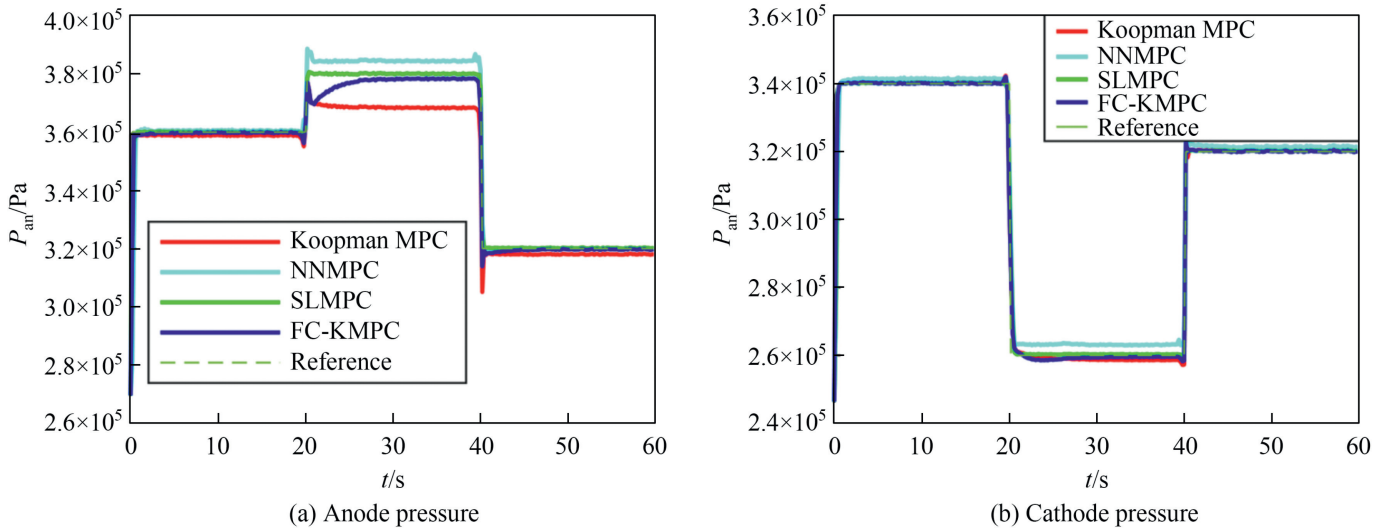


Fig. 7. Reference tracking results from the proposed FC-KMPC and other controllers.

7. Conclusions

This paper aims to develop a control-oriented model of PEM electrolyzer and achieve pressure regulation *via* the proposed FC-KMPC scheme. The first step in implementing the proposed controller involved creating a virtual process using a state-space model of the PEM electrolyzer to generate data for constructing a Koopman linear predictor. This model is then used with the EDMD method to identify the linear predictor and its accuracy was validated. A fuzzy logic system is then developed to estimate and compensate for any plant-model mismatches. The performance of the proposed controller is compared to that of the nominal Koopman MPC, NNMPC, and successive linearization MPC (SLMPC), and it is found that the FC-KMPC is able to effectively compensate for any mismatch and achieve zero steady-state offset.

CRediT Authorship Contribution Statement

Haokun Xiong: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. Lei Xie: Writing – review & editing, Supervision, Project administration, Funding acquisition. Cheng Hu: Writing – review & editing, Supervision, Formal analysis, Conceptualization. Hongye Su: Supervision, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by 2022 Zhejiang Provincial Science and Technology Plan Project (2022C01035).

Nomenclature

M	molar mass, $kg \cdot mol^{-1}$
$\dot{N}$	mass flow rate, $kg \cdot s^{-1}$
P, p	pressure, partial pressure, Pa
T	temperature, K
V	voltage, potential, V
W	mass flow rate, $kg \cdot s^{-1}$

References

[1] W. Lubitz, W. Tumas, Hydrogen: an overview, *Chem. Rev.* 107 (10) (2007) 3900–3903.
[2] S.A. Grigoriev, V.N. Fateev, D.G. Bessarabov, P. Millet, Current status, research trends, and challenges in water electrolysis science and technology, *Int. J. Hydrogen Energy* 45 (49) (2020) 26036–26058.
[3] J. Koponen, A. Kosonen, V. Ruuskanen, K. Huoman, M. Niemelä, J. Ahola, Control and energy efficiency of PEM water electrolyzers in renewable energy systems, *Int. J. Hydrogen Energy* 42 (50) (2017) 29648–29660.
[4] P. Millet, R. Ngameni, S.A. Grigoriev, V.N. Fateev, Scientific and engineering issues related to PEM technology: water electrolysers, fuel cells and unitized regenerative systems, *Int. J. Hydrogen Energy* 36 (6) (2011) 4156–4163.
[5] G. Correa, P. Marocco, P. Muñoz, T. Falagüerra, D. Ferrero, M. Santarelli, Pressurized PEM water electrolysis: dynamic modelling focusing on the cathode side, *Int. J. Hydrogen Energy* 47 (7) (2022) 4315–4327.
[6] B. Han, S.M. Steen III, J.K. Mo, F.Y. Zhang, Electrochemical performance modeling of a proton exchange membrane electrolyzer cell for hydrogen energy, *Int. J. Hydrogen Energy* 40 (22) (2015) 7006–7016.
[7] H. Görgün, Dynamic modelling of a proton exchange membrane (PEM) electrolyzer, *Int. J. Hydrogen Energy* 31 (1) (2006) 29–38.
[8] Z. Abdin, C.J. Webb, E.M. Gray, Modelling and simulation of a proton exchange membrane (PEM) electrolyser cell, *Int. J. Hydrogen Energy* 40 (39) (2015) 13243–13257.
[9] N.V. Dale, M.D. Mann, H. Salehfar, Semiempirical model based on thermodynamic principles for determining 6kW proton exchange membrane electrolyzer stack characteristics, *J. Power Sources* 185 (2) (2008) 1348–1353.
[10] R. Datta, D.J. Martino, Y. Dong, P. Choi, Modeling of PEM water electrolyzer. PEM Electrolysis for Hydrogen Production—Principles and Applications, CRC Press, Boca Raton, FL, USA, 2016.
[11] S. Dutta, S. Shimpalee, J.W. van Zee, Numerical prediction of mass-exchange between cathode and anode channels in a PEM fuel cell, *Int. J. Heat Mass Tran.* 44 (11) (2001) 2029–2042.
[12] C. Marr, X. Li, An engineering model of proton exchange membrane fuel cell performance, *ARI Int. J. Phys. Eng. Sci.* 50 (4) (1997) 190–200.
[13] T. Yigit, O.F. Selamet, Mathematical modeling and dynamic Simulink simulation of high-pressure PEM electrolyzer system, *Int. J. Hydrogen Energy* 41 (32) (2016) 13901–13914.
[14] A. Tabanjat, M. Becherif, M. Emziane, D. Hissel, H.S. Ramadan, B. Mahmah, Fuzzy logic-based water heating control methodology for the efficiency enhancement of hybrid PV–PEM electrolyser systems, *Int. J. Hydrogen Energy* 40 (5) (2015) 2149–2161.
[15] J.T. Pukrushpan, huei Peng, A.G. Stefanopoulou, Control-oriented modeling and analysis for automotive fuel cell systems, *J. Dyn. Syst. Meas. Control* 126 (1) (2004) 14–25.

[16] C. Kunusch, P.F. Puleston, M.A. Mayosky, J. Riera, Sliding mode strategy for PEM fuel cells stacks breathing control using a super-twisting algorithm, *IEEE Trans. Control Syst. Technol.* 17 (1) (2009) 167–174.
[17] I. Matraji, S. Laghrouche, M. Wack, Pressure control in a PEM fuel cell *via* second order sliding mode, *Int. J. Hydrogen Energy* 37 (21) (2012) 16104–16116.
[18] A. Majumdar, M. Haas, I. Elliot, S. Nazari, Control and control-oriented modeling of PEM water electrolyzers: a review, *Int. J. Hydrogen Energy* 48 (79) (2023) 30621–30641.
[19] B. Flamm, C. Peter, F.N. Büchi, J. Lygeros, Electrolyzer modeling and real-time control for optimized production of hydrogen gas, *Appl. Energy* 281 (2021) 116031.
[20] R. Findeisen, F. Allgöwer, An introduction to nonlinear model predictive control, in: 21st Benelux Meeting on Systems and Control, Veldhoven, 2002.
[21] M.M. Morato, J.E. Normey-Rico, O. Sename, Model predictive control design for linear parameter varying systems: a survey, *Annu. Rev. Control* 49 (2020) 64–80.
[22] J.H. Lee, N.L. Ricker, Extended Kalman filter based nonlinear model predictive control. 1993 American Control Conference, IEEE, San Francisco, CA, USA, 1993.
[23] M. Korda, I. Mezić, Linear predictors for nonlinear dynamical systems: koopman operator meets model predictive control, *Automatica* 93 (2018) 149–160.
[24] M.O. Williams, I.G. Kevrekidis, C.W. Rowley, A data–driven approximation of the koopman operator: extending dynamic mode decomposition, *J. Nonlinear Sci.* 25 (6) (2015) 1307–1346.
[25] M. Korda, Y. Susuki, I. Mezić, Power grid transient stabilization using Koopman model predictive control, *IFAC-PapersOnLine* 51 (28) (2018) 297–302.
[26] S.H. Son, H.K. Choi, J.S.I. Kwon, Application of offset-free Koopman-based model predictive control to a batch pulp digester, *AIChE J.* 67 (9) (2021) e17301.
[27] L. Magdalena, Fuzzy Rule-Based Systems, Springer Handbook of Computational Intelligence. Springer Berlin Heidelberg, 2015, pp. 3–218.
[28] H. Ashraf, M.M. Elkholy, S.O. Abdellatif, A.A. El-Fergany, Synergy of neuro-fuzzy controller and tuna swarm algorithm for maximizing the overall efficiency of PEM fuel cells stack including dynamic performance, *Energy Convers. Manag. X* 16 (2022) 100301.
[29] A.T. Nguyen, T. Taniguchi, L. Eciolaza, V. Campos, R. Palhares, M. Sugeno, Fuzzy control systems: past, present and future, *IEEE Comput. Intell. Mag.* 14 (1) (2019) 56–68.
[30] M.H. Cano, S. Kelouwani, K. Agbossou, Y. Dubé, Power management system for off-grid hydrogen production based on uncertainty, *Int. J. Hydrogen Energy* 40 (23) (2015) 7260–7272.
[31] N.A. Spielberg, M. Brown, J.C. Gerdes, Neural network model predictive motion control applied to automated driving with unknown friction, *IEEE Trans. Control Syst. Technol.* 30 (5) (2022) 1934–1945.
[32] A. Surana, A. Banaszuk, Linear observer synthesis for nonlinear systems using Koopman Operator framework, *IFAC-PapersOnLine* 49 (18) (2016) 716–723.
[33] A. Surana, Koopman operator framework for time series modeling and analysis, *J. Nonlinear Sci.* 30 (5) (2020) 1973–2006.
[34] J.N. Kutz, S.L. Brunton, B.W. Brunton, J.L. Proctor, Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems, SIAM Press U.S, 2016.
[35] R.R. Yager, J. Kacprzyk, G. Beliakov, Recent Developments in the Ordered Weighted Averaging Operators: Theory and Practice, Springer Berlin Heidelberg, 2011.
[36] E.W. Lemmon, M.L. Huber, J.W. Leachman, Revised standardized equation for hydrogen gas densities for fuel consumption applications, *J. Res. Natl. Inst. Stand. Technol.* 113 (6) (2008) 341–350.

Appendix A. Explicit Expressions for Matrices of MPC Problem

The matrices in the "compact" and "dense-form" MPC problem are given by

$$\boldsymbol{\Psi}_A = \begin{pmatrix} \boldsymbol{I}_{n_z} \\ \boldsymbol{A} \\ \boldsymbol{A}^2 \\ \vdots \\ \boldsymbol{A}^N \end{pmatrix}, \boldsymbol{\Psi}_B = \begin{pmatrix} \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{B} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{AB} & \boldsymbol{B} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{A}^{N-1}\boldsymbol{B} & \boldsymbol{A}^{N-2}\boldsymbol{B} & \cdots & \boldsymbol{B} & \boldsymbol{0} \end{pmatrix},$$

$$\boldsymbol{\Psi}_D = \begin{pmatrix} \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{D} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{AD} & \boldsymbol{D} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{A}^{N-1}\boldsymbol{D} & \boldsymbol{A}^{N-2}\boldsymbol{D} & \cdots & \boldsymbol{D} & \boldsymbol{0} \end{pmatrix},$$

$$\overline{\boldsymbol{C}} = \boldsymbol{I}_{N+1} \otimes \boldsymbol{C},\ \overline{\boldsymbol{F}} = \boldsymbol{I}_{N+1} \otimes \boldsymbol{F},\ \overline{\boldsymbol{G}} = \boldsymbol{I}_{N+1} \otimes \boldsymbol{G},$$
$$\overline{\boldsymbol{E}} = \mathrm{diag}(\boldsymbol{E}_0, \cdots, \boldsymbol{E}_N), \overline{\boldsymbol{L}} = \mathrm{diag}(\boldsymbol{L}_0, \cdots, \boldsymbol{L}_N),$$
$$\boldsymbol{M} = \mathrm{diag}(\boldsymbol{M}_0, \cdots, \boldsymbol{M}_N), \overline{\boldsymbol{b}} = \begin{pmatrix} \boldsymbol{b}_0^{\mathrm{T}} & \boldsymbol{b}_1^{\mathrm{T}} & \cdots & \boldsymbol{b}_N^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}$$
$$\boldsymbol{E} = \overline{\boldsymbol{E}} + \boldsymbol{M}\overline{\boldsymbol{C}}, \boldsymbol{L} = \overline{\boldsymbol{L}} + \boldsymbol{M}\overline{\boldsymbol{F}}, \boldsymbol{b} = \overline{\boldsymbol{b}} - \boldsymbol{M}\overline{\boldsymbol{G}}\boldsymbol{W}_k$$
$$\boldsymbol{H} = (\overline{\boldsymbol{C}}\boldsymbol{\Psi}_B + \overline{\boldsymbol{F}})^{\mathrm{T}} \boldsymbol{\Lambda} (\overline{\boldsymbol{C}}\boldsymbol{\Psi}_B + \overline{\boldsymbol{F}}) + \boldsymbol{R}$$
$$\boldsymbol{h} = 2(\overline{\boldsymbol{C}}\boldsymbol{\Psi}_B + \overline{\boldsymbol{F}})^{\mathrm{T}} \boldsymbol{\Lambda} \left[\overline{\boldsymbol{C}}\boldsymbol{\Psi}_A \boldsymbol{z}_{k|k} + (\overline{\boldsymbol{C}}\boldsymbol{\Psi}_D + \overline{\boldsymbol{G}})\boldsymbol{W}_k - \boldsymbol{Y}_k^{\mathrm{r}} \right]$$

Appendix B. Predictor Based on Neural Network and NNMPC Framework

To construct the input to the network, the states $\boldsymbol{x}$, the control inputs $\boldsymbol{u}$ and disturbances $\boldsymbol{w}$ are formed into the input $\boldsymbol{r} = \begin{pmatrix} \boldsymbol{x}^{\mathrm{T}} & \boldsymbol{u}^{\mathrm{T}} & \boldsymbol{w}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}$ to the network. The network consists of a simple two-layer feedforward network with $n_\diamond = 20$ units in the hidden layer. The input to each activation is represented using $\boldsymbol{q}_\diamond$ and the output of the activation using $\boldsymbol{a}_\diamond$, where $\diamond$ represents the layer number (note that the counting index $\diamond$ starts with the first hidden layer up to the output layer). As shown below, the network can include a variety of nonlinear activation functions, where $\mathrm{act}_\diamond(\boldsymbol{q}_\diamond)$ represents the activation function at a given layer. $\boldsymbol{W}_{\mathrm{NN},\diamond}$ are the network weights and $\boldsymbol{b}_{\mathrm{NN},\diamond}$ are the network biases. The output of the network $\boldsymbol{a}_2 = \begin{pmatrix} \boldsymbol{x}'^{\mathrm{T}} & \boldsymbol{y}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}$ consists of the successor state of the system $\boldsymbol{x}' = \boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})$ and system output $\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})$.

$$\boldsymbol{q}_1 = \boldsymbol{W}_{\mathrm{NN},1}^{\mathrm{T}} \boldsymbol{r} + \boldsymbol{b}_{\mathrm{NN},1}$$
$$\boldsymbol{a}_1 = \mathrm{act}_1(\boldsymbol{q}_1)$$
$$\boldsymbol{q}_2 = \boldsymbol{W}_{\mathrm{NN},2}^{\mathrm{T}} \boldsymbol{a}_1 + \boldsymbol{b}_{\mathrm{NN},2}$$
$$\boldsymbol{a}_2 = \mathrm{act}_2(\boldsymbol{q}_2)$$

Types of activation functions used in the hidden layer and output layer are different. All hidden layers usually use the tanh activation function. However, the output layer will typically use the linear activation function from the hidden layers.

$$\mathrm{act}_1(\boldsymbol{q}) = \tanh(\boldsymbol{q}) = \frac{1 - \exp(-2\boldsymbol{q})}{1 + \exp(-2\boldsymbol{q})}$$
$$\mathrm{act}_2(\boldsymbol{q}) = \mathrm{purelin}(\boldsymbol{q}) = \boldsymbol{q}$$

The data $\boldsymbol{x}_{(k)}, \boldsymbol{x}'_{(k)}, \boldsymbol{u}_{(k)}, \boldsymbol{w}_{(k)}, \boldsymbol{y}_{(k)}$ used to train the network is the same as the data used to approximate matrix of the Koopman operator (see Eq. (21) and Eq. (22)). The network is optimized using the gradient descent optimization algorithm with momentum. Including the neural network model in the optimization results in

the following neural network model predictive control (NNMPC) problem:

$$
\begin{aligned}
\min_{\boldsymbol{u}_{k+i|k},\boldsymbol{y}_{k+i|k}} \quad & J\left(\left\{\boldsymbol{u}_{k+i|k}\right\}_{i=0}^{N},\left\{\boldsymbol{y}_{k+i|k}\right\}_{i=0}^{N}\right) \\
\text{s.t.} \quad & \begin{pmatrix}\boldsymbol{x}_{k+i+1|k}\\ \boldsymbol{y}_{k+i|k}\end{pmatrix}=\boldsymbol{f}_{\mathrm{NN}}\left(\begin{pmatrix}\boldsymbol{x}_{k+i|k}\\ \boldsymbol{u}_{k+i|k}\\ \boldsymbol{w}_{k+i|k}\end{pmatrix}\right), \quad i=0,\cdots,N \\
& \boldsymbol{E}_i\boldsymbol{z}_{k+i|k}+\boldsymbol{L}_i\boldsymbol{u}_{k+i|k}+\boldsymbol{M}_i\boldsymbol{y}_{k+i|k}\preceq\boldsymbol{b}_i, \quad i=0,\cdots,N \\
& \boldsymbol{z}_{k|k}=\boldsymbol{z}_k,\boldsymbol{w}_{k|k}=\boldsymbol{w}_k
\end{aligned}
$$