Digital Chemical Engineering 15 (2025) 100237

Contents lists available at ScienceDirect

Digital Chemical Engineering

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

Review article

Machine learning in modeling, analysis and control of electrochemical reactors: A tutorial review

Wenlong Wang [a], Zhe Wu [a], Dominic Peters [b], Berkay Citmaci [b], Carlos G. Morales-Guio [b,*], Panagiotis D. Christofides [b,c,**]

[a] Department of Chemical and Biomolecular Engineering, National University of Singapore, 117585, Singapore
[b] Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA, 90095-1592, USA
[c] Department of Electrical and Computer Engineering, University of California, Los Angeles, CA 90095-1592, USA

ARTICLE INFO

Keywords:
Electrochemical reactors
Machine learning modeling
Reactor operation
Process control
Model predictive control
Real-time implementation

ABSTRACT

Electrochemical reactors play a critical role in various industrial sectors, including energy storage, chemical production, and environmental engineering. The complexity of these systems – arising from coupled electrochemical reactions with mass, heat and charge transport phenomena – poses significant challenges in modeling, analysis, and control. Machine learning (ML) has emerged as a promising tool for addressing these challenges by providing data-driven solutions to complex process modeling, optimization, and advanced control. This tutorial review explores the state-of-the-art applications of ML in electrochemical reactor systems, including ML-based modeling techniques and ML-based advanced control strategies, followed by the discussions of practical challenges and their solutions. An electrochemical carbon dioxide (CO_2) reduction reactor is used as a demonstration example to show the effectiveness of various modeling and control methods. In addition, an integrated data infrastructure platform is presented for the digitalization and control of the electrochemical CO_2 reduction reactor. By identifying current gaps and future opportunities, this article provides a roadmap for leveraging ML tools to improve the analysis and operation of electrochemical reactors.

1. Introduction

Electrochemical reactors are devices that facilitate the interconversion between electrical and chemical energy through the transfer of electrons and the formation and breaking of chemical bonds. Chemical reactions that occur in electrochemical reactors involve electron transfer steps driven by electric potential gradients at the electrode interface. A typical electrochemical reactor consists of several key components, including two electrodes (anode and cathode) on which half-cell electrochemical reactions take place, an electrolyte that conducts charges through the movement of ions between the two electrodes, and often an ion selective membrane that controls and directs ion flow between the electrodes. Unlike traditional chemical reactors, which usually require the presence of high temperatures to achieve high production rates, electrochemical reactors can operate under mild temperature conditions at high reaction rates, which could eventually lead to an increase in energy efficiency and a reduced carbon footprint if powered by renewable electricity. Higher energy efficiency makes electrochemical reactors attractive for a wide range of applications, notably in chemical manufacturing, renewable energy storage, and wastewater treatment. The significance of electrochemical reactors is growing as sustainability and efficiency gain importance, which has driven research in the modeling, control, and optimization of such reactors to further enhance operational performance and process scalability.

However, accurate first-principles modeling and precise control of electrochemical reactors are challenging tasks due to the complex, multiscale, nonlinear dynamics of electrochemical reactions and the difficulties in obtaining real-time measurements of key process variables. Specifically, electrochemical reactors often facilitate multiple reactions that occur at different length and time scales, each with complex kinetics governed by different sets of coupled nonlinear equations. These reactions are sensitive to disturbances in process variables such as current density, reaction temperature, and reactant concentration, putting greater demands on the robustness of predictive models. Furthermore, electrode degradation, dendrite formation, and fouling can irreversibly affect reactor dynamics, which may inhibit the development of predictive models that maintain accuracy over time. To achieve the desired control performance for electrochemical reactors, several problems need to be addressed, particularly those related to

* Corresponding author.
** Corresponding author at: Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA, 90095-1592, USA.
E-mail addresses: moralesguio@ucla.edu (C.G. Morales-Guio), pdc@seas.ucla.edu (P.D. Christofides).

https://doi.org/10.1016/j.dche.2025.100237
Received 3 March 2025; Received in revised form 14 April 2025; Accepted 14 April 2025
Available online 25 April 2025

the real-time measurement of process variables. For example, while process states such as current, voltage, pressure, and temperature can typically be measured in real-time, other critical variables such as concentration and pH often rely on inferential techniques that require additional processing time. The delays resulting from data acquisition and processing (e.g., noise filtering) can potentially lead to suboptimal or unreliable control responses due to the time lag between process variable measurement and control action implementation.

Over the past few decades, machine learning (ML) techniques have garnered considerable attention for their effectiveness in modeling and controlling chemical processes by capturing complex nonlinear dynamics (e.g., Wu et al. (2025)). Machine learning (ML) methods significantly simplify the development of predictive models for complex chemical processes by learning process dynamics directly from data, especially when first-principles models are unavailable. This capability is particularly valuable in fields like electrochemistry, where typical reaction mechanisms are highly complex and not fully understood. Additionally, ML models have been successfully integrated into model predictive control (MPC) to develop ML-MPC (please see, Wu et al. (2025), and the references therein), where optimal control actions are determined by solving real-time ML-based optimization problems. Using data-driven models, ML-MPC can effectively predict and optimize reactor performance without requiring comprehensive first-principles knowledge of the electrochemical processes, and therefore, is particularly well-suited for controlling electrochemical reactors, especially in scenarios where electrochemical reaction pathways are newly discovered.

Existing reviews on ML applications in electrochemical reactors primarily focus on the predicting conversion performance or optimal designs of electrode materials. For example, Refs. Wang et al. (2020), Mistry et al. (2021), Shirkoohi et al. (2022), Chen et al. (2022), Giordano et al. (2023), Ming et al. (2023), Li et al. (2024), Liu et al. (2024) and Ji et al. (2024) introduce how ML techniques can be utilized effectively to predict the performance (e.g., efficiency and durability) of electrochemical reactors while Refs. Wang et al. (2020), Liu et al. (2020), Ding et al. (2022), Tamtaji et al. (2022), Shirkoohi et al. (2022), Chen et al. (2022), Sun and Liu (2023), Yang et al. (2023), Wu and Li (2023), Ming et al. (2023), Iqbal et al. (2024), Park and Lee (2024), Ding et al. (2024) and Gholizadeh et al. (2024) summarize recent progress in the rational or optimal design of electrode catalysts via ML approaches. In addition, Refs. Mistry et al. (2021), Li et al. (2024), Ji et al. (2024) and Gholizadeh et al. (2024) highlight the critical role of ML techniques in elucidating the complex mechanisms of electrochemical reactions, and Refs. Li et al. (2024) and Park and Lee (2024) showcase the effectiveness of optimizing operational conditions of electrochemical reactors using ML techniques. Since ML models are generally black-box models that lack interpretability, Refs. Wang et al. (2020), Guo et al. (2022) and Ji et al. (2024) review physics-informed ML models for the development of electrochemical reactors, where physical knowledge is utilized to increase the accuracy of ML models. Although mathematical modeling and simulation of electrochemical reactors have been extensively reviewed in Refs. Walsh and de Léon (2018), Muddemann et al. (2019), Catañeda et al. (2019), Noël et al. (2019), Perry et al. (2020), Lin et al. (2020), Zhao et al. (2021), Rivera et al. (2021), Zhou et al. (2023), Mehta and Gupta (2024) and Granados-Fernández et al. (2024), there are currently no reviews that focus on modeling and control of electrochemical reactors using ML-based approaches. Therefore, this review aims to bridge this gap by providing an overview of recent advances in ML-based modeling and control of electrochemical reactors.

The remaining contents of this review are organized into six sections. Section 2 introduces an example of CO_2 electrochemical reactor, which will be used as the demonstration example throughout this article to discuss the development and performance of ML modeling and control methods. In Section 3, we present a systematic review of ML techniques for modeling electrochemical reactors. In Section 4, we discuss the development of ML-MPC for electrochemical reactors and provide solutions to practical problems such as data availability, delayed measurements, state estimation, and plant-model mismatch. In Section 5, we demonstrate an integrated platform for the digitalization and control of the electrochemical CO_2 reduction reactor using ML tools. Section 6 concludes the review with open research questions and future directions for ML-based modeling and control of electrochemical reactors.

2. A CO_2 reduction electrochemical reactor example

In this section, we first present an overview of electrochemical processes, and then introduce an electrochemical reactor example that will be used throughout this article to show the application of ML-based modeling and control methods. Specifically, a rotating cylinder electrode (RCE) reactor designed for CO_2 reduction (CO_2R) into various chemical products has been developed at the University of California, Los Angeles (UCLA). In the RCE reactor, the applied potential strongly influences the reaction energetics while the electrode rotation speed affects the hydrodynamic boundary layer and the film mass transfer coefficient through convective and diffusive transport. Specifically, when the electrode rotates above 400 RPM (revolutions per minute), the mass transfer of molecules and ions to the electrode is enhanced. This RCE reactor is developed to investigate the impact of mass transfer and reaction kinetics on productivity, and provides a platform for investigating machine learning-based modeling, optimization, and control for CO_2 electrochemical reactors.

2.1. An overview of electrochemical processes

Electrochemical processes involve chemical reactions generating or driven by electrical energy via the transfer of electrons. These reactions typically occur at the interface of an electrode and an electrolyte, which enables the interconversion between chemical and electrical energy. Electrodes are classified as either cathodes or anodes based on the type of reactions occurring at their conductive surfaces under the flow of electrons. Electrodes where reduction reactions occur are identified as cathodes, while anodes refer to electrodes that facilitate oxidation reactions. To sustain the loss of electrons at the anode and the gain of electrons at the cathode, an electrolyte is placed between the cathode and the anode. When dissolved, the electrolyte dissociates into positive ions (cations) and negative ions (anions), which disperse uniformly throughout the solvent. These ions act as a chemical conductor that facilitates the flow of electricity within the solution. In addition, a membrane is typically introduced to control the flow of ions while preventing the undesired mixing of reactants or products across the electrodes. The membrane electrode assembly functions by selectively allowing the passage of specific ions (e.g., protons or hydroxide ions) between the cathode and the anode.

As mentioned above, electrochemical processes involve chemical reactions that are either driven by electrical energy or generate electrical energy. These processes correspond to two distinct types of reactors: the electrochemical cell and the electrolytic cell. In electrochemical cells, chemical energy is converted into electrical energy through spontaneous redox reactions. Fuel cells and primary batteries (e.g., a galvanic cell) are typical examples of electrochemical cells. In contrast, electrolytic cells have an external power source that can drive non-spontaneous redox reactions to occur, enabling the conversion of electrical energy to chemical energy. Common examples of electrolytic cells include various electrolyzer-based electrolysis processes. Note that secondary batteries (i.e., rechargeable batteries) are considered as electrolytic cells when they are charging.

2.2. Process overview and reactor setup

Electrochemical CO_2 reduction on copper (Cu) is a complex process. There are 17 chemicals produced, and their reaction pathways are complicated because processes of different time scales, including mass and charge transfer, adsorption and desorption, and surface reaction, are convoluted and involve multiple reaction intermediates. The mass transport characteristics of an electrochemical system affect the transfer of the reactant to the catalyst surface as well as the removal of intermediates and products away from the surface. Among various products generated from this electrochemical reactor from CO_2R on polycrystalline Cu, hydrogen (H_2), carbon monoxide (CO), methane (CH_4), and ethylene (C_2H_4) are in the gas phase and can be detected using gas chromatography (GC). The relevant reactions for these products are shown below:

$$2H_2O + 2e^- \rightarrow H_2 + 2OH^- \quad (1a)$$

$$CO_2 + H_2O + 2e^- \rightarrow CO + 2OH^- \quad (1b)$$

$$CO_2 + 6H_2O + 8e^- \rightarrow CH_4 + 8OH^- \quad (1c)$$

$$2CO_2 + 8H_2O + 12e^- \rightarrow C_2H_4 + 12OH^- \quad (1d)$$

The production of gas products (C_2H_4, CH_4, CO, and H_2) measured by the GC are the controlled variables of the feedback control system. The temperature-programmed GC sensor separates molecules in internal separation columns on the basis of molecular elution times. Subsequently, separated molecules were detected using a thermal conductivity detector (TCD) and a flame ionization detector (FID) in the GC system. In this case study, the temperature program runs for 14.33 min, after which the heated column oven requires 6 min of cooling time before the next injection is taken. As a result, the GC has a sampling period of 20.33 min. The delay in GC analysis limits the applications of real-time control, but this can be overcome to some extent through the development of an ML-based estimator as we will discuss below, which allows for the calculation of real-time gas compositions in the cell overhead.

In the UCLA experimental system, all components of the experimental device are digitalized using the Laboratory Virtual Instrument Engineering Workbench (*LabVIEW*) software. The experimental system involves GC signals (Fig. 1: light-green box), controllers (Fig. 1: purple box), and actuators (Fig. 1: red and dark green boxes), and it is fully automated with Python scripts that are integrated into a *LabVIEW* interface. Fig. 1 shows the experimental RCE reactor at UCLA. The specific feedback controller parameters are determined in advance. The control system in closed-loop experiments is designed to drive the states (i.e., C_2H_4, CH_4, CO, and H_2) to their desired setpoints. The process data flow is connected to the database of a smart manufacturing innovation platform, which will be introduced in Section 5. The RCE system consists of two electrode chambers divided by an anion-exchange membrane, a mass flow controller (MFC), a potentiostat, a temperature controller, and a modulated speed rotator (MSR). In the experiment, pure CO_2 gas is fed at a fixed mass flowrate at 20 mL min^{-1} into both the cathode chamber, where nanoporous Cu cylindrical electrode is rotating in 0.2 M $KHCO_3$ electrolyte solution, and the anode (Pt foil) chamber. CO_2 and H_2O molecules are transformed into 12 liquid-phase and 5 gas-phase (H_2, CO, CH_4, C_2H_4, and C_2H_6) products. Hydrodynamics and convective mass transport can be regulated systemically through the control of electrode rotation speed actuated by the MSR. Furthermore, the potentiostat can set the applied potential on the working electrode, and measurements are taken using Ag/AgCl as the reference electrode. Thus, the reaction kinetics and diffusion effects can be deconvoluted by running experiments at multiple applied potential and electrode rotation speeds. Finally, the electrochemical cell is hermetically sealed so that gas phase products can be quantified by the GC sensing device. An automated GC code is written for triggering injections, peak detection, baselining, and calculation of the areas

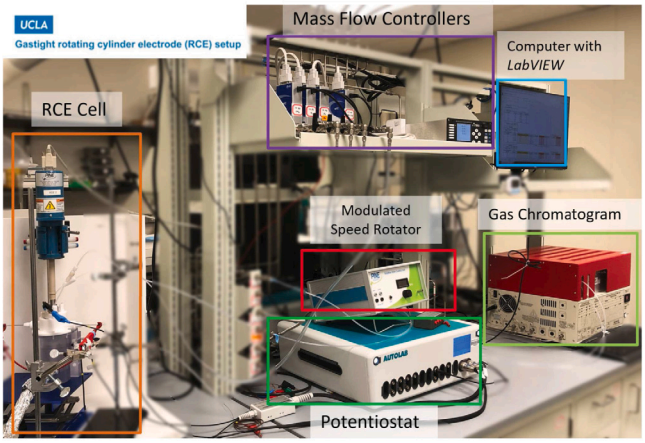


Fig. 1. Gastight RCE reactor setup at UCLA that includes the RCE electrochemical cell and electronic sensors along with mass and energy actuators. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

under the peaks to quantify the gas phase concentrations in ppm using available calibration data, as explained in Çıtmacı et al. (2022b). One GC injection takes 14.33 min to complete and is followed by 6 min of cooling before the next GC injection. Thus, when a GC measurement is obtained, it is delayed and is related to the reactor overhead gas concentrations of 14.33 min ago. The liquid phase products accumulate in the electrolyte solution and are measured by NMR (nuclear magnetic resonance) at the end of the experiment. In this example, the main output of the reactor is the production rates, denoted by $r_{C_2H_4 \mid CO}$ for C_2H_4 and CO. The GC measures the concentrations in ppm and these concentrations are converted to production rates via the following equation:

$$r_{C_2H_4 \mid CO} = \frac{C^{ppm}_{C_2H_4 \mid CO}}{10^6} \times \frac{\dot{V}_{CO_2}}{60 \times 10^6} \times \frac{P}{RT} \quad (2)$$

where $C^{ppm}_{C_2H_4 \mid CO}$ is the concentration of C_2H_4 or CO measured by the GC in ppm, $\dot{V}_{CO_2}$ is the CO_2 inlet flowrate (mL min^{-1}) at standard temperature and pressure (STP), P is the standard pressure at 1 atm, R (L atm K^{-1} mol^{-1}) is the universal gas constant and T is the standard temperature at 0 °C. The GC takes a fixed volume of gas (for example, 1 mL) at atmospheric pressure. Since all the terms except the concentration on the right hand side of Eq. (2) are constants, the production rates are proportional to the concentration in ppm.

This experimental setup is automated and digitalized, as explained in detail in Çıtmacı et al. (2022b). Before any digitalization or automation efforts had begun, open-loop steady-state experimental data was obtained under different input parameters. These open-loop steady-state experiments were conducted under a fixed applied potential (V vs. Ag/AgCl) and catalyst rotation speed (RPM) and the setup was operated until the system reached a steady state. During the experiments, gas phase concentrations are measured by GC at 15th, 35th, 55th, and 75th min, and the resulting current (A) and a calculated variable surface potential (V vs. SHE) is measured and recorded each second. The surface potential is the remaining potential across the surface of the catalyst electrode after accounting for the Ohmic drop in the electrolyte due to the solution resistance, and it is the more relevant type of potential parameter, as it affects the charge transfer on the surface of the catalyst electrode. The surface potential (V vs. SHE) is calculated as follows:

$$E_{surface} = E_{applied} - i \times R + E_0 \quad (3)$$

where $E_{surface}$ is the surface potential, $E_{applied}$ is the applied potential measured against the reference electrode, i is the electrical current,

and R is the solution resistance between the working electrode and the reference electrode measured by electrochemical impedance spectroscopy (EIS) (Jang et al., 2022). E_0 is the standard reduction potential of the reference electrode used (Ag/AgCl/1 M KCl). Eq. (3) removes the potential drop in the solution due to the resistance to ion transport in these systems.

2.3. Catalyst deactivation

Catalysts play a central role in electrochemical reactors by promoting the desired product speciation of an electrochemical process. Cu has previously been shown to have the highest selectivity for C_2 products and fuels, making it the catalyst of choice in CO_2 electrolysis (Kim et al., 2014). However, Cu catalysts continuously degrade as reactions proceed, causing a drift in steady state parameters, even when the electrical input (typically applied potential) is held constant. The catalyst deactivation is attributed to surface restructuring, blocking of sites by reactive carbon species, and absorption of impurities from the electrolyte (Nitopi et al., 2019). The deactivation is particularly faster on flat, non-porous catalysts. Electrodes with a higher porosity have a higher density of active sites and can be operated at lower overpotentials, and therefore, the deactivation processes appear to occur slowly over time (Kas et al., 2015).

The deactivation mechanism of Cu under electrochemical CO_2 reduction environments is complicated, and different potential sources of deactivation have been explored. Hori et al. (2005) showed that the metal impurities, mainly Zn^{2+} and Fe^{2+}, and trimethylamine present in a 0.5 M $KHCO_3$ electrolyte, accumulated on the Cu electrode during the tests, affecting product selectivity and current densities. Current density is an output metric for the amount of electrical current that goes into making a reactor product normalized over the active catalyst surface area. These authors recommended the utilization of reagents of the highest purity or the use of pre-electrolysis using platinum black cathodes as a way to purify the electrolyte solution and delay catalyst deactivation (Hori et al., 2005). Another source of deactivation that has been proposed is the formation of graphitic carbon species from the decomposition of reactive intermediates (DeWulf et al., 1989; Shiratsuchi et al., 1993; Xie et al., 2014). Such reaction intermediates could block catalytic sites and poison the electrode surface. On the other hand, Kim et al. (2014, 2018) reported surface reconstruction of polycrystalline copper to Cu(100) facet both in KOH and $KHCO_3$ electrolytes under reductive potentials and associated these surface restructuring to changes in product distribution. Despite the complexity of the deactivation mechanism which can be attributed to various factors, anodic pulsing could be used to mitigate catalyst deactivation and prevent changes in activity and selectivity of Cu under operation (Shiratsuchi et al., 1993; Engelbrecht et al., 2018).

Catalyst reactivation procedures such as anodic pulsing or potential sweeping could be programmed and integrated eventually into the control system of future CO_2 electrolyzers. However, catalyst oxidation procedures can also lead to excessive surface roughening, loss of electric conductivity and catalyst dissolution and must be further investigated. In general, either the energy required for molecules to pass the activation energy barrier increases or the number of active sites decreases as the catalyst continues to deactivate. One way to compensate for the loss of activity is to increase the applied electric potential to ensure that a similar number of reactant molecules can continue to be transformed despite the deactivation. This is accomplished by interpreting general trends of deactivation and introducing the integral of the current parameter. Integration of the current passed in an electrolyzer is a simple, yet effective, way to track the degree of use of a catalyst within an electrochemical system with broad applications beyond CO_2 electrolyzers. Overall, the complex mechanisms behind catalyst deactivation, including poisoning by contaminants and mechanical changes in the catalyst structure, present another critical challenge for developing accurate first-principles models for electrochemical reactors.

3. Machine learning techniques for modeling electrochemical reactors

Modeling electrochemical reactors introduces numerous complexities due to the stochastic and multiscale nature of electrochemical reactions. As a consequence, the development of accurate first-principles models is challenging and requires extensive knowledge and insight about the electrochemical subprocesses discussed in the previous section. The unavailability of electrochemical first-principles models has led researchers to explore alternative modeling methods that require minimal process knowledge. In this section, traditional data-driven and ML methods for modeling of electrochemical reactions are introduced. Hybrid modeling that aims to improve modeling accuracy by considering physical knowledge of the processes is also discussed.

3.1. Data collection and pre-processing

The ML model developed for the RCE reactor in Section 2 is to predict the production rates of various chemicals (e.g., C_2H_4, CH_4, CO, and H_2) based on real-time measurements of applied potential and rotation speed. The first step in developing an ML model is to obtain a high-quality dataset. The data can be collected from open-loop experiments. Specifically, mass flow controllers maintain 20 mL min^{-1} of CO_2 gas flow to the reactor. Before each experiment, the resistance of the buffer solution is measured, and the applied potential (Eq. (3)) is adjusted to compensate for the measured solution resistance. This is done automatically by the potentiostat during the experiment. A steady rotation speed is set for the cylinder electrode so that the hydrodynamics are well developed in the cell, and then the applied potential (V vs. Ag/AgCl) is set to a desired value. Both rotation speed and applied potential are kept constant throughout the experiment. The experiment takes approximately 80 min, and 4 GC measurements are taken during the experiment at the 15th, 35th, 55th and 75th min, respectively. The corresponding real-time current and applied potential values are recorded simultaneously. It is known that the concentration of the gases in the reactor overhead is not equilibrated by the first GC injection, and therefore, this data point is ignored for the purpose of modeling reactor performance. As a result, each experiment produces 3 data points of concentrations for each gas product. The corresponding current densities and surface potentials for relevant GC runs are averaged in a time frame to represent the gas equilibrium in the reactor overhead. The average residence time of this electrochemical reactor is assumed to be around 5–7 min under conditions that operate close to steady state. Therefore, the surface potential and current values are averaged in a 3 min time window, from 5 to 8 min prior to the GC injection to best represent the current density and surface potential corresponding to each GC result. This was the most accurate method of accounting for the residence time of gasses in the reactor, the surface potential control input, and the GC sensor measurement of the control outputs. The experiments with GC measurements below the detection limit or with unusual increases in the electrolyte resistance are marked as outliers. After these data points have been eliminated, the results of 48 experiments with varied thermodynamic conditions and electrolyte concentrations are collected for modeling. Therefore, the resulting database includes 144 data points of surface potential, current, rotation, and gas product concentrations.

3.2. Traditional data-driven modeling approaches

Data-driven methods for modeling electrochemical reactors typically employ statistical or machine learning techniques that use historical data to capture the relationships between input variables (e.g., rotation speed, applied potential, current density) and output variables (e.g., reaction rate, product yield, energy efficiency). These models can be further utilized for systems analysis and control, particularly in processes where deriving first-principles models is challenging or

impractical, offering a balance between simplicity, interpretability, and computational efficiency. Before delving into machine learning modeling techniques, it is useful to first review traditional data-driven approaches that have been widely used across various fields of chemical engineering. These conventional methods laid the foundation for modeling complex systems based on empirical data. They focus on identifying patterns, relationships, and trends directly from the data, without the need for prior knowledge of the underlying physical or mathematical principles.

3.2.1. Linear regression

Linear regression is one of the earliest and most widely used data-driven modeling techniques. It models the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data. A general form of linear regression can be expressed as

$$y = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n + \epsilon \tag{4}$$

where $\alpha_0, \alpha_1, \ldots, \alpha_n$ are coefficients determined by formulating regression problems, $x_0, x_1, \ldots, x_n$ are independent input variables, ϵ is the error term to account for residual variability, and y is the dependent variable to be predicted. The extensions of linear regression, such as polynomial regression, Ridge regression, and Lasso regression, provide better fitting abilities by introducing additional terms that serve different purposes. For example, polynomial regression expands linear regression by including higher-order terms, which allows the model to capture complex, nonlinear trends while maintaining interpretability. Both Ridge and Lasso regression introduce regularization terms to mitigate multicollinearity and prevent overfitting. Ridge regression introduces the L_2 regularization term to penalize the sum of squared coefficients, while Lasso regression adds the L_1 regularization term to enforce sparsity, leading to models with fewer predictors. Linear regression is particularly useful for developing static models for systems with simple input–output relationship without accounting for dynamic changes or temporal dependencies.

3.2.2. Time-series models

Time-series models are tailored for systems where temporal dependencies are prominent. Some examples of traditional data-driven time-series models include autoregressive (AR) model, nonlinear autoregressive exogenous (NARX) model, and moving average (MA) model. In AR model, future values are predicted by regressing the current value on its previous values, which is expressed as follows:

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_n y_{t-n} + \epsilon_t \tag{5}$$

where $\phi_0, \phi_1, \ldots, \phi_n$ are coefficients that can be determined in various ways, such as using Yule–Walker equations, ordinary least squares procedure, maximum likelihood estimation, and Burg's method. $y_{t-1}, y_{t-2}, \ldots, y_{t-n}$ represent past values while y_t and ϵ_t denote the current value and the error term, respectively. NARX model extends the AR model to include nonlinearity and exogenous inputs (i.e., external variables that influence the target variable). A NARX model takes the following mathematical form:

$$y_t = f(y_{t-1}, y_{t-2}, \ldots, y_{t-n}, u_{t-1}, u_{t-2}, \ldots, u_{t-n}) + \epsilon_t \tag{6}$$

where $u_{t-1}, u_{t-2}, \ldots, u_{t-n}$ are past values of exogenous input variables and $f(\cdot)$ is a nonlinear mapping function represented by polynomials, neural networks, or other basis functions. Compared with AR model, MA model utilizes a linear combination of past forecast errors to predict the current value, which is expressed as

$$y_t = \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + \cdots + \phi_n \epsilon_{t-n} \tag{7}$$

where $\epsilon_{t-1}, \epsilon_{t-2}, \ldots, \epsilon_{t-n}$ denote the past forecast errors. Many variants of AR and MA models have been proposed to address specific modeling issues encountered in time-series data. For example, autoregressive integrated moving average (ARIMA) models integrate AR and MA models and incorporate differencing to model non-stationary time-series data.

3.3. Machine learning methods

ML provides another data-driven solution to modeling complex, nonlinear, and high-dimensional systems with improved flexibility and accuracy. ML models learn from training data to generalize patterns and apply them to new, unseen data. The development of an ML model involves data, algorithms, parameters, model training, and model prediction. Data for ML modeling in the context of supervised learning consists of features (independent variables) and labels (dependent variables in supervised learning), and are be used for training, validation, and testing. Decision trees, neural networks, and support vector machines are examples of ML algorithms. The values learned by the ML model during training that define the mapping between inputs and outputs are the parameters of ML models. Model training refers to the process in which the parameters are adjusted using the data to maximize accuracy or minimize error. Once trained, an ML model can make predictions or decisions based on a new data sample.

ML models can be categorized into the following classes based on the learning objective and the type of data available: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. In this article, we focus on supervised learning, where ML models learn from labeled data (i.e., input–output pairs) to recognize patterns and make predictions. Decision trees, random forests, support vector machines, and neural networks are some popular examples of supervised learning-based ML models. Here, we will introduce two types of neural networks for modeling the electrochemical reactor: feed-forward and recurrent.

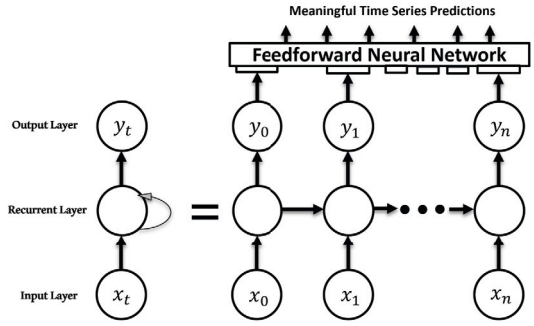
3.3.1. Feed-forward neural network

A Feed-Forward Neural Networks (FNN) is one of the simplest and most fundamental types of artificial neural networks in the field of ML. The term feed-forward indicates that information flows in one direction inside FNN models without loops or cycles. An FNN model consists of an input layer that accepts the features or data points, an output layer that generates the final predictions or outputs of the network, and hidden layers that process the data inputs by applying transformations and nonlinear activation functions. A hidden layer has many neurons or nodes that are connected to all nodes in the previous layer. Each neuron is an operator that applies a weighted sum to its inputs, adds a bias term, and passes the result through an activation function, which is shown as follows:

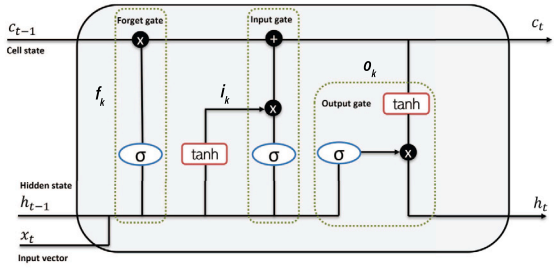
$$y = \sigma\left(\sum (w_i \cdot x_i) + b\right) \tag{8}$$

where x_i is the input from the previous layer, w_i is the weight of the connection, b is the bias, σ denotes the activation function, and y is the output after activation. Note that the number of nodes in the output layer depends on the task. For example, a single node that produces continuous value output is preferred in regression tasks, whereas an output layer with multiple nodes is usually used for classification tasks (e.g., two nodes for binary classification).

FNNs have shown great potential for modeling the steady-state behavior of electrochemical reactors using experimental data. For example, in Luo et al. (2022), the authors developed an FNN model that takes the manipulated variables of the RCE reactor in Section 2 as input to predict the production rates of various hydrocarbon products such as methane, ethylene, methanol, and ethanol, generated by the RCE reactor. Specifically, the hidden layer is densely connected to the output layer and the Softplus activation function, $S(x) = \log(1 + e^x)$, is used in the output layer to ensure nonnegative predictions. The reactor does not consume any of the outlet products, and thus the FNN model cannot predict negative values in the output nodes. Additionally, the Softplus function predicts output variables using a smoother curve which aligns with physically relevant expectations better than other activation function candidates, such as the ReLU or Sigmoid functions. The mean squared error (MSE) function is utilized as a cost function to train the FNN model. For a comprehensive discussion on constructing FNNs, including data generation, processing, design, and training, interested readers may refer to the review paper by Ren et al. (2022).



(a) Rolled and unrolled RNN architectures.



(b) Long short-term memory (LSTM) unit.

Fig. 2. Structures of RNN and LSTM networks, where the LSTM unit in the bottom figure shows the calculation for one hidden state in the entire LSTM network.

3.3.2. Recurrent neural network and long short-term memory

Recurrent Neural Network (RNN) models can be used to model the dynamic behavior of electrochemical reactors. RNNs effectively learn the temporal patterns within time series datasets over a defined time window to detect correlations between sequential data points (Wu et al., 2019b). The model architecture of RNNs is depicted in Fig. 2(a). Unlike FNNs, RNNs process input data sequentially over time, with each input feature undergoing recurrent calculations within the hidden layers. RNNs are designed to capture temporal dependencies by maintaining and updating hidden states that reflect information from previous time steps. The final results are passed to the output layer, which is typically a fully connected dense layer that maps the hidden states to the network outputs.

One of the most powerful RNN architectures for handling time-series data is the Long Short-Term Memory (LSTM) model. Compared to simple RNNs, LSTMs improve the resistance to exploding or vanishing loss function gradients that are commonly exhibited in other neural network models. The forget, input, and output gates in each recurrent unit of LSTMs hold long-term cell state information and selectively update memory cells, which prevents repeated multiplications that can explode or eliminate gradient values. Each gate and LSTM recurrent unit is shown in Fig. 2(b). Here, the cell state is used to transfer useful information and from initial recurrent units to subsequent recurrent units, and thus the memory of the sequence is maintained (Yu et al., 2019). The cell state (Eq. (9c)) keeps all the information from the recurrent unit at the initial time step. More relevant relationships are kept in long-term memory, while less relevant information is removed at each time step (Eq. (9b)). A hidden state (Eq. (9e)) is used to keep the output (Eq. (9d)) from each recurrent unit and transfer it to the following LSTM layer. These hidden state outputs are fed into a dense layer to ensure proper array sizing for the following LSTM recurrent unit hidden state input. At each recurrent LSTM unit, the previous hidden-state vector is combined with the input vector at each new time step. Finally, the combination of vectors is fed into the LSTM gates. The LSTM structure can be represented mathematically as follows:

$$i_k = \sigma(\omega_i^x x_k + \omega_i^h h_{k-1} + b_i) \tag{9a}$$

$$f_k = \sigma(\omega_f^x x_k + \omega_f^h h_{k-1} + b_f) \tag{9b}$$

$$c_k = f_k c_{k-1} + i_k \tanh(\omega_c^x x_k + \omega_c^h h_{k-1} + b_c) \tag{9c}$$

$$o_k = \sigma(\omega_o^x x_k + \omega_o^h h_{k-1} + b_o) \tag{9d}$$

$$h_k = o_k \tanh(c_k) \tag{9e}$$

$$x_k = \omega_y h_k + b_y \tag{9f}$$

where k is the time step, i is the output from the input gate, h is the hidden state, c is the cell state, f is the forget gate, and o is the output gate. Furthermore, w^h and w^x denote the weight matrix to the hidden state vector h and input vector x_k, b_i, b_f, b_c, b_o, b_y represents biases and the subscript y indicates the relationship to the output (Wu et al., 2020).

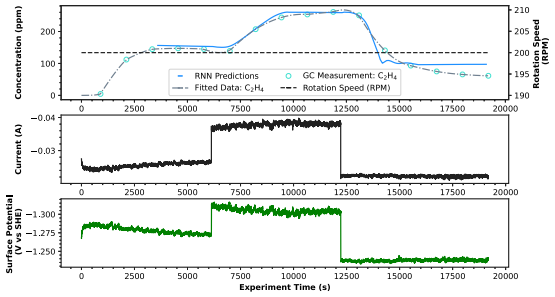
In Çıtmacı et al. (2023), two LSTM models are trained using the data calculated using the best polynomial fits between the GC measurements collected from the RCE reactor. The first LSTM model uses surface potential, rotation speed, and electric current to predict the production rate of C_2H_4 in ppm. For the second LSTM, the inputs are surface potential and rotation speed, while the output is CO production rate in ppm. Based on experimental observations, this RCE reactor has dead times of 600 to 1500 s for step changes on different input parameters. The authors determined that the best time windows for C_2H_4 and CO are 3600 s and 3800 s, respectively. The performance of LSTM models are evaluated with an unseen testing set, and one set of the testing examples is demonstrated in Fig. 3.

To further improve the performance of the developed LSTM models, the authors also explored how regularization techniques affect the modeling accuracy of electrochemical reactors using experimental data. Specifically, the dropout method and L_2 regularization method are used during the training process to prevent data overfitting (Srivastava et al., 2014) and improve the generalization performance of the model (Cortes et al., 2012). The core idea of the dropout method is that when a unit is randomly excluded for an epoch, it prevents the rest of the neurons from excessively co-adapting. In comparison, a L_2 regularization technique is designed to prevent model parameters from growing excessively large (Cortes et al., 2012). The implementation of L_2 regularization in the loss function is represented in Eq. (10).

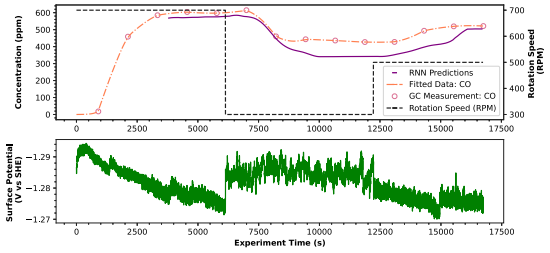
$$\hat{J}(w; X, y) = J(w; X, y) + \frac{1}{2}\lambda w^T w \tag{10}$$

where J is the original loss function (e.g., MSE), and $\hat{J}$ is the modified loss function with an L_2 regularization term. w is the weight matrix that will be optimized, and λ is the weight coefficient for the L_2 regularization term.

Fig. 4 illustrates the improved model predictions using L_2 regularization and dropout. The orange profile is the model trained without any regularization techniques, and predictions with large deviations from the experimental data are observed. Introducing L_2 regularization with $\lambda = 0.04$ provides a slight improvement between the 7500th and 10 000th s; however, the latter stages of the experiment still exhibit significant errors. Increasing the L_2 regularization value to $\lambda = 0.08$ (dashdotted blue curve) reduces noise in the predictions and better captures the trends, but the model overshoots the probable experimental trajectory. Further increasing the regularization value to $\lambda = 0.15$ (dotted blue curve) results in a drift and increased noise in predictions during the later stages of the experiment. This demonstrates that both low and high values of the L_2 regularization parameter λ fail to improve the model adequately, whereas a carefully fine-tuned λ significantly enhances model performance. Finally, the best model with $\lambda = 0.08$ is further improved with an appropriate percentage of recurrent dropout (red curve). This adjustment captures dynamic trends more effectively and greatly reduces prediction errors. These findings highlight that the application of properly tuned regularization parameters, combined with techniques such as recurrent dropout, significantly improves model generalization to unseen data.



(a) C_2H_4 concentration prediction model with surface potential (V vs. SHE), rotation speed, and current as inputs for a time window of 3600 seconds.



(b) CO concentration prediction model with surface potential (V vs. SHE) and rotation speed as inputs for a time window of 3800 seconds. This experiment was conducted under constant applied potential and the change in surface potential is due to the change in current caused by the electrode rotation speed variation.

Fig. 3. RNN predictions of C_2H_4 and CO for open-loop experiments from the testing set.

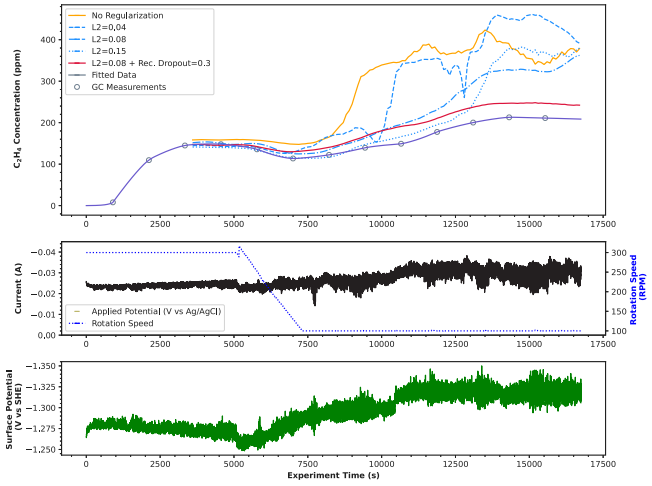


Fig. 4. The effect of regularization on model predictions under varied weight coefficients for the L_2 regularization term. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.4. Maximum likelihood estimation

Although conventional neural networks can effectively model the relationship between input and output variables in complex nonlinear processes, they inherently assume equal importance for all data points. This assumption can result in overfitting, particularly when the dataset contains varying levels of random error from experimental measurements. To address this limitation, the maximum likelihood estimation (MLE) method, originally developed by R. A. Fisher in the 1920s, is incorporated into the FNN model. The MLE method optimizes the parameter set by maximizing the likelihood function of a probabilistic model (Myung, 2003). The likelihood function, $\mathcal{L}(\cdot)$, links an unknown parameter vector (θ) to a set of random variables (z) through their probability density function, $f(z,\theta)$. By maximizing the "likelihood of the sample" represented by $\prod_{i=1}^{n} f(z_i,\theta)$, the MLE method identifies the optimal parameter set, θ^*.

However, the classical MLE approach assumes that all data originate from a single population with a uniform standard deviation. To overcome this limitation, this section proposes a modification for the problem where each set of input parameters corresponds to a distinct population. Each data point, along with its associated standard deviation, is treated as an independent random variable, ensuring a more tailored and robust modeling approach. To apply this method during the training of an FNN model, the experimental dataset is considered to be comprised of pseudo probabilistic samples following a Gaussian distribution. Therefore, the FNN outputs $\hat{y}_{i,j}$ must follow the same distribution as the reference data $y_{i,j}$, which means the joint likelihood of the neural network output also follows a Gaussian distribution and is expressed as follows:

$$\begin{aligned}\mathcal{L}(\mathbf{X};\mathbf{W},\sigma) &= \prod_{k=1}^{d\times m} f_Y(y_k) \\ &= \prod_{k=1}^{d\times m}\left((2\pi\sigma_k^2)^{-0.5}\times\exp\left[-\frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{m}\left|\frac{y_{i,j}-\hat{y}_{i,j}(\mathbf{X},\mathbf{W})}{\sigma_{i,j}}\right|^2\right]\right)\end{aligned} \tag{11}$$

where $\mathbf{X} = [x_1,\ldots,x_n] \in \mathbf{R}^n$ is the input vector of the FNN model, $\mathbf{W}$ denotes the parameter vector that contains all the weights and bias of the FNN model, and σ_i is the standard deviation for each data point. d and m are the number of data points in the training dataset and the number of output states, respectively. Subsequently, we find the optimum weight matrix $\mathbf{W}^*$ by maximizing the logarithm of the joint

likelihood function:

$$\begin{aligned}\mathbf{W}^* &:= \arg\max_{\mathbf{w}} \log \mathcal{L}(\mathbf{X};\mathbf{W},\sigma)\\ &= \arg\max_{\mathbf{w}} \left(-\frac{1}{2}\sum_{k=1}^{d\times m}\log(2\pi\sigma_k^2) - \frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{m}\left|\frac{y_{i,j}-\hat{y}_{i,j}(\mathbf{X},\mathbf{W})}{\sigma_{i,j}}\right|^2 \right)\\ &= \arg\max_{\mathbf{w}} \left(-\sum_{k=1}^{d\times m}\log(2\pi\sigma_k^2) - \sum_{i=1}^{d}\sum_{j=1}^{m}\left|\frac{y_{i,j}-\hat{y}_{i,j}(\mathbf{X},\mathbf{W})}{\sigma_{i,j}}\right|^2 \right) \end{aligned} \tag{12}$$

Since the first term of Eq. (12) is independent of $\mathbf{W}$, the maximum likelihood estimation of this model can be further simplified into Eq. (13).

$$\mathbf{W}^* = \arg\min_{\mathbf{w}} \left(\sum_{i=1}^{d}\sum_{j=1}^{m}\left|\frac{y_{i,j}-\hat{y}_{i,j}(\mathbf{X},\mathbf{W})}{\sigma_{i,j}}\right|^2 \right) \tag{13}$$

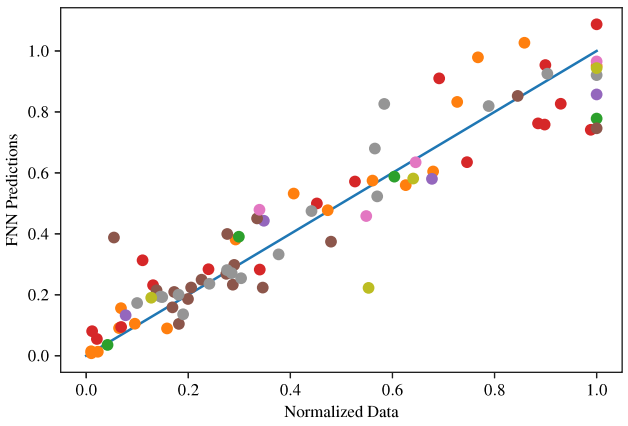
The MLE method can be integrated with ML models to improve learning efficiency. For example, the maximum likelihood estimation FNN model (MLE-FNN) can be constructed using the same architecture and dataset as the standard FNN (Luo et al., 2022). However, the MLE-FNN model considers the standard deviation of each data point in its training process. Specifically, the sample standard deviation is calculated for each data point. The coefficient of variance (v) of each data point then is determined by the ratio of standard deviation and the respective output mean. This normalizes the data variability to allow for unbiased comparison between quantities of different magnitudes. Thus, the weight matrix of the MLE-FNN is optimized to maximize both the accuracy of the prediction and the likelihood function during the training process. The revised mean squared error (MSE) loss function is as follows:

$$\text{loss} = \frac{1}{d}\frac{1}{m}\sum_{i=1}^{d}\sum_{j=1}^{m}\frac{1}{v_{i,j}^2}\left|y_{i,j}-\hat{y}_{i,j}\right|^2. \tag{14}$$

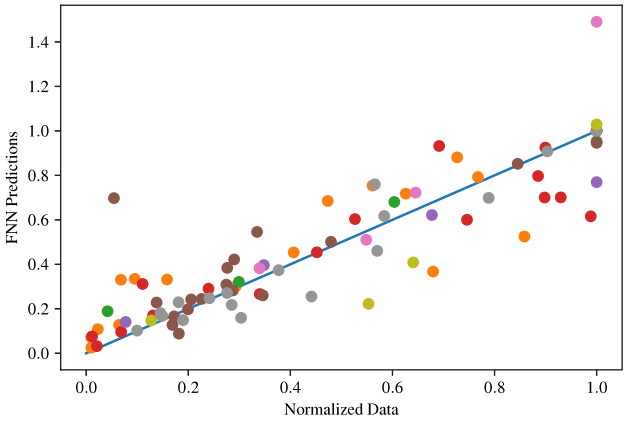
In Luo et al. (2022), a comparison was made between the standard FNN model and the MLE-FNN model on prediction performance. Specifically, to account for the stochastic nature during the neural network training process, a Python script is used to train 100 FNN models in parallel and with randomly partitioned training and validation sets. The best FNN and MLE-FNN are chosen to minimize the MSE for the training dataset. This training method ensures that the selected models are trained consistently, following the same criteria. The selected FNN and MLE-FNN models are then evaluated with respect to the test dataset, using the MSE between the normalized FNN outputs and the normalized testing set. The MSEs for the standard FNN and MLE-FNN are 0.0751 and 0.0791, respectively, which demonstrates a slightly better performance of the standard FNN. Fig. 5 shows that both models give accurate predictions in most data points, but the overall MSE of the MLE-FNN prediction increases, since it ignores the data points with high variance. However, the MSE of the two methods are sufficiently small, which implies that both models capture the input–output relationship well.

To further compare the performance of the two models, the predictions for the CO production rate are compared to labeled outlier points originating from slight drifts in operating conditions, as shown in Fig. 6. The MLE-FNN weighs the data points with critical experimental uncertainty, whereas the standard FNN overfits these data points. This demonstrates the ability of the MLE-FNN model to improve its prediction by accounting for data variance. The goal of the MLE method is to generate models with a higher statistical significance that are suitable to be implemented with an experimental dataset. The MLE-FNN model demonstrates that it can provide an accurate approximation of the experimental data while outperforming the standard FNN in its ability to mitigate the impact of experimental uncertainty.

Remark 1. Bayesian optimization is another widely recognized approach for developing statistical machine learning models. Similar to the MLE method, Bayesian optimization uses the likelihood function to account for data variance. However, instead of focusing solely on the likelihood function, Bayesian optimization centers on the posterior distribution of the machine learning model, as defined by Bayes' rule. This approach incorporates the prior distribution of the parameter vector, $p(\theta)$, and the marginal likelihood of the observed data, $p(D)$, to construct a more robust statistical model. By combining prior knowledge with observed data, Bayesian optimization enables more informed and probabilistic decision-making in the optimization process.



(a) Standard FNN model predictions.

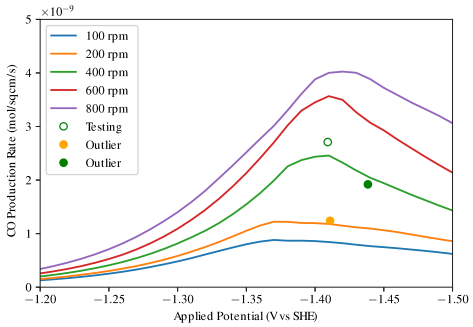


(b) MLE-FNN model predictions.

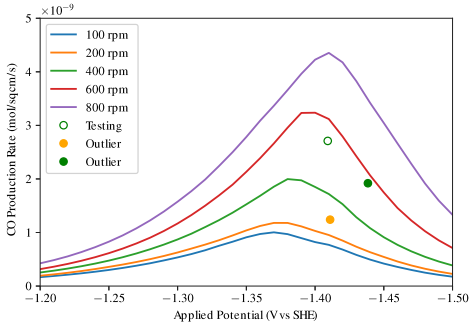
Fig. 5. Comparison between the observed experimental outcome and the neural network predictions from (a) standard FNN and (b) MLE-FNN models.

3.5. Hybrid modeling

Modeling electrochemical reactors often involves a trade-off between accuracy and computational efficiency. Physics-based models, derived from fundamental principles such as mass transfer, reaction kinetics, and thermodynamics, provide a robust framework for understanding the underlying mechanisms of electrochemical processes. However, these models can become complex and computationally intensive, particularly when modeling processes with intricate geometries, coupled phenomena, or time-varying dynamics. Moreover, they may struggle to capture unmodeled dynamics or uncertainties that stem from operational variability or measurement noise. On the other hand, machine learning models provide an alternative by leveraging data-driven techniques to capture complex system behavior with reduced computational demands. Despite their advantages, ML models have notable limitations, including dependence on high-quality training data, difficulty in generalizing beyond the training domain, and a lack of interpretability compared to physics-based approaches. Moreover,



(a) Standard FNN model CO production rate predictions for varied RPM settings.



(b) MLE-FNN model CO production rate predictions for varied RPM settings.

Fig. 6. CO production rate predictions for various applied potentials, in units of V vs. the standard hydrogen electrode (V vs. SHE). The solid points are labeled uncertain data as having a drift in potential. The open point is from the testing set. The CO prediction of standard FNN model overfitted the labeled uncertain data points while the MLE-FNN model successfully learned the experimental uncertainty and provide prediction accordingly; however, this feature introduces additional error to the testing results.

they may fail to respect fundamental physical constraints, potentially leading to unrealistic predictions. To address these limitations, hybrid modeling approaches, which integrate ML models with physics-based models, have emerged as a promising paradigm.

Hybrid models take advantage of the complementary strengths of physics-based and ML approaches to enhance accuracy, generalizability, and computational efficiency. In this framework, physics-based models provide a foundational understanding of the process, ensuring interpretability and adherence to physical laws, while ML models serve to bridge gaps in knowledge, such as unmodeled dynamics or complex nonlinear interactions that are difficult to derive analytically. By combining these approaches, hybrid models achieve higher predictive accuracy than either method alone, particularly in scenarios where data availability or physical understanding is limited. Most importantly, predictions are constrained within the limits of the first-principles or physics-based constraints.

One of the key advantages of hybrid modeling is its ability to incorporate prior knowledge through physics-based constraints. Unlike purely data-driven ML models, which may produce physically inconsistent predictions in the absence of sufficient training data, hybrid models can ensure that predictions remain consistent with established physical laws. Additionally, hybrid models often require less training data than purely data-driven approaches, as the physics-based component reduces the dimensionality of the problem and provides a structured starting point for ML optimization. Hybrid modeling techniques have been successfully applied to complex electrochemical processes where pure knowledge-based or data-based models may perform poorly. For example, a hybrid modeling approach was adopted in Singh et al. (2023) to predict the state of charge and state of health of lithium-ion cells by integrating the partial differential equation of Fick's law of diffusion from a single particle model into the neural network training process. Additionally, a phenomenological mathematical model and a neural network were incorporated in Piuleac et al. (2012) to properly schedule batch electrolysis processes for wastewater treatment. Similarly, in Chen et al. (2023), a two-dimensional mathematical model that consists of a series of governing equations and their corresponding boundary conditions was utilized to constrain the output of a composite neural network in predicting the performance of an all-vanadium redox flow battery.

Partially-Connected Recurrent Neural Networks (PCRNN) are an additional example of hybrid models that function to decouple competing model inputs from the respective outputs of the overall ML-model. Fig. 7 compares the information flow path of the standard RNN model structure to a hybrid, partially-connected model structure (Alhajeri et al., 2022). In the PCRNN, rather than feeding the complete input u to the RNN, u_a and u_b are sent to independent input layers. The first hidden layer of the RNN transforms u_a into x_a, then x_a and u_b are sent to the second hidden layer of the RNN to predict x_b. This hybrid approach allows for an RNN training process based on known physical interactions, or lack of interactions, between input variables on the output states of the process for physically consistent modeling. This method can also be used to decouple measurement noise from training data to develop an RNN that is more robust and generalized in the presence of noisy sensor data.

4. Machine learning-based model predictive control for electrochemical reactors

4.1. Model predictive control

Model Predictive Control (MPC) is an advanced control strategy used in various industrial processes. It uses a process model (e.g., first-principles model or data-driven model) to predict future system states or outputs by solving an optimization problem. MPC minimizes a user-defined cost function and satisfies state or input constraints to determine optimal control actions. The MPC algorithm can be mathematically formulated into the following optimization problem:

$$\mathcal{J} = \min_{u \in S(\Delta)} \int_{t_k}^{t_{k+N_h}} L(\hat{x}(t), u(t))\, \mathrm{d}t \tag{15a}$$

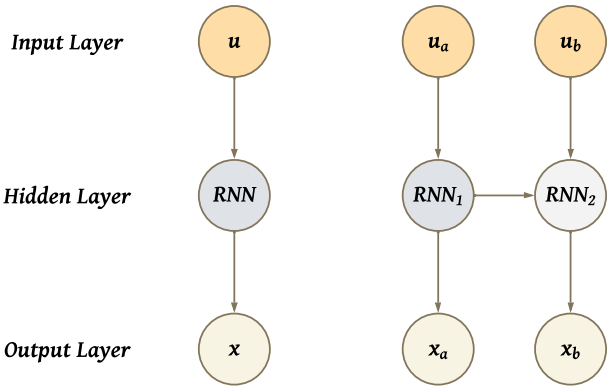


Fig. 7. Standard vs. partially-connected RNN architectures for $u = [u_a, u_b]$ and $x = [x_a, x_b]$.

$$\text{s.t.} \quad \hat{x}(t) = F_{pred}(x(t), u(t)) \tag{15b}$$

$$u(t) \in U, \ \forall\, t \in [t_k, t_{k+N_h}) \tag{15c}$$

$$\hat{x}(t_k) = x(t_k) \tag{15d}$$

$$|u(t_k) - u(t_{k-1})| \leq u_c \tag{15e}$$

where x and u are the process states and control actions, respectively. The set U represents the control action space that defines the upper and lower bounds of all possible control actions applied to the reactor. The change in control actions within two consecutive time steps, that is, $|u(t_k) - u(t_{k-1})|$, is bounded by u_c. The MPC cost function that aims to operate the reactor in steady state is typically designed as $L(\hat{x}(t), u(t)) = (\hat{x}(t) - x_r)^\top Q(\hat{x}(t) - x_r) + (u(t) - u_r)^\top R(u(t) - u_r)$. x_r and u_r are the reference values for the output states and control actions. Q and R represent the weight parameters (both are positive definite matrices) of the penalty terms for the output states and control actions, respectively, in the cost function $L(x, u)$. Therefore, by minimizing the cost function L with an appropriate manipulated input trajectory, the reactor can be driven to the desired setpoint given by x_r by applying the first calculated control action $u(t_k)$ at each sampling time, and then repeating this process in the next sampling time. Finally, F_{pred} is the predictive model, N_h is the prediction horizon, and the set $S(\Delta)$ comprises of piecewise constant functions with a period of Δ.

Compared to traditional control strategies such as proportional–integral–derivative (PID) control, MPC offers several key advantages. First, MPC can explicitly incorporate the state and input constraints into the control problem. For example, manipulated inputs, such as applied potential and rotation speed in electrochemical reactors, are often bounded to ensure safe operations. Second, MPC can effectively control systems with multiple interacting variables (for example, an electrochemical reactor with several input variables that need to be coordinated, including applied potential, rotation speed, reactant input flow rates, and environmental temperature), as it accounts for the interactions among process variables through the prediction model.

MPC can take all of these control variables into consideration simultaneously and find their optimal values by solving real-time optimization problems. Third, MPC can reduce overshoot, oscillations, and settling time based on the predictions of the process models. This predictive capability makes MPC highly effective for controlling electrochemical reactors, as it enables efficient and precise adjustments during the production process. Last but not least, MPC provides a customizable framework where the objective function can be tailored to meet specific performance goals. For example, MPC for electrochemical reactors can be designed to maximize profitability or minimize energy consumption by including terms for production yield or operational cost.

4.2. Machine Learning-Based Model Predictive Control (ML-MPC)

The control performance of an MPC architecture is highly dependent on the quality of the predictive model. However, as explained in previous sections, developing accurate mechanistic models for electrochemical systems is challenging when complex reactions or physics are involved. To address this issue, the adoption of ML models for machine learning-based model predictive control (ML-MPC) has been proposed. In ML-MPC, ML models (e.g., neural networks, Gaussian processes, or support vector machines) can be developed to approximate the system dynamics and will be used to replace the traditional first-principles models. The design of ML-MPC closely follows the formulation of Eq. (15), except that the process model of Eq. (15b) is replaced by an ML model $\hat{x}(t) = F_{ML}(x(t), u(t))$ such as the FNN or RNN model that has been developed in the previous section.

While ML-MPC offers significant advantages over traditional MPC architectures based on first-principles models, it also presents various challenges that must be addressed to ensure the effectiveness and reliability of such controllers. These challenges include data availability, delayed measurements, state estimation, and plant-model mismatch when applying ML-MPC to electrochemical reactors. This section discusses each of these challenges in detail and introduces the corresponding ML-MPC designs developed to address them. Beyond the specific challenges discussed here, it is noted that there exist many other practical issues (e.g., computational efficiency of ML-MPC, curse of dimensionality for large-scale systems, etc.) associated with general ML modeling and predictive control. For a comprehensive review of state-of-the-art solutions to these challenges, interested readers may refer to Wu et al. (2025).

4.2.1. Data availability

The performance of ML models heavily depends on the quality, diversity, and quantity of training data. Insufficient or biased data can lead to ML models with poor generalization and inaccurate predictions. Gathering high-quality, representative data can be expensive and time-consuming. In addition, for systems where the dynamics change over time, maintaining an up-to-date dataset that reflects the actual system dynamics can be challenging.

Data scarcity in electrochemical reactors for ML models can be addressed through several strategies: (1) generation of synthetic data: one way to augment the dataset is to create synthetic data based on physical models in a simulator. Simulations of electrochemical reactors using physical or simplified models can provide a supplementary source of data for model training. These simulations can help the model learn patterns and dynamics when experimental data from an electrochemical process is sparse. Specifically, simulating different operating conditions or disturbances provides data samples to augment the experimental datasets. While this approach generates extensive electrochemical data, it is worth noting that the accuracy of the simulation model significantly impacts the performance of the ML models. Inaccurate simulations can introduce errors that propagate throughout the learning process, leading to suboptimal or unreliable model predictions. (2) transfer learning: when there is limited data for a specific reactor (termed target process) and abundant data exists for similar systems (termed source process), models trained on source processes can bootstrap learning for the target outputs of electrochemical reactors. Transfer learning allows the model to leverage pre-trained knowledge and adapt this information to the specific, data-limited reactor system. (3) Integration of domain knowledge: another approach to handle data scarcity is to incorporate domain knowledge into the modeling process. Physics-informed machine learning or hybrid models may provide an efficient modeling approach that integrates first-principles models or physical constraints (e.g., thermodynamics, kinetics) with experiment data to improve the performance of ML models while reducing the reliance on extensive experiment data.

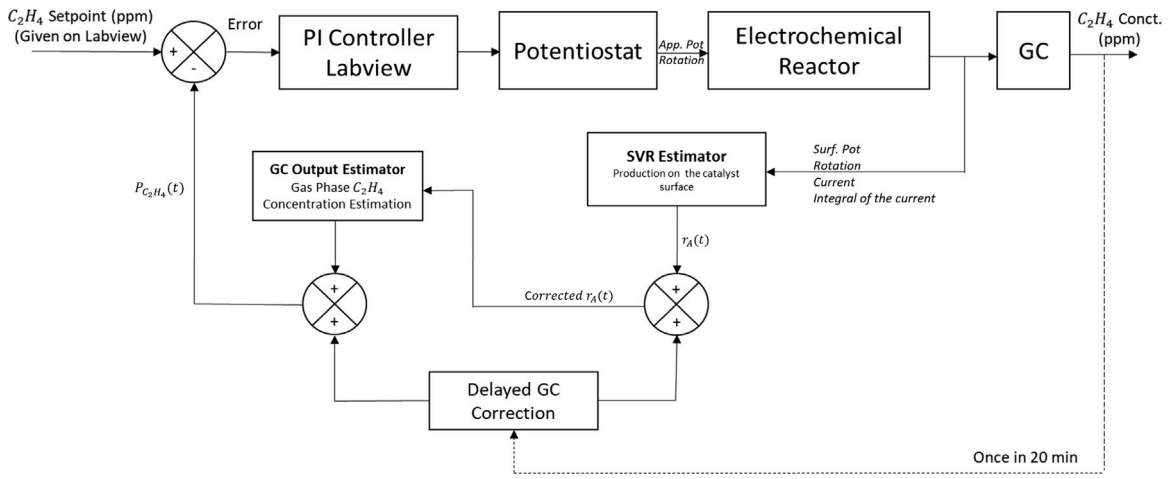


Fig. 8. Process control diagram for a closed-loop controller design that incorporates the SVR model and GC feedback information within the loop.

4.2.2. Delayed measurements

Real-time measurement of process variables is essential to achieve precise control in electrochemical reactors. Although process variables such as electric current, voltage, pressure, and temperature can be measured in near-real-time, critical parameters such as concentration and pH often depend on inferential techniques, which require additional time for analysis. These delays in data acquisition can introduce time lags between sensor measurement and control action implementation, potentially leading to suboptimal, ineffective, or unreliable control responses. The delay introduced by inferential techniques limits real-time control capabilities. However, this challenge can be mitigated to some extent by developing ML-based estimators that can provide real-time insights into process variables, enabling faster and more accurate decision-making. For example, in Çıtmacı et al. (2022a), an ethylene concentration estimator was developed based on an support vector regression (SVR) model for real-time control of the RCE reactor in Section 2. Specifically, the real-time production rates predicted by the SVR model were used as inputs for the gas-phase ethylene concentration model to control the reactor. In addition, GC results obtained every 20 min were used to improve the state estimator and document experimental uncertainties. The closed-loop control diagram with model-based and GC-incorporated configuration is shown in Fig. 8.

4.2.3. State estimation

Obtaining accurate measurements of certain process variables in industrial settings is challenging when it is difficult to detect process parameters, or sensors that detect these process variables come with a high cost. Although some physical parameters are measurable, the time required for analysis often leads to infrequent and delayed data collection. Challenges remain in detecting specific physical quantities and managing infrequent or delayed measurement scenarios. To address these issues, state estimation methods have been employed in both theoretical and practical applications. The state observer (SO) is a commonly used framework for estimating state values (Wang and Gao, 2003). For example, Luenberger developed observer theory and compared it with alternative state estimation methods (Luenberger, 1966). Han (1995) proposed a class of nonlinear extended Luenberger state observers. The Kalman filter is another prevalent state estimation method. Fundamentally, state estimation methods aim to derive all state variables and necessary information for MPC from limited measurement data.

An extended Luenberger observer (ELO) is a type of state observer that estimates state values from limited measurement information. To discuss the ELO algorithm, we consider a general class of nonlinear dynamic systems as follows:

$$\dot{x} = F(x,u) \tag{16a}$$

$$y = h(x) \tag{16b}$$

where x is the state vector, u is the manipulated input vector, and y is the measurement (i.e., system output). $F(x,u)$ and $h(x)$ are nonlinear functions. To implement the ELO method to estimate the state values, $F(x,u)$ is augmented by an observer error term to correct the state values as follows:

$$\dot{\hat{x}} = F(\hat{x},u) + K(y - h(\hat{x})) \tag{17}$$

where $\hat{x}$ represents the estimated state values by ELO, K is the observer gain, and $h(\hat{x})$ is the prediction of the measurable physical information from the state value. The term $y - h(\hat{x})$ represents the error between the physical measurement and the estimated state variables. To tune the observer gain, the error between the estimated and real state values is introduced ($e = x - \hat{x}$). The derivative of this error is calculated by the following equation (Dochain, 2003):

$$\dot{e} = F(\hat{x}+e,u) - F(\hat{x},u) + K(h(\hat{x}-e) - h(\hat{x})) \tag{18}$$

The observer gain (K) should be selected by considering a stability condition for the linearized system shown below:

$$\dot{e} = (A - KH)e \tag{19}$$

where $A = \left.\frac{\partial F(x,u)}{\partial x}\right|_{x=x_s}$ and $H = \left.\frac{\partial h(x,u)}{\partial x}\right|_{x=x_s}$ are the linearization terms of the nonlinear model around the steady-state. To ensure that the estimation error (sufficiently close to the operating steady-state) converges to zero, all eigenvalues of the matrix $A - KH$ should have strictly negative real part.

ELO has been widely used due to its simplicity in controller design and implementation. For instance, in Cui et al. (2024), an ELO scheme is introduced to estimate all state variables of an electrically-heated steam methane reforming (e-SMR) reactor, including the outlet concentrations of all chemical species and the reactor temperature. The ELO state estimator utilizes a system of ordinary differential equations (ODEs) that combines a first-principles model with the error between measured and predicted values. This estimated state is then passed to the predictive controller as the initial state of the optimizer streamline control actions over the prediction horizon. The closed-loop system that incorporates state estimators for an electrically-heated SMR reactor in Cui et al. (2024) under MPC is shown in Fig. 9. Sensor measurements including reactor temperature T_m and H_2 production rate F_{H_2} are inputs to the ELO for state estimation. These two measurable parameters are

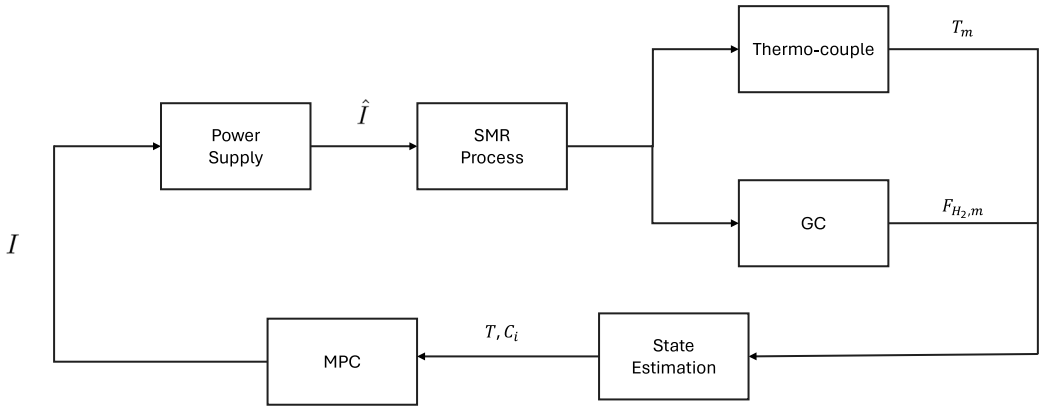


Fig. 9. Closed-loop system structure under estimation-based MPC for the SMR process.

captured by real-time process sensors (K-type thermocouple and Agilent 7890B GC). Subsequently, the MPC algorithm calculates the required control input I and sends an optimal value to the power supply. The power supply applies an electric current $\hat{I}$ to the SMR system, and as a result, the reactor temperature and the H_2 production rate change accordingly.

In Alhajeri et al. (2021), another data-driven estimation approach utilizes a hybrid model that integrates feed-forward neural networks with first-principles models to employ a physics-informed RNN architecture to predict process dynamics. The first-principles model is not able to provide complete state estimation, and the FNN is integrated into the overall model after being trained on a robust dataset. The hybrid FNN model is applied to an MPC feedback architecture for data-driven state estimation to bring the overall process to the desired steady-state setpoints.

4.2.4. Model-plant mismatch

Model-plant mismatch occurs when there are discrepancies between the predictive model used in MPC (e.g., ML models introduced in the previous sections) and the actual dynamics of the system being controlled. This mismatch can arise due to various factors, including simplifications in the model, unmodeled dynamics, parameter uncertainties, or time-varying behaviors of the real system. As the optimization of control actions under MPC relies on the prediction models, any mismatch can significantly affect control performance.

One common issue is parameter uncertainty, where the parameters used in the model differ from those of the actual plant due to estimation errors or temporal variations. Unmodeled dynamics, such as high-frequency behaviors or nonlinearities, can also introduce inaccuracies, particularly when the model is simplified for computational efficiency. Additionally, external disturbances and noise, which are difficult to predict, can cause deviations between the model and the actual state of the plant. Thus, time-varying systems present a unique challenge when plant dynamics change over time and models trained on historical data do not adequately capture the evolved dynamics of the process. Model-plant mismatch leads to suboptimal control actions, constraint violations, or even unsafe operation of chemical processes. Therefore, addressing model-plant mismatch is crucial for maintaining both performance and safety in MPC applications.

Several strategies for mitigating model-plant mismatch in MPC have been investigated in Wang et al. (2024) where an e-SMR process is considered. For the e-SMR process, a key operational disturbance is catalyst deactivation caused by coking and sintering, which leads to a reduced ability of the catalyst to effectively lower the activation energies of the reactions. Consequently, higher activation energies for SMR reactions are considered as a disturbance for the e-SMR process. Three approaches are proposed to address the model-plant mismatch issue, including MPC combined with an integrator, MPC with real-time online retraining of a ML model, and an offset-free MPC scheme.

In the first control scheme, where MPC is combined with an integrator, the control actions from MPC optimization are corrected by adding an additional integral controller, which is written as follows:

$$u_I = \frac{1}{\tau_I'} \int_0^t (F_{\mathrm{H}_2,\mathrm{sp}} - \hat{F}_{\mathrm{H}_2(t_n)}) \mathrm{d}\tau \tag{20a}$$

$$u = u_{MPC} + u_I \tag{20b}$$

$$I = u + I_s \tag{20c}$$

where u is the deviation form of the control action vector (electric current, I) from the newly designed controller, u_{MPC} is the deviation form of the control action vector from the model predictive controller, I_s is the initial steady state value of the electric current, τ_I' is the tuning parameter for the integrator, and $\hat{F}_{\mathrm{H}_2}(t_n)$ is the simulated H_2 production at $t = t_{n,\mathrm{H}_2}$, where t_{n,H_2} is the corresponding time instant for the measured value from the simulated GC involving measurement delay. The integral term of Eq. (20) can compensate for the offset between the target and experimental H_2 production rate by considering the accumulation of errors, using the same function as an integrator in proportional–integral control. In this approach, the initial control action of the integrator (u_I) is tuned to be numerically much smaller than the initial control action obtained from MPC (u_{MPC}), thus showing less impact on overall control action effectiveness compared to the MPC action at the beginning of the closed-loop implementation. Therefore, in the initial stage, the H_2 production rate is driven primarily by MPC actions until the control action u_{MPC} reaches a plateau, indicating alignment with the setpoint reached by the H_2 production rate estimated by the predictive model. After u_{MPC} stabilizes at a constant value, the remaining offset between the H_2 production rate, estimated by the new controller, and the setpoint is eliminated by the integrator term u_I.

Remark 2. Tuning τ_I' in Eq. (20) is challenging for the MPC scheme with an integrator. On the one hand, τ_I' cannot be too small to prevent substantial overshoots in the H_2 production rate. Additionally, a small τ_I' can significantly increase the integrator term, causing step changes in the electric current that exceed the constraint of the electric current changing rate. On the other hand, τ_I' cannot be too large either, as a large value of τ_I' can lead to minimal changes in the electric current, leading to a much longer time to eliminate the offset.

The second approach to address plant-model mismatch is to utilize MPC with online retraining of an ML model. The online retraining method can be utilized to improve the ML model using real-time data. Specifically, in Wang et al. (2024), an RNN model was developed and improved for an e-SMR process using historical data sets. The updated ML model was implemented within the MPC scheme, and therefore, the control action obtained from MPC is more accurate and reduces offset between the ML model and the simulated real process. The components

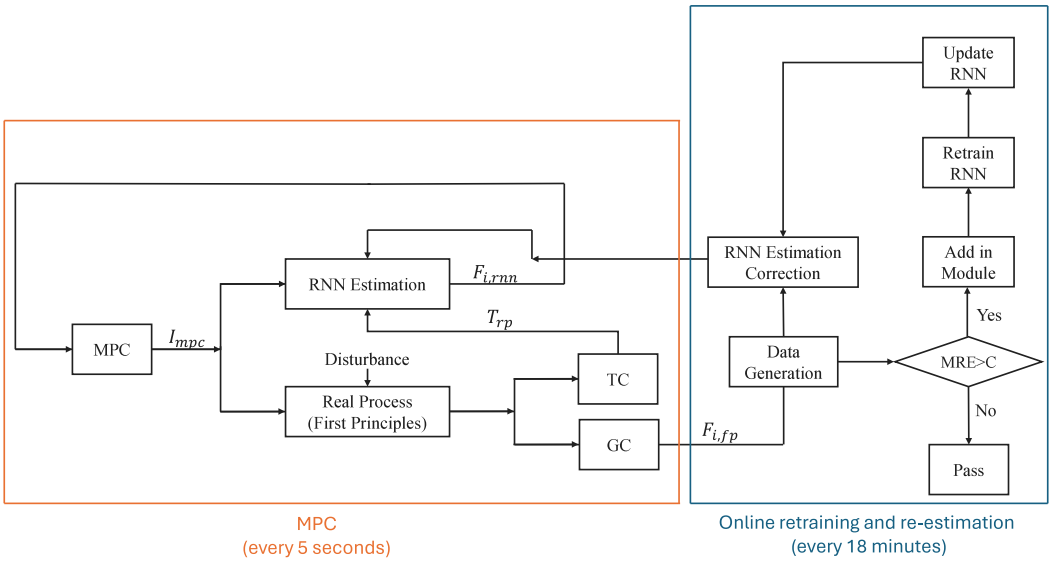


Fig. 10. Setpoint tracking control of the H_2 production rate with an RNN-based model predictive controller using online RNN retraining by utilizing real-time data for the e-SMR process under disturbances.

and process flow of the online RNN retraining-based MPC are shown in Fig. 10. The left orange box represents the MPC of this system every 5 s, while the right blue box signifies the online retraining and re-estimation process that occurs every 18 min when GC data becomes available. Retraining and re-estimation are initiated only upon receiving new GC data.

Another approach to handling model-plant mismatch is to develop an offset-free MPC that regulates the output of the system to a desired reference while simultaneously estimating and compensating for unknown disturbances or offsets. Compared with the traditional MPC scheme, an additional term (θ) is incorporated to augment the system model by tracking the accumulation of the error between the process feedback information and the model-estimated values, which resolves any steady-state errors resulting from model-plant mismatch or process disturbances. Therefore, the model is modified as follows (Maeder et al., 2009; Wallace et al., 2016):

$$\dot{\hat{x}}(t) = F(\hat{x}(t), u(t)) + G_\theta \theta(t) \tag{21a}$$

$$\dot{\theta}(t) = 0 \tag{21b}$$

where θ is the error accumulation term with its corresponding coefficient, G_θ. This augmented model can be further written as follows:

$$\dot{\bar{x}}(t) = \bar{F}(\bar{x}(t), u(t)) \tag{22a}$$

$$\dot{\bar{x}}(t) = \begin{bmatrix} \dot{\hat{x}} \\ \dot{\theta} \end{bmatrix} \tag{22b}$$

$$\bar{F}(\bar{x}(t), u(t)) = \begin{bmatrix} F(\hat{x}(t), u(t)) + G_\theta \theta(t) \\ 0 \end{bmatrix} \tag{22c}$$

This modification subsequently utilizes the updated model in the control scheme. To estimate the current augmented term in real time and improve the estimation of other state variables in the state vector, a Luenberger observer is employed of the form:

$$\dot{\bar{x}}(t) = \bar{F}(\bar{x}(t), u(t)) + K\left[y(t_n) - \bar{y}(t_n)\right] \tag{23a}$$

$$K = \begin{bmatrix} K_y \\ K_\theta \end{bmatrix} \tag{23b}$$

where K is the gain matrix of the Luenberger observer, y is the measured output, $\bar{y}$ is the estimated output, and t_n is the time instant for measurement. A constant error between measurement and estimation is assumed for the interval between two consecutive measurements, and θ is the integral of this error. In this way, the model undergoes continuous correction until no mismatch between measurement and estimation is observed.

5. Integrated data infrastructure platform

The development of an integrated data infrastructure platform is important for implementing machine learning (ML) methods for electrochemical reactors. Such a platform enables data collection, storage, and management of high-quality experimental and operational data, which serve as the foundation for training robust ML models. In this section, we present an integrated platform developed under the Clean Energy Smart Manufacturing Innovation Institute (CESMII) program that focuses on developing scalable, standardized approaches for data collection and contextualization through a Smart Manufacturing Innovation Platform (SMIP). SMIP enables contextualized operational data access via reusable information models, called smart manufacturing (SM) profiles, to facilitate model building, data visualization, and insight. By automating data transmission, leveraging machine learning for real-time estimation and optimization, and using Docker-based reusable application packages, CESMII showcases scalable and efficient SM practices for advanced manufacturing. Specifically, this platform is developed to demonstrate the digitalization and control of the electrochemical CO_2 reduction reactor in Section 2 and online data interfacing with the SMIP.

5.1. Smart Manufacturing Innovation Platform (SMIP) in electrochemical operation research

The Smart Manufacturing Innovation Platform is a standards-based software platform for connection, ingestion and contextualization of data to be used for building applications. SMIP uses standardized information models and ensures the availability of contextualized data from machines and process components for broad application. It is a software infrastructure that integrates the information (IT) and operational technologies (OT) needed for building and deploying data and model applications in operations. Consistent OT and IT integration is facilitated when data are exchanged and models are shared between operations, factories, and companies (Davis et al., 2020; Edgar and Pistikopoulos, 2018). In Çıtmacı et al. (2022b), a SMIP architecture was developed for the control and optimization of the electrochemical reactor that converts CO_2 into valuable chemicals as shown in Section 2, as shown in Fig. 11. Sensors collect data from the reactor through a

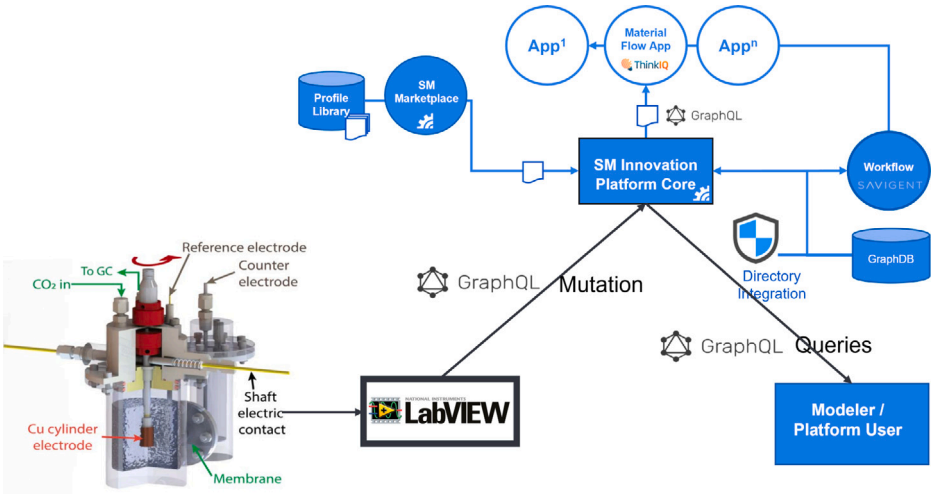


Fig. 11. Smart Manufacturing Innovation Platform (SMIP) architecture.

LabVIEW edge interface. The data is transmitted securely to the SMIP on a per-second basis via GraphQL commands. The coupling of time stamps and data values prohibits overwritten data entries. The time stamp also allows users to select data from particular operating time windows, including entries that occur during the same experimental run. GraphQL API commands can be issued in several programming languages such as Python, JavaScript, Curl, etc.

SM profiles provide a clear understanding of the expected data structure and its content. They can be created with a Profile designer and placed in an SM library. Since the SM Library and the SM Profiles are fully compatible with the SMIP, SM Profiles can be selected and used for specific applications. A key objective is to allow for the standardization and reusability of these information models for similar equipment and operational types. Once an SM Profile is built, consumers, e.g., manufacturers, researchers, etc., it can be reused or extended for new applications, simplifying data collection and modeling. Using object-oriented programming, SM Profiles support varying levels of detail (e.g., equipment, vendor, or service types). Profiles can also reference existing Profiles and are not limited to equipment-focused applications. An example is automated GC analysis, when automated GC analytics are developed and made available for extension to other on-line GC applications in similar or disparate automated systems.

The SM Profile constructed for the electrochemical reduction reactor in Section 2 is shown in Fig. 12. The "CO_2 Reduction Reactor" is the top level profile constructed with the "Gas Chromatograph", "Potentiostat", and "Modeling" as sub-profile. Operational data are collected and transmitted to the SMIP where each raw data item is stored in the context of a corresponding data expectation called an attribute in the profile. Each attribute has the relevant data type information and the associated measurement units. Each data item is also time-stamped. For example, a single temperature measurement from a particular sensor at a certain point in time is collected, ingested, and stored as a number (but with expected units for that particular sensor device defined from the Profile) and time-stamped. Every attribute in a Profile is automatically assigned a tag ID number to define the data storage location. The tag ID and time-stamp are the two required attribute parameters which are needed to store or retrieve the data from the SMIP. Consumers of the data can use GraphQL queries to identify equipment and tag IDs for attributes and access data for any desired time interval.

For reusability, a new user setting up a similar electrochemical CO_2 reactor system can use an existing SM Profile created by previous users. This Profile includes information models for sub-equipment like the potentiostat, GC, and rotation unit. If Fourier Transform Infrared Spectroscopy (FTIR) replaces the GC for gas product measurement, a standard-based FTIR profile would need to be located in the SM Library, extended, or newly developed. The FTIR profile would replace the GC sub-profile, while the other sub-equipment Profiles remain unchanged, avoiding the need to reconstruct models and interfaces for those components. The new FTIR equipment would require a tag ID update for its attributes and adjustments for equipment-specific data, such as vendor name and model number.

5.2. Smart manufacturing in experimental electrochemical reactor setup

Smart Manufacturing principles have been effectively implemented in the setup of an experimental electrochemical reactor in Section 2. The primary components of this Smart Manufacturing setup are shown in Fig. 13. Through automated processes and workflows, the data collected from the reactor is utilized to derive scientific insights and develop operational machine learning (ML) models. Reactor inputs include electrical potential, current, and rotational speed, which are measured by a potentiostat equipped with sensors for electrical current and potential, along with an actuator to tune the applied potential. While liquid product concentrations are measured post-experiment using nuclear magnetic resonance (NMR), gas concentrations are monitored on-line using gas chromatography at 20 min intervals. For real-time control, GC analysis was prioritized to serve as the controlled variable. Automating the gas injection and GC measurement processes was necessary to enable the use of gas product analysis for real-time control applications.

The collected input and output data were processed to establish relationships among the applied potential, electrode rotation speed, temperature, and gas production rates and concentrations for ML model development. However, building models from experimental data presented unique challenges. For example, the limited GC data available per experiment precluded the use of certain ML approaches, such as recurrent neural networks, which require extensive datasets for effective training. Additionally, critical reaction phenomena needed to be captured and reflected in the datasets to ensure accurate modeling. One such phenomenon was the rapid deactivation of the atomically flat catalyst, which altered selectivity and shifted production away from desired products. This challenge was addressed by integrating statistical ML methods with kinetic constants, as detailed in Luo et al. (2022). The cumulative integral of the current was employed to establish a correlation between current and catalyst deactivation, while feature engineering techniques described in Çıtmacı et al. (2022a) improved model performance. These enhancements were vital for implementing a feedback control strategy that optimizes product yields while driving the process toward an energy-efficient setpoint.

CO2 Reduction Reactor	UCLA CHE	Electrochemical reactor in Morales-Guio lab	In Work	8	4
Gas Chromatogram	UCLA CHE / CO2 Reduction Reactor	Sensor for quantifying gas products in real-time	In Work	5	1
C2H4	...CHE / CO2 Reduction Reactor / Gas Chromatogram	Ethylene gas concentration in reactor headspace	In Work	27	
CH4	...CHE / CO2 Reduction Reactor / Gas Chromatogram	Methane gas concentration in reactor headspace	In Work	27	
CO	...CHE / CO2 Reduction Reactor / Gas Chromatogram	Carbon monoxide gas concentration in reactor headsp...	In Work	27	
H2	...CHE / CO2 Reduction Reactor / Gas Chromatogram	Hydrogen gas concentration in reactor headspace	In Work	27	
Mass Flow Controllers	UCLA CHE / CO2 Reduction Reactor	Controls the mass flowrate of CO2 into the reactor	In Work	1	1
Modeling	UCLA CHE / CO2 Reduction Reactor	Relevant modelling data for steady-state and dynamic ...	In Work	9	3
Database	...CHE / CO2 Reduction Reactor / Modeling	Previously processed data used for modelling	In Work	9	
Steady State Database	...CHE / CO2 Reduction Reactor / Modeling	Database for steady-state reactor operation data	In Work	14	
Nuclear Magnetic Resonance	UCLA CHE / CO2 Reduction Reactor	Sesonr for quantifying liquid products offline	In Work		1
Potentiostat	UCLA CHE / CO2 Reduction Reactor	Arranges the potential given to the reactor solution	In Work	8	1

Fig. 12. Hierarchical equipment profile-interface on the SMIP for the SM Profile of the RCE electrochemical reactor.

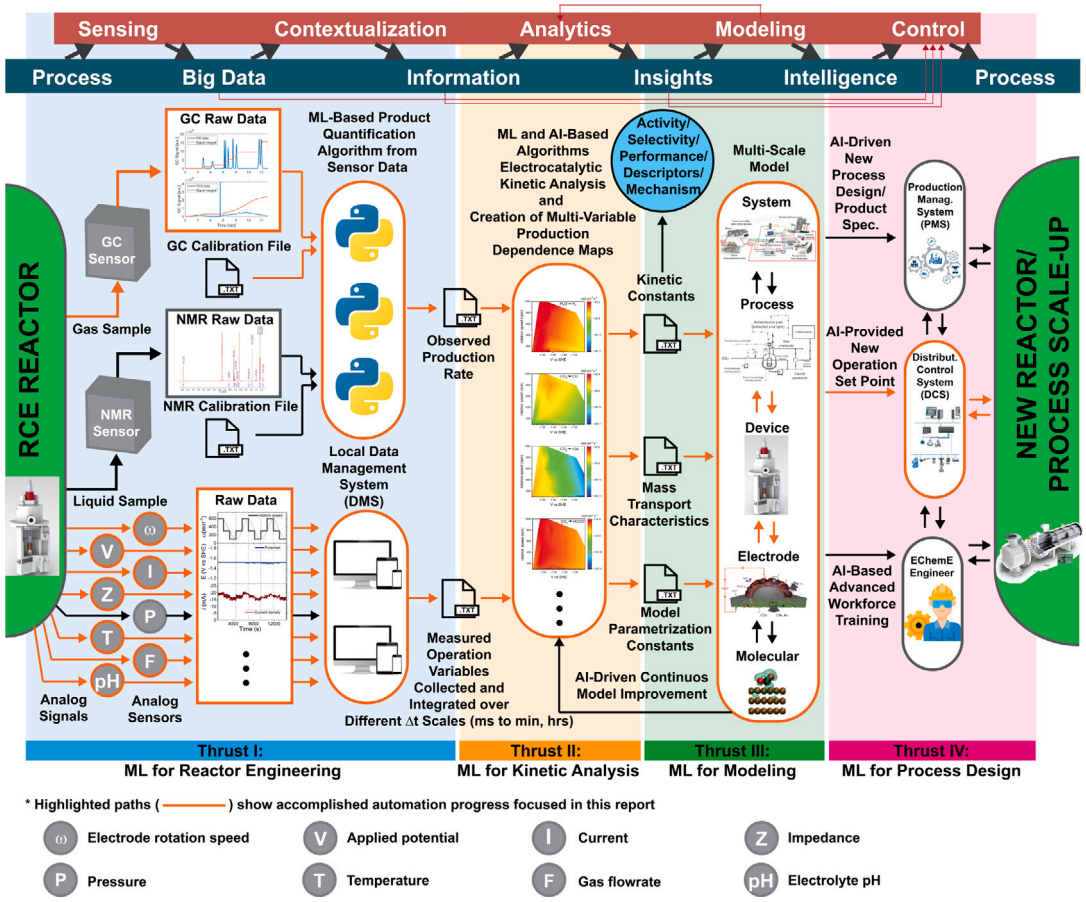


Fig. 13. Data flow and automation strategy for electrochemical scale-up projects.

5.3. Advanced sensors

As process sensors become more affordable and capable of measuring complex process properties, advanced algorithms are increasingly necessary to transform raw sensor data into actionable information for modeling and control. A simple example is the thermocouple, widely used in process industries, where measured voltage is converted into temperature through a well-established algorithm for data contextualization. In contrast, more complex data, such as infrared images from IR cameras in a steam methane reformer (SMR) furnace, require sophisticated algorithms. These cameras, positioned around the furnace, captured radial and axial temperature distributions of tubular reforming reactors (Kumar et al., 2017). However, some regions of the tubes were obscured due to camera positioning and orientation. To address this, an algorithm was developed to convert the infrared images into temperature values and interpolate data for unseen regions, enabling uniform temperature distribution via optimized fuel allocation.

Another example of smart sensing is the automated gas chromatography processing algorithm described in Çıtmacı et al. (2022b). Gas chromatography separates a gas mixture into its components, which are quantified based on detector signals. Gases are separated in a column containing stationary phases, with a carrier gas transporting analytes toward the detectors. The resulting detector signals appear as peaks, where the peak area correlates with the gas concentration. Quantification is achieved by comparing the peak areas to those in a calibration file generated using reference gases. For instance, the concentration of a gas produced during electrolysis can be determined by relating its peak area to known calibration values stores on a local file system. Automating GC analysis for real-time application requires baseline correction to ensure accurate numerical integration of peak areas. Incorrect baselines can lead to misleading results, as proprietary

GC software often generates baselines that fit entire datasets rather than individual peaks. Effective automation requires analyzing each peak's baseline and recalibrating for the components of interest. Manual baseline adjustments are prone to substantial human error, compromising consistency. The automated approach, however, delivers more reliable and reproducible results, facilitating on-line data utilization with greater precision.

6. Challenges and open research questions

In this section, we discuss several open questions in this area, highlighting the opportunities and challenges associated with the implementation of machine learning methods for large-scale electrochemical reactor systems.

6.1. Scaling challenges

Scaling electrochemical reactors from laboratory or pilot setups to industrial production presents significant modeling and engineering challenges. Unlike traditional chemical reactors, electrochemical systems are highly sensitive to operating conditions such as current density, electrode configuration, electrolyte composition, and mass transport dynamics. These factors must be carefully balanced to ensure consistent product quality, safety, and economic feasibility when scaling up. Specifically, translating laboratory-scale reactor designs to industrial reactors requires addressing changes in heat and mass transfer, current distribution, and electrode performance at larger volumes. Process safety should also be taken into account during scale-up, since operating at industrial scales often involves higher pressures, larger current densities, and more reactive intermediates, which requires stringent safety protocols. In addition, a technoeconomic analysis and a life-cycle assessment can be carried out to assess its economic viability and environmental impact. An estimate of operational costs, including electricity consumption, electrode material durability, and maintenance requirement, will provide insight into the scalability of electrochemical processes.

6.2. Selection of input and output variables

In large-scale electrochemical reactors, the increasing number of input and output variables creates additional complexities for process control and optimization. Inputs such as current density, electrode potential, electrolyte flow rates, and reactant concentrations interact in complex ways, affecting key outputs such as product yield, purity, and energy efficiency. The complexity increases as scaling up introduces variability in factors such as mass transfer limitations, non-uniform current distributions, and thermal gradients. However, in large-scale reactors, not all variables affect reactor performance equally. Therefore, identifying the most critical outputs and the relevant inputs is important for efficient operation. Taking into account all process variables can lead to inefficiencies, increased operational costs, and impractical computational burdens during ML model training procedures. Sensitivity analysis and dimensionality reduction techniques can therefore be employed to identify the most influential input and output variables. Feature extraction techniques, such as principal component analysis (PCA) or mutual information analysis, can guide the selection of the most important input features that determine system outputs (Kramer, 1991; Zhao et al., 2023). Once key variables are selected, advanced process control strategies, such as model predictive control, can be designed following the discussions in the previous sections to optimize operations by dynamically adjusting inputs to achieve desired outputs.

6.3. Data collection and model construction

Developing machine learning models for large-scale electrochemical reactors requires vast amounts of data to capture complex interactions between the physical, chemical, and operational parameters of electrochemical systems in dynamic states. In fact, collecting high-quality, representative data is an arduous process for large-scale reactors due to sensor limitations and the high cost of instrumentation, especially at the initial stages of operations. Models trained on small-scale datasets may not directly translate to higher-capacity reactors due to scale-dependent phenomena, such as mass transport effects for different reactor sizes. Various machine learning and data collection methods can be applied to address the problem of data scarcity. For example, systematic sampling techniques, such as Design of Experiments (DoE), can optimize data collection by efficiently covering the parameter space. Active learning frameworks can guide experimental efforts by identifying the most informative data points, further enhancing the robustness of the model with minimal data. Regarding ML models, the development of hybrid models or physics-informed machine learning models could be an effective way to improve the generalizability of ML models while reducing the data requirement (Alhajeri et al., 2022; Sharma and Liu, 2022; Zheng et al., 2023). These models integrate process knowledge with data and use fundamental knowledge of reactor dynamics to guide ML predictions, therefore improving generalizability across scales. Additionally, transfer learning techniques can adapt ML models trained on small-scale datasets to large-scale reactors by fine tuning model hyperparameters (Xiao et al., 2023, 2024; Alhajeri et al., 2024).

6.4. Uncertainty quantification and robustness

Quantifying uncertainty in predictions is critical to ensuring the reliability of ML models in electrochemical systems. In large-scale electrochemical reactors, uncertainty arises from various sources, including unmeasured disturbances, sensor inaccuracies, parameter variability, and incomplete knowledge of the underlying system dynamics. For example, fluctuations in environmental conditions, feedstock quality, or equipment failure can lead to deviations from the dynamics learned by the ML model using historical data. Probabilistic models, such as Bayesian neural networks or Gaussian processes, provide confidence intervals, identifying conditions where predictions are less reliable. Additionally, regularization techniques, such as dropout or robust loss functions, help improve generalization by minimizing sensitivity to noise or outliers in the data (Wu et al., 2021). Furthermore, for electrochemical systems with time-varying dynamics due to external disturbances or model uncertainties, continuous and online learning can be used to update ML models in real-time (Hu et al., 2023). With online learning of ML models, adaptive control systems can be developed to dynamically adjust to changes in system conditions or model inaccuracies by continuously updating model parameters and control actions (Wu et al., 2019a).

6.5. Process operational safety under ML-MPC

The application of ML-MPC in large-scale electrochemical reactors introduces unique safety challenges, particularly when reactors operate under high pressure to improve mass transfer and reaction kinetics. High-pressure conditions can increase the risks of equipment failure, leaks, or hazardous reactions, especially when dealing with reactive intermediates or flammable gases such as hydrogen. Moreover, the nonlinear and complex dynamics of electrochemical systems at high pressures add further complexity, making it difficult to ensure the stability and robustness of ML-MPC algorithms under all operating conditions. Therefore, to improve the safety of control systems for large-scale electrochemical reactors, ML-MPC can be improved by integrating safety constraints (e.g., pressure, temperature, and voltage

thresholds) as hard constraints within the optimization process to ensure that the recommended control actions do not lead to unsafe operating conditions (Albalawi et al., 2018; Wu and Christofides, 2021). Furthermore, existing safety systems can be integrated into the design of ML-MPC in the way that fail-safe mechanisms are introduced into the control loop to provide an additional layer of protection. If the ML-MPC system fails to maintain the process systems within safe thresholds, it can trigger the safety system (e.g., automatic shutdowns, pressure relief systems, or transitions to manual control) to avoid unsafe operations (Zhang et al., 2018).

CRediT authorship contribution statement

Wenlong Wang: Writing – original draft, Methodology, Conceptualization. **Zhe Wu:** Writing – original draft, Supervision, Methodology, Conceptualization. **Dominic Peters:** Writing – original draft, Methodology, Conceptualization. **Berkay Citmaci:** Investigation. **Carlos G. Morales-Guio:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Panagiotis D. Christofides:** Writing – review & editing, Supervision, Project administration, Conceptualization.

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.

Acknowledgments

Financial support from the National Science Foundation and the Department of Energy is gratefully acknowledged. The authors are also grateful to former and current UCLA graduate students Joon Baek Jang, Junwei Luo, Dereck Richard, Vito Canuso and Xiaodong Cui whose work and insights as included in their theses and expressed in group meeting discussions have been partially used to write the present tutorial review paper.

References

Albalawi, F., Durand, H., Christofides, P.D., 2018. Process operational safety via model predictive control: Recent results and future research directions. Comput. Chem. Eng. 114, 171–190.

Alhajeri, M.S., Abdullah, F., Wu, Z., Christofides, P.D., 2022. Physics-informed machine learning modeling for predictive control using noisy data. Chem. Eng. Res. Des. 186, 34–49.

Alhajeri, M.S., Ren, Y.M., Ou, F., Abdullah, F., Christofides, P.D., 2024. Model predictive control of nonlinear processes using transfer learning-based recurrent neural networks. Chem. Eng. Res. Des. 205, 1–12.

Alhajeri, M.S., Wu, Z., Rincon, D., Albalawi, F., Christofides, P.D., 2021. Machine-learning-based state estimation and predictive control of nonlinear processes. Chem. Eng. Res. Des. 167, 268–280.

Catañeda, L.F., Rivera, F.F., Pérez, T., Nava, J.L., 2019. Mathematical modeling and simulation of the reaction environment in electrochemical reactors. Curr. Opin. Electrochem. 16, 75–82.

Chen, W., Fu, Y., Stinis, P., 2023. Physics-informed machine learning of redox flow battery based on a two-dimensional unit cell model. J. Power Sources 584, 233548.

Chen, L., Zhang, X., Chen, A., Yao, S., Hu, X., Zhou, Z., 2022. Targeted design of advanced electrocatalysts by machine learning. Chin. J. Catal. 43 (1), 11–32.

Çıtmacı, B., Luo, J., Jang, J.B., Canuso, V., Richard, D., Ren, Y.M., Morales-Guio, C.G., Christofides, P.D., 2022a. Machine learning-based ethylene concentration estimation, real-time optimization and feedback control of an experimental electrochemical reactor. Chem. Eng. Res. Des. 185, 87–107.

Çıtmacı, B., Luo, J., Jang, J.B., Korambath, P., Morales-Guio, C.G., Davis, J.F., Christofides, P.D., 2022b. Digitalization of an experimental electrochemical reactor via the smart manufacturing innovation platform. Digit. Chem. Eng. 5, 100050.

Çıtmacı, B., Luo, J., Jang, J.B., Morales-Guio, C.G., Christofides, P.D., 2023. Machine learning-based ethylene and carbon monoxide estimation, real-time optimization, and multivariable feedback control of an experimental electrochemical reactor. Chem. Eng. Res. Des. 191, 658–681.

Cortes, C., Mohri, M., Rostamizadeh, A., 2012. L2 regularization for learning kernels. arXiv preprint arXiv:1205.2653.

Cui, X., Çıtmacı, B., Peters, D., Abdullah, F., Wang, Y., Hsu, E., Chheda, P., Morales-Guio, C.G., Christofides, P.D., 2024. Estimation-based model predictive control of an electrically-heated steam methane reforming process. Digit. Chem. Eng. 11, 100153.

Davis, J., Malkani, H., Dyck, J., Korambath, P., Wise, J., 2020. Cyberinfrastructure for the democratization of smart manufacturing. In: Smart Manufacturing. Elsevier, pp. 83–116.

DeWulf, D.W., Jin, T., Bard, A.J., 1989. Electrochemical and surface studies of carbon dioxide reduction to methane and ethylene at copper electrodes in aqueous solutions. J. Electrochem. Soc. 136 (6), 1686.

Ding, R., Chen, J., Chen, Y., Liu, J., Bando, Y., Wang, X., 2024. Unlocking the potential: machine learning applications in electrocatalyst design for electrochemical hydrogen energy transformation. Chem. Soc. Rev..

Ding, R., Zhang, S., Chen, Y., Rui, Z., Hua, K., Wu, Y., Li, X., Duan, X., Wang, X., Li, J., et al., 2022. Application of machine learning in optimizing proton exchange membrane fuel cells: a review. Energy AI 9, 100170.

Dochain, D., 2003. State and parameter estimation in chemical and biochemical processes: a tutorial. J. Process Control 13 (8), 801–818.

Edgar, T.F., Pistikopoulos, E.N., 2018. Smart manufacturing and energy systems. Comput. Chem. Eng. 114, 130–144.

Engelbrecht, A., Uhlig, C., Stark, O., Hämmerle, M., Schmid, G., Magori, E., Wiesner-Fleischer, K., Fleischer, M., Moos, R., 2018. On the electrochemical CO2 reduction at copper sheet electrodes with enhanced long-term stability by pulsed electrolysis. J. Electrochem. Soc. 165 (15), J3059.

Gholizadeh, R., Pavlin, M., Huš, M., Likozar, B., 2024. Multiscale modeling of CO2 electrochemical reduction on copper electrocatalysts: A review of advancements, challenges, and future directions. ChemSusChem e202400898.

Giordano, G.F., Ferreira, L.F., Bezerra, Í.R., Barbosa, J.A., Costa, J.N., Pimentel, G.J., Lima, R.S., 2023. Machine learning toward high-performance electrochemical sensors. Anal. Bioanal. Chem. 415 (18), 3683–3692.

Granados-Fernández, R., Cárdenas-Arenas, A., Montiel, M.A., 2024. Revolutionizing lab-scale electrochemical reactors: Innovative breakthroughs with 3D printing fabrication. ChemElectroChem 11 (15), e202400205.

Guo, W., Sun, Z., Vilsen, S.B., Meng, J., Stroe, D.I., 2022. Review of "grey box" lifetime modeling for lithium-ion battery: Combining physics and data-driven methods. J. Energy Storage 56, 105992.

Han, J., 1995. A class of extended state observers for uncertain systems. Control Decis. 10 (1), 85–88.

Hori, Y., Konishi, H., Futamura, T., Murata, A., Koga, O., Sakurai, H., Oguma, K., 2005. "Deactivation of copper electrode" in electrochemical reduction of CO2. Electrochim. Acta 50 (27), 5354–5369.

Hu, C., Cao, Y., Wu, Z., 2023. Online machine learning modeling and predictive control of nonlinear systems with scheduled mode transitions. AIChE J. 69 (2), e17882.

Iqbal, S., Aftab, K., tul Jannat, F., Baig, M.A., Kalsoom, U., 2024. A bibliographic analysis of optimization of hydrogen production via electrochemical method using machine learning. Fuel 372, 132126.

Jang, J., Rüscher, M., Winzely, M., Morales-Guio, C.G., 2022. Gastight rotating cylinder electrode: Toward decoupling mass transport and intrinsic kinetics in electrocatalysis. AIChE J. 68 (5), e17605.

Ji, S., Zhu, J., Yang, Y., Dos Reis, G., Zhang, Z., 2024. Data-driven battery characterization and prognosis: Recent progress, challenges, and prospects. Small Methods 2301021.

Kas, R., Kortlever, R., Yılmaz, H., Koper, M.T., Mul, G., 2015. Manipulating the hydrocarbon selectivity of copper nanoparticles in CO2 electroreduction by process conditions. ChemElectroChem 2 (3), 354–358.

Kim, Y.-G., Baricuatro, J.H., Javier, A., Gregoire, J.M., Soriaga, M.P., 2014. The evolution of the polycrystalline copper surface, first to Cu (111) and then to Cu (100), at a fixed CO2RR potential: a study by operando EC-STM. Langmuir 30 (50), 15053–15056.

Kim, Y.-G., Baricuatro, J.H., Soriaga, M.P., 2018. Surface reconstruction of polycrystalline Cu electrodes in aqueous KHCO 3 electrolyte at potentials in the early stages of CO 2 reduction. Electrocatalysis 9, 526–530.

Kramer, M.A., 1991. Nonlinear principal component analysis using autoassociative neural networks. AIChE J. 37 (2), 233–243.

Kumar, A., Baldea, M., Edgar, T.F., 2017. A physics-based model for industrial steam-methane reformer optimization with non-uniform temperature field. Comput. Chem. Eng. 105, 224–236.

Li, Z., Zou, S., Wang, Z., Postole, G., Hu, L., Zhao, H., 2024. Machine learning in electrochemical oxidation process: A mini-review. Chin. Chem. Lett. 110526.

Lin, R., Guo, J., Li, X., Patel, P., Seifitokaldani, A., 2020. Electrochemical reactors for CO2 conversion. Catalysts 10 (5), 473.

Liu, X., Fan, K., Huang, X., Ge, J., Liu, Y., Kang, H., 2024. Recent advances in artificial intelligence boosting materials design for electrochemical energy storage. Chem. Eng. J. 151625.

Liu, Y., Guo, B., Zou, X., Li, Y., Shi, S., 2020. Machine learning assisted materials design and discovery for rechargeable batteries. Energy Storage Mater. 31, 434–450.
Luenberger, D., 1966. Observers for multivariable systems. IEEE Trans. Autom. Control 11 (2), 190–197.
Luo, J., Canuso, V., Jang, J.B., Wu, Z., Morales-Guio, C.G., Christofides, P.D., 2022. Machine learning-based operational modeling of an electrochemical reactor: Handling data variability and improving empirical models. Ind. Eng. Chem. Res. 61 (24), 8399–8410.
Maeder, U., Borrelli, F., Morari, M., 2009. Linear offset-free model predictive control. Automatica 45 (10), 2214–2222.
Mehta, R., Gupta, A., 2024. Mathematical modelling of electrochemical, thermal and degradation processes in lithium-ion cells—A comprehensive review. Renew. Sustain. Energy Rev. 192, 114264.
Ming, W., Sun, P., Zhang, Z., Qiu, W., Du, J., Li, X., Zhang, Y., Zhang, G., Liu, K., Wang, Y., et al., 2023. A systematic review of machine learning methods applied to fuel cells in performance evaluation, durability prediction, and application monitoring. Int. J. Hydrog. Energy 48 (13), 5197–5228.
Mistry, A., Franco, A.A., Cooper, S.J., Roberts, S.A., Viswanathan, V., 2021. How machine learning will revolutionize electrochemical sciences. ACS Energy Lett. 6 (4), 1422–1431.
Muddemann, T., Haupt, D., Sievers, M., Kunz, U., 2019. Electrochemical reactors for wastewater treatment. ChemBioEng Rev. 6 (5), 142–156.
Myung, I.J., 2003. Tutorial on maximum likelihood estimation. J. Math. Psych. 47 (1), 90–100.
Nitopi, S., Bertheussen, E., Scott, S.B., Liu, X., Engstfeld, A.K., Horch, S., Seger, B., Stephens, I.E., Chan, K., Hahn, C., et al., 2019. Progress and perspectives of electrochemical CO2 reduction on copper in aqueous electrolyte. Chem. Rev. 119 (12), 7610–7672.
Noël, T., Cao, Y., Laudadio, G., 2019. The fundamentals behind the use of flow reactors in electrochemistry. Acc. Chem. Res. 52 (10), 2858–2869.
Park, J., Lee, J., 2024. Electrochemical energy conversion and storage processes with machine learning. Trends Chem..
Perry, S.C., de León, C.P., Walsh, F.C., 2020. The design, performance and continuing development of electrochemical reactors for clean electrosynthesis. J. Electrochem. Soc. 167 (15), 155525.
Piuleac, C.-G., Sáez, C., Cañizares, P., Curteanu, S., Rodrigo, M.A., 2012. Hybrid model of a wastewater-treatment electrolytic process. Int. J. Electrochem. Sci. 7 (7), 6289–6301.
Ren, Y.M., Alhajeri, M.S., Luo, J., Chen, S., Abdullah, F., Wu, Z., Christofides, P.D., 2022. A tutorial review of neural network modeling approaches for model predictive control. Comput. Chem. Eng. 165, 107956.
Rivera, F.F., Pérez, T., Castañeda, L.F., Nava, J.L., 2021. Mathematical modeling and simulation of electrochemical reactors: A critical review. Chem. Eng. Sci. 239, 116622.
Sharma, N., Liu, Y., 2022. A hybrid science-guided machine learning approach for modeling chemical processes: A review. AIChE J. 68 (5), e17609.
Shiratsuchi, R., Aikoh, Y., Nogami, G., 1993. Pulsed electroreduction of CO 2 on copper electrodes. J. Electrochem. Soc. 140 (12), 3479.
Shirkoohi, M.G., Tyagi, R.D., Vanrolleghem, P.A., Drogui, P., 2022. Artificial intelligence techniques in electrochemical processes for water and wastewater treatment: a review. J. Environ. Heal. Sci. Eng. 20 (2), 1089–1109.
Singh, S., Ebongue, Y.E., Rezaei, S., Birke, K.P., 2023. Hybrid modeling of lithium-ion battery: Physics-informed neural network for battery state estimation. Batteries 9 (6), 301.
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929–1958.
Sun, J., Liu, C., 2023. What and how can machine learning help to decipher mechanisms in molecular electrochemistry? Curr. Opin. Electrochem. 39, 101306.
Tamtaji, M., Gao, H., Hossain, M.D., Galligan, P.R., Wong, H., Liu, Z., Liu, H., Cai, Y., Goddard, W.A., Luo, Z., 2022. Machine learning for design principles for single atom catalysts towards electrochemical reactions. J. Mater. Chem. A 10 (29), 15309–15331.
Wallace, M., Pon Kumar, S.S., Mhaskar, P., 2016. Offset-free model predictive control with explicit performance specification. Ind. Eng. Chem. Res. 55 (4), 995–1003.
Walsh, F.C., de Léon, C.P., 2018. Progress in electrochemical flow reactors for laboratory and pilot scale processing. Electrochim. Acta 280, 121–148.
Wang, Y., Cui, X., Peters, D., Çıtmacı, B., Alnajdi, A., Morales-Guio, C.G., Christofides, P.D., 2024. Machine learning-based predictive control of an electrically-heated steam methane reforming process. Digit. Chem. Eng. 12, 100173.
Wang, W., Gao, Z., 2003. A comparison study of advanced state observer design techniques. In: Proceedings of the 2003 American Control Conference, 2003. Vol. 6, IEEE, pp. 4754–4759.
Wang, Y., Seo, B., Wang, B., Zamel, N., Jiao, K., Adroher, X.C., 2020. Fundamentals, materials, and machine learning of polymer electrolyte membrane fuel cell technology. Energy AI 1, 100014.
Wu, Z., Christofides, P.D., 2021. Process Operational Safety and Cybersecurity. Springer.
Wu, Z., Christofides, P.D., Wu, W., Wang, Y., Abdullah, F., Alnajdi, A., Kadakia, Y., 2025. A tutorial review of machine learning-based model predictive control methods. Rev. Chem. Eng..
Wu, L., Li, T., 2023. Machine learning enabled rational design of atomic catalysts for electrochemical reactions. Mater. Chem. Front. 7 (19), 4445–4459.
Wu, Z., Rincon, D., Christofides, P.D., 2019a. Real-time adaptive machine-learning-based predictive control of nonlinear processes. Ind. Eng. Chem. Res. 59 (6), 2275–2290.
Wu, Z., Rincon, D., Christofides, P.D., 2020. Process structure-based recurrent neural network modeling for model predictive control of nonlinear processes. J. Process Control 89, 74–84.
Wu, Z., Rincon, D., Luo, J., Christofides, P.D., 2021. Machine learning modeling and predictive control of nonlinear processes using noisy data. AIChE J. 67 (4), e17164.
Wu, Z., Tran, A., Rincon, D., Christofides, P.D., 2019b. Machine-learning-based predictive control of nonlinear processes. Part I: Theory. AIChE J. 65 (11), e16729.
Xiao, M., Hu, C., Wu, Z., 2023. Modeling and predictive control of nonlinear processes using transfer learning method. AIChE J. 69 (7), e18076.
Xiao, M., Vellayappan, K., Pravin, P., Gudena, K., Wu, Z., 2024. Optimization-based multi-source transfer learning for modeling of nonlinear processes. Chem. Eng. Sci. 295, 120117.
Xie, J.-F., Huang, Y.-X., Li, W.-W., Song, X.-N., Xiong, L., Yu, H.-Q., 2014. Efficient electrochemical CO2 reduction on a unique chrysanthemum-like Cu nanoflower electrode and direct observation of carbon deposite. Electrochim. Acta 139, 137–144.
Yang, H., He, Z., Zhang, M., Tan, X., Sun, K., Liu, H., Wang, N., Guan, L., Wang, C., Wan, Y., et al., 2023. Reshaping the material research paradigm of electrochemical energy storage and conversion by machine learning. EcoMat 5 (5), e12330.
Yu, Y., Si, X., Hu, C., Zhang, J., 2019. A review of recurrent neural networks: LSTM cells and network architectures. Neural Comput. 31 (7), 1235–1270.
Zhang, Z., Wu, Z., Durand, H., Albalawi, F., Christofides, P.D., 2018. On integration of feedback control and safety systems: Analyzing two chemical process applications. Chem. Eng. Res. Des. 132, 616–626.
Zhao, J., Li, X., Shum, C., McPhee, J., 2021. A review of physics-based and data-driven models for real-time control of polymer electrolyte membrane fuel cells. Energy AI 6, 100114.
Zhao, T., Zheng, Y., Wu, Z., 2023. Feature selection-based machine learning modeling for distributed model predictive control of nonlinear processes. Comput. Chem. Eng. 169, 108074.
Zheng, Y., Hu, C., Wang, X., Wu, Z., 2023. Physics-informed recurrent neural network modeling for predictive control of nonlinear processes. J. Process Control 128, 103005.
Zhou, Y., Ouyang, Y., Zhang, Y., Li, Q., Wang, J., 2023. Machine learning assisted simulations of electrochemical interfaces: recent progress and challenges. J. Phys. Chem. Lett. 14 (9), 2308–2316.