UNIVERSITY OF CALIFORNIA

Los Angeles

Digitalization, Machine Learning Modeling and Control of an Experimental Electrochemical

Reactor

A dissertation submitted in partial satisfaction of the

requirement for the degree Doctor of Philosophy

in Chemical Engineering

by

Berkay Çıtmacı

2024

ABSTRACT OF THE DISSERTATION

Digitalization, Machine Learning Modeling and Control of an Experimental Electrochemical

Reactor

by

Berkay Çıtmacı

Doctor of Philosophy in Chemical Engineering

University of California, Los Angeles, 2024

Professor Panagiotis D. Christofides, Chair

Greenhouse gas emissions from industry and transportation contribute significantly to climate change and its associated adverse impacts on the environment and economy. With the increase in electricity supply from clean energy sources, electrochemical reduction of carbon dioxide (CO_2) has received increasing attention; however, a first-principles model for electrochemical CO_2 reduction has not been fully developed because of the complexity of its reaction mechanism. At this point, data driven methods and machine learning (ML) can be used to model the process. At UCLA, we have constructed an experimental rotating cylinder electrode cell (RCE) setup to develop a deeper understanding of the process, and design scale-up strategies. The experimental

equipment is digitalized on a computer interface using Smart Manufacturing principles, legacy sensors (such as Gas Chromatogram (GC)) are automated, and voluminous steady state and dynamic data sets are generated. Leveraging these datasets and data analytics, machine learning and hybrid models are built using machine learning methods, such as Support Vector Machines (SVR), and artificial neural networks (ANN) and recurrent neural networks (RNN), that are capable of capturing nonlinearities and time dependencies. These models are used to optimize the most profitable setpoints and are used in single-input single-output (SISO) and multi-input multi-output (MIMO) feedback control schemes using multiple proportional integral (PI) controllers and model predictive control. This study proposes approaches for experimental implementation such as incorporating delayed GC feedback into control loops, training dynamic ML models with dead times, data variability and noise, and linearizing the RNN models using Koopman operators for fast real-time optimization.

The electrochemical CO_2 reduction process has potential to use other chemical processes as CO_2 source while sustainably producing other useful chemicals. In addition to the RCE setup, an experimental electrically-heated steam methane reforming (SMR) setup for hydrogen production from natural gas is digitalized, modeled, and controlled. A lumped parameter approach is used for fast calculations, and an extended Luenberger observer is used to compensate for missing and delayed sensor feedback. Finally, a model predictive controller using the estimation scheme is experimentally implemented to control hydrogen flow rates by manipulating the current, showing much faster response compared to PI control.

James F. Davis

Carlos G. Morales-Guio

Tsu-Chin Tsao

Panagiotis D. Christofides, Committee Chair

University of California, Los Angeles

2024

Contents

4 Machine Learning-Based Ethylene and Carbon Monoxide Estimation, Real-Time Optimization, and Multivariable Feedback Control of an Experimental Electrochemical Reactor

List of Figures

xvii

List of Tables

ACKNOWLEDGMENTS

Curriculum Vitae

Education

Istanbul Technical University Sep 2014 - Jan 2019
B.Eng., Chemical Engineering **Istanbul, Turkey**

Publications

1. X. Cui, B. Çıtmacı, D. Peters, F. Abdullah, Y. Wang, E. Hsu, P. Chheda, C. G. Morales-Guio, P. D. Christofides, "Estimation-based model predictive control of an electrically-heated steam methane reforming process", *Digital Chemical Engineering*, 100153, 2024

2. B. Çıtmacı, X. Cui, F. Abdullah, D. Richard, D. Peters, Y. Wang, E. Hsu, P. Chheda, C. G. Morales-Guio, P. D. Christofides, "Model predictive control of an electrically-heated steam methane reformer", *Digital Chemical Engineering*, 10, 100138, 2024

3. D. Richard, J. Jang, B. Çıtmacı, J. Luo, V. Canuso, P. Korambath, O. Morales-Leslie, J. Davis, H. Malkani, P.D. Christofides, C.G. Morales-Guio, "Smart manufacturing inspired approach to research, development, and scale-up of electrified chemical manufacturing systems", *Iscience*, 26 (6), 2023

4. B. Çıtmacı, J. Luo, J. Jang, C.G. Morales-Guio, P.D. Christofides, "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, 2023

5. B. Çıtmacı, J. Luo, J. Jang, C.G. Morales-Guio, P.D. Christofides, "Machine learning-based ethylene and carbon monoxide estimation, real-time optimization, and multivariable feedback control of an experimental electrochemical reactor", *Computer Aided Chemical Engineering* 52, 1519-1524, 2023

6. B. Çıtmacı, J. Luo, J. Jang, P. Korambath, C.G. Morales-Guio, J. Davis, P.D. Christofides, "Digitalization of an experimental electrochemical reactor via the smart manufacturing innovation platform", *Digital Chemical Engineering*, 5, 100050, 2022

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

Chapter 1

Introduction

1.1 Motivation

Global warming emerges as one of the most prominent crises that humanity is being exposed in the recent decades. The increase of the average temperature of the earth is leading to climate change, which threatens agricultural production, melting of polar ice caps, natural catastrophes, and extreme heat waves. The main contributor of the global warming is the greenhouse gases, mainly carbon dioxide (CO_2). Since the First Industrial Revolution in the 18th century, the manufacturing, transportation, and energy sectors have been generating tons of CO_2 emissions. The Paris Agreement [182] accepted in the United Nations Conference in 2015 suggests several key measures to limit the temperature increase below 2 $C°$ above pre-industrial levels and to pursue efforts to limit the temperature increase even further to 1.5 $C°$ until the end of 21st century. This objective would not be reachable if the current manufacturing and transportation methods are sustained without any efforts to limit their emissions. To address this challenge, there has been a growing focus on sustainable and renewable technologies in research.

As CO_2 is the key gas component that causes the aforementioned detrimental effects, carbon capture emerges as an alternative solution. One method for carbon utilization is CO_2 reduction, which is converting CO_2 gas into other useful chemicals through catalytic chemical reactions [100]. However, traditional CO_2 reduction methods require high energy input, which is conventionally obtained through combustion of fossil fuels that results in CO_2 generation. Thus, the conventional CO_2 reduction is not the most efficient way of capturing CO_2. Here, electrochemical CO_2 reduction appears as a clean and sustainable alternative due to its need for electric potential rather than heat, and its availability to work with renewable electricity. The electrochemical CO_2 reduction process can convert CO_2 into many different organic chemicals and fuels, including ethylene, methane (CH_4), ethyl alcohol (C_2H_5OH), acetone (C_3H_6O), and n-propanol (C_3H_8O) when a copper catalyst is used [73]. As electrochemical CO_2 reduction is very favorable yet complicated process, this process is predominantly limited to bench scale. There have been growing efforts to explain the reaction mechanisms, such as the one proposed by [120]. However, the electrocatalytic reaction pathways are very complicated, and at the time of this study, the first principle equations of the CO_2 reduction are not entirely available [73]. Thus, this brings about the opportunity of using data-driven methods for modeling purposes.

As the amount of data has been increasing exponentially in the last decades with the special focus on process digitalization, the availability of data-based models are revisited. The machine learning (ML) methods emerged as an alternative for process representation using the collected data.Among many options, neural networks appeared to be very powerful, particularly in capturing nonlinear dependencies when sufficient quality data is provided. Various machine learning methods were proven to be efficient in steady state and dynamic process behavior modeling, and

even used in process controls [208]. These methods are also used to explain the physical and chemical phenomena of the processes to converge faster to the first-principles information. At this point, ML and neural networks appear to be efficient methods to represent the electrochemical CO_2 reduction behavior based on experimental data.

In order to explore the CO_2 reduction process in deeper detail and conduct research towards the scale-up of the technology, an experimental setup was built including a rotating cylinder electrode cell (RCE) to decouple the effects of reaction kinetics and mass transfer for CO_2 reduction on copper catalysts. This experimental setup is digitalized by connecting multiple sensors and actuators to a common computer interface with the objective of generating voluminous datasets to build ML models.

From a broader perspective, electrochemical CO_2 reduction process requires CO_2, which can be obtained from power plants. However, it is possible to obtain alternative clean fuels like H_2 while producing CO_2 through well-known steam methane reforming (SMR) process, and CO_2 can be reconverted by the electrochemical CO_2 reduction reactors. Steam methane reforming is a highly endothermic reaction which requires very high temperatures between 600-1000 $^\circ C$. In conventional SMR process, the heating is done through burning natural gas [35]. If the SMR process could be heated with a sustainable energy source rather than fossil fuels, the use of CH_4 and CO_2 could be achieved much more efficiently. There have been research endeavors to make SMR more sustainable, such as electrically-heated SMR [200] and proton-conducting membrane reactor [109]. At UCLA, an SMR experimental setup was constructed to analyze these novel technologies, and explore their usability in a cyclic process with electrochemical CO_2 reduction. Our exploration of improved SMR processes starts with the electrically-heated SMR setup. SMR

3

reaction kinetics are well studied since 1989 [215], and has detailed kinetic models. Thus, this process can be modeled by first principle knowledge rather than ML models. However, the nature of complex catalytic reactions, mass and heat transfer phenomena and tubular reactor dynamics make it challenging to build models that can generate fast results. Therefore, existing reaction kinetics data are integrated into a lumped parameter model to efficiently and accurately simulate the setup.

Depending on fluctuating demands for chemicals and energy consumption, these processes can be optimized and made more efficient through effective process controls. Specifically, the advanced controls schemes for both processes are very limited. Thus, proposing optimization and control strategies for these setups holds significant operational values that can influence the scaled-up production facilities in the future, while helping to understand the processes in detail. The developed ML and first principle lumped models are used in advanced control schemes to ensure operational smoothness at most optimized conditions. In the end, this endeavor is anticipated to significantly contribute to addressing global warming by effectively reducing fossil fuel dependency through the fusion of machine learning and electrochemistry.

1.2 Background

The recent milestones in the electrochemical CO_2 reduction research have been yielding promising results towards the scalability of the technology. While some researchers are working on the reaction mechanism for a deeper understanding [43, 91], there are other research endeavors focusing on the catalysts. [47] has introduced a catalyst architecture to decouple electron, ion, and

gas transport, resulting in reduced transfer boundary thickness and increase in cathodic energy efficiency on copper electrode. [98] modified the surface of electrocatalysts with organic compounds to improve energy efficiency towards increasing the selectivity towards C_2H_4. The current research also demonstrated methods to increase selectivity towards the other products, such as n-propanol and acetic acid [191, 32]. Accompanying the technical breakthroughs, a techno-economic analysis [77] suggests that CO and formic acid are the sole profitable products from CO_2 reduction, however, CO_2 reduction would become more profitable if high carbon alcohols, such as ethanol and n-propanol, under the conditions of 300 mA/cm^2 and 0.5 V overpotential at 70% Faradaic efficiency, and a case where the electricity price in brought to 0.03 $\$/kWh$ by 2030. The analysis takes into account that liquid products are easier to store and transport. [57, 56, 199] show that the electricity prices are expected to decrease, specifically as the share of photovoltaics and wind increase in the grid. However, the low selectivity of electrochemical reduction of CO_2 in terms of Faradaic efficiency remains to be a major challenge that requires a deeper understanding of the reaction mechanismn [210]. As the possibility of operational electrochemical CO_2 reduction plants become more realistic, [146] offered a road map to build terawatt scale electrolyzers starting from bench scale reactors.

Through the phase of developing a better understanding of the reaction mechanism, data driven methods appear to be a powerful option. [105] modeled the product gas concentrations of 16 species coming from an experimental electrochemical reactor with a flat copper electrode using artificial neural networks (ANN) and incorporated maximum likelihood estimation method into the cost function of the ANN to account for the data variability. The model showed close correspondence to the available empirical models, and can be operationally used to adjust the product

5

selectivity in the experimental reactor. One other advantage might be using ML models to quickly test proposed reaction mechanisms to accelerate the first-principle development phase. As it is very important to understand the reaction mechanism and selectivity at the steady-state conditions, the dynamics of the process must be well understood before moving to the larger scale productions. For example, [101] improves selectivity through dynamically controlling the surface composition through pulse electrolysis. Thus, with no prior work on implementing feedback control loops on electrochemical CO_2 reduction, there is a great opportunity to examine process dynamics and control to develop strategies for electrochemical CO_2 reduction process. Besides that, an advanced control system can maintain the electrochemical reactor to operate at maximum selectivity.

The bench scale electrochemical CO_2 reduction setup is capable of producing large experimental datasets to be used in data-driven models. Big datasets are proven to be handled more efficiently if smart manufacturing principles are followed, as explained in [31, 30]. The digitalization of the experimental setups can be achieved through LabVIEW, a programming language that is very convenient for connecting sensors and actuators produced by National Instruments. LabVIEW has been widely used for experimental test bed automation and control [9, 170], and it is becoming a handy alternative also for electrochemical process digitalization [118, 178]. [190] proposed a data pipeline to Smart Manufacturing Institute's (CESMII) Innovation Platform (SMIP) through Lab-VIEW, using compatible National Instruments (NI) data acquisition devices (NI-DAQ), Python and Matlab script integration. This approach is expected to digitalize the legacy sensors, build smart sensors using AI or state-of-art data processing methods, and establish secure and ready-for-use data connections for data driven models. Hence, it has great potential to collect data faster and record data securely under efficient monitoring. Thus, a collaboration between Smart Manufactur-

ing and bench scale electrochemical test beds can enable the communication of various equipment in the experimental setup and significantly boost modeling and control efforts, as mentioned in [146].

With the surge in size of the datasets and improvements in computational power through the last decades, the involvement of ML increased in the manufacturing industry. Following this, ML applications started to span a wide spectrum of chemical processes. For example, unsupervised learning methods, such as principal component analysis (PCA) and k-means clustering, are being used in classifying anomalies in processes [158, 106]. On the other hand, supervised learning methods appear to be a better option for process behavior modeling. [72] used support vector regression to predict product compositions with respect to temperature in batch distillation. [124] used tree-based and linear methods to predict salt passage and permeate flow rates in a reverse osmosis membrane. However, as the complexity and nonlinearity of the process increase, more complicated ML methods usually start to perform better as process behavior approximators in the presence of ample data points. For the last three decades, neural networks (NN) have gained popularity for being used to capture complicated data relations [69]. [50] developed a soft-sensor using a feed-forward artificial neural network to provide real-time estimates of polyethylene terephthalate (PET) viscosity, which is crucial for process control. In addition to ANNs, graph neural networks (GNN) are proven to be very efficient in representing molecular structures and can be used in material discovery [142]. However, these methods are usually not taking time dependencies into account, thus might not be very powerful for modeling dynamic systems. At this point, recurrent neural networks (RNN) are introduced for non-Markovian effects [158].

RNNs capture the time dependencies of a dynamic process within a time window, and Long-

Short-Term-Memory (LSTM) architecture improves the RNN performance by solving vanishing and exploding gradient problem in vanilla RNNs [65]. Especially, RNNs in process industries have been extensively applied in advanced process controls, such as model predictive control (MPC) [208]. A model predictive controller requires a process model to predict the future of a process in a horizon, and optimizes the model input with respect to a cost function at each time step for an improved process trajectory. The following examples also demonstrate MPC applications while solving dynamic process modeling challenges. Neural network applications can be combined with available physical information for modeling complicated systems. [205] introduced the co-teaching method, which trains the NN with both smooth and noisy data generated from first-principle-based simulation for a better representation of the process. Physics informed neural networks (PINN) can be used to model partial differential equation (PDE)-based systems such as batch crystallization by incorporating the process equations into the cost function of neural networks [203]. Also, available neural networks can be retrained with new data preserving the inherent dynamics of the model to adapt to new conditions, which is called transfer learning [211]. Furthermore, [2] shows that RNN-based models can be combined with first-principles-based models to build hybrid models, and used along a state estimator, such as a Luenberger observer, for implementing MPC control. However, most of the RNN-based MPC demonstrations are limited to simulations since RNNs are nonlinear black box models and this inherently requires longer times to solve optimization problems that makes it difficult to implement in real-time. Thus, as RNNs would be a great candidate for dynamically modeling complicated electrochemical CO_2 reduction process, the real-time implementation challenges should be addressed by proposing efficient linearized optimization schemes.

In addition to electrochemical CO_2 reduction, modeling and control strategies should be developed for electrically-heated steam methane reformer if both setups are expected to work together in the future. SMR process is challenging to dynamically model since the tubular reactors are modeled using PDEs against time, and heat, mass, and momentum transfer are also involved. [201] investigated the transport properties of electrically-heated SMR. [200] suggests that the heat gradients are more uniform in electrified SMR process, and if heated by renewable electricity, this process has the potential to substantially reduce emissions. SMR reactions are thermal reactions, and conventional SMR process have been well-studied for modeling. Even though the heating source is different, the kinetics remain the same in electrified-heating. The reaction rate equations were proposed by [215]. Using this mechanism, [94, 179] proposed a computational fluid dynamics (CFD)-based modeling strategy along with a computational PI control scheme, and [204] extended this model to an MPC control. However, CFD models require long times to yield results, and thus are not applicable to real-time control. Therefore, the available models can be simplified using lumped-parameter models to be solved in real-time. [110] explores the controllability of lumped-parameter models. Using the smart manufacturing principles, steady state data can be generated to fit parameters to a lumped-parameter model, which can eventually be used to perform advanced control of the electrically-heated SMR setup.

1.3   Dissertation Objectives and Structure

This dissertation presents a data-driven and a lumped parameter first principles modeling method to implement advanced process control for a rotating electrode cylinder cell and joule-

heated steam methane reformer, which employs data analytics, neural network modeling, advanced multi-input multi-output (MIMO) and predictive control, and experimental demonstration of the proposed methods. The objectives of this dissertation are summarized as the following:

1. To digitalize experimental systems on a LabVIEW interface through connecting lab equipment using Smart Manufacturing principles, and securely storing our data in Smart Manufacturing Innovation Platform.

2. To construct machine learning models that effectively capture the input-output dynamics of an electrochemical rotating cylinder electrode cell for CO_2 reduction. These models account for the nonlinear dynamics and dead times to tackle related control challenges.

3. To present online linearization of RNN-based black box models in real-time for fast optimization. The data transfer is made through cloud for efficient experimental implementation.

4. To build simplified models for computationally expensive SMR reactors to implement MPC on an experimental setup, while accounting for missing experimental feedback by Luenberger observer-based estimations.

The chapters of this dissertation are organized as the following:

Chapter 2 presents solutions to the challenges arising in modeling, optimizing, and controlling the electrochemical CO_2 reduction process due to its complex reaction mechanism and the lack of efficient concentration measurement sensors. Gas chromatography (GC), the most common monitoring equipment, provides delayed measurements, and the process undergoes a selectivity shift, posing challenges for conventional control methods. To address these issues, a machine learning-

based modeling approach is developed that integrates support vector regression and first-principles modeling to capture the gas-phase production rates and dynamic behavior of the experimental electrochemical reactor, respectively. This model, along with limited GC measurements, predicts the evolution of gas-phase ethylene (C_2H_4) concentration. A proportional-integral (PI) controller manipulates the applied potential to regulate the gas-phase C_2H_4 concentration at energy-optimal setpoint values calculated by a real-time process optimizer.

Chapter 3 discusses the transformative impact of the exponential growth in data over the past two decades on various industries, including manufacturing. It elaborates on the challenges associated with digitizing and utilizing data collected from advanced sensors and introduces the Clean Energy Smart Manufacturing Innovation Institute (CESMII) and its Smart Manufacturing Innovation Platform (SMIP). The SMIP aims to facilitate efficient data storage, accelerate machine learning model building, and improve data visualization and insight extraction. A case study demonstrating the application of the SMIP in operating an experimental electrochemical CO_2 reduction reactor at UCLA is presented, showcasing real-time sensor data transmission and automated data-based modeling processes. The study utilizes Python scripts for automated data collection and transmission, and LabVIEW for controlling the reactor and monitoring data flow through a single interface.

Chapter 4 presents a methodology to develop a MIMO control scheme for the RCE reactor using techniques from artificial and recurrent neural network modeling, nonlinear optimization, and process controller design. Two products, ethylene and carbon monoxide, are controlled by manipulating applied potential and catalyst rotation speed. The study analyzes process dynamics, designs a feedback control strategy, tunes controllers, and implements the multivariable control system using experimental data through extracting transfer functions from the process data. The results

demonstrate excellent closed-loop performance and regulation of outputs at different set-points, including economically-optimal set-points.

Chapter 5 demonstrates the application of model predictive control using a neural network model in an electrochemical reactor for CO_2 reduction. A long short-term memory network (LSTM) model is developed based on historical experimental data to capture the nonlinear input-output relationship. The Koopman operator method is used to linearize the LSTM model, simplifying and accelerating the optimization step in MPC to be solved in real-time. The performance of the LSTM model, Koopman-based optimization, and MPC is evaluated through simulations and experiments, demonstrating the ability to drive output states to desired setpoints in real-time. Additionally, a transfer learning-based method is employed to update the LSTM model to handle process variability.

Chapter 6 presents a lumped parameter approximation and algebraic equations for gas-phase variables, incorporates reaction parameters derived from experimental data. A first-order dynamic model is employed to capture temperature changes with electric current variations. The resultant dynamic process model is then utilized in a computational model predictive control scheme to regulate the H_2 production process under normal conditions and steam flowrate disturbances. Compared to a classical proportional-integral controller, the proposed MPC scheme exhibits superior performance and robustness in closed-loop operation.

Chapter 7 presents catalyst synthesis, data collection, and thermal considerations for an electrically-heated experimental steam methane reformer. Control strategies are developed for catalyst preservation during temperature ramping using a PI controller. For advanced control strategies like model predictive control, a process model predicts time-evolution based on sensor feedback. Using the

previously developed lumped-parameter model for SMR in Chapter 6, H_2 production is aimed to be controlled by adjusting current flowing through the reactor. However, the MPC requires continuous feedback from all variables, whereas our GC provides discrete measurements with long sampling periods. Hence, the process model is integrated into an extended Luenberger observer (ELO) using reactor temperature and GC data to estimate all MPC variables. The ELO-based MPC system is experimentally implemented, demonstrating faster closed-loop response compared to the PI controller using delayed GC feedback.

Chapter 2

Machine Learning-Based Ethylene Concentration Estimation, Real-Time Optimization and Feedback Control of an Experimental Electrochemical Reactor

2.1 Introduction

Over the last decade, a great deal of interest has been elicited around the idea of using renewable-based electricity rather than fossil fuels as the energy source for large scale manufacturing of chemicals (e.g., [33]). Direct electrocatalytic transformation of carbon dioxide (CO_2) to fuels and chemicals can enable global scale renewable energy storage and close the anthropogenic chemical carbon cycle. However, outside of well-established chloralkali, water electrolysis, and aluminum refining processes, most electrified chemical manufacturing processes are limited to

bench-scale demonstrations. This is particularly the case for processes that use gaseous reactants and produce a complex mixture of multiple different products, such as the electrochemical reduction of CO_2 (e.g., CO_2 electrolyzers are currently limited to electrode areas of around 5 cm^2 or less) or the reduction of nitrogen to ammonia [74, 122]. One of the major challenges of industrially implementing electrochemical reduction of CO_2 is the absence of a well-understood reaction mechanism [75].

To further investigate the fundamentals of electrochemical CO_2 reduction, a gastight rotating cylinder electrode (RCE) cell was recently developed, which can decouple the effects of mass transfer and reaction kinetics as well as promote the production of multiple valuable products [73]. This novel electrochemical reactor shown in Figure 2.1 has demonstrated that mass transport phenomena and intrinsic reaction kinetics can independently affect the productivity and selectivity of the reactor, which implies the potential to control the product distribution of the reactor by manipulating certain inputs. The demonstration of control in production rates during CO_2 electrochemical reduction is rare, with only one example by [28] demonstrating that it is possible to control the selectivity of CO by using light to illuminate the cathode in plasmonic catalyzed electrochemical CO_2 reduction reactions.

The development of chemical process models for control can be approached from two opposite directions, one using first-principles to build up a model and the other using a data-driven approach that condenses the input-output relations in a black-box model. The use of machine learning (ML) models in the context of electrochemical reactors has received attention over the past ten years due to their demonstrated ability to approximate universal linear/non-linear relations with little a priori knowledge of the system [25]. ML has been used broadly, from efforts geared towards
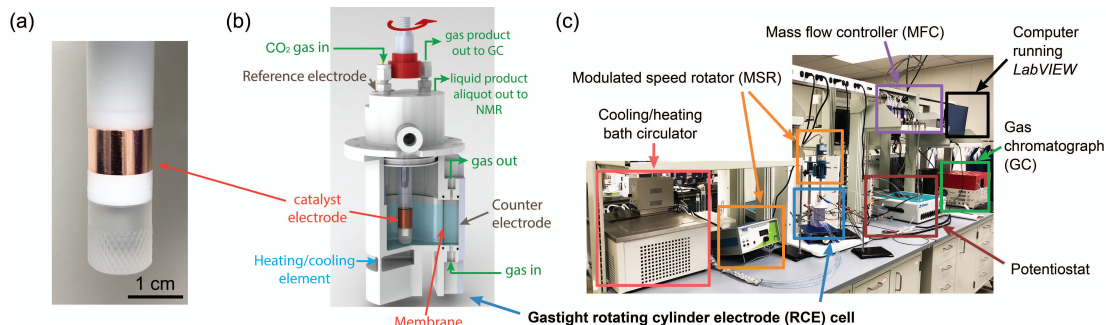
Figure 2.1: Components of the electrochemical reactor. (a) Electropolished atomically-flat poly-crystalline copper cylinder electrode, (b) Schematics of gastight rotating cylinder electrode (RCE) cell, and (c) Bench-scale electrochemical reactor setup with sensors, actuators, and automation devices.

gaining understanding of the role of adsorbates in changing the catalytic properties of copper electrodes to the discovery of CO_2 electrocatalysts based on a large data set of electrolyzer literature. For example, [202] used a four-layer neural network as well as other ML algorithms to investigate the influence of different supplementary adsorbates on the performance of electrochemical CO_2 reduction on copper surface, and reported tens of adsorbates that have a major impact among other candidates. In another work, [108] tested various ML algorithms, such as bagging regressors, regression trees, and gradient boosting, to predict the most feasible electrochemical CO_2 reduction catalyst material using an extensive database comprising academic and industrial reports.

Alternatively, classic ML algorithms usually require less training data than deep learning techniques and are better suited for the analysis of electrochemical data sets acquired under more controlled and better defined conditions of transport, such as those generated in the RCE cell [107]. Among ML algorithms, support vector machines (SVM) provide a complementary way to perform data-driven modelling with lower probability of over fitting [172, 141]. Proposed by [184], support vector regression (SVR), based on SVM, can be used to perform regression for static and dynamic

models by defining a margin around a proposed hyperplane for data fitting and penalizing data points beyond the margin. This method has shown strong performance in representing time-series data and pattern recognition [8, 217, 18]. For example, [127] developed an SVR model to predict the steady-state performance of a reverse osmosis desalination plant using 3990 steady-state data points and further simulated its dynamic perturbations from steady-state operation. [163] demonstrated the ability of SVR to model a dynamic Photo-Fenton process, a photochemical oxidation reaction common in pollutant treatment, and the SVR model effectively predicted the production of OH radicals with data from a pilot plant. [172] modeled NO_X emissions from coal combustion boilers based on observations of 73 hours of experiment using SVR and Artificial Neural Network (ANN), which showed that SVR is a more robust option than ANN for this process.

Motivated by the above considerations, this work proposes an ML-based scheme to implement real-time optimization (RTO) and feedback control in an experimental electrochemical reactor for CO_2 reduction. Specifically, an SVR model is developed on the basis of existing experimental data to estimate the dynamic response of the reactor operation, and to account for inherent disturbances such as sensor uncertainties and catalyst degradation. Subsequently, the information from the sensors and ML model is integrated and used by a Proportional-Integral (PI) controller that manipulates the input of the reactor. In addition, a real-time optimizer (RTO) is developed to compute the optimum setpoint for the reactor by integrating the steady-state prediction from a neural network model and valid market information. The proposed control and optimization scheme is demonstrated by a series of experiments that control ethylene production of the RCE reactor.

The rest of this chapter is organized as follows. In the section entitled "Preliminaries", the experimental reactor setup and catalyst deactivation are described. In the next section, entitled

17

"Machine-Learning Modeling", the experimental method of extracting, formulating and building the SVR model and related model improvements are discussed. In the following section entitled "Real-Time Optimization", the methodology to calculate the most economically feasible setpoints is discussed. In the section entitled "Feedback Control", the implementation of a PI controller is integrated with an estimator for the reactor overhead ethylene concentration and real-time GC measurements; then the performance of the model and the controller is evaluated.

2.2   Preliminaries

This section introduces the background of the experimental electrochemical reactor employed in this work. Specifically, the experimental system and methodology are discussed and important variables are explained. An overview of the catalyst deactivation is then used to explain the background of the control objective. All process equipment mentioned in this section, except nuclear magnetic resonance (NMR), is connected to a Laboratory Virtual Instrument Engineering Workbench (LabVIEW) interface. The operational procedures required in the experiment, including but not limited to real-time GC data processing, controller activation, and equipment actuation, are fully automated by a computer program developed with Python and implemented through a LabVIEW interface. Figure 2.1 shows UCLA's RCE system along with the array of sensors and actuators in the system.

The ML model is integrated into LabVIEW and the outputs of the ML model are calculated on a per-second basis. The parameters of the feedback controllers are assigned a priori to take specific values for specific time intervals of operation. This includes closed-loop control experiments

18

where the setpoint is changed between two time intervals. For gas product quantification, a Python script was developed to automatically process the raw GC data, detect the gas product signals, and calculate the corresponding gas phase concentrations utilizing a GC calibration file. This GC script is then used to get real-time data to correct the ML model and improve the controller performance. The process data is synchronized with an online database of the Smart Manufacturing Innovation Platform (*SMIP*) provided by The Smart Manufacturing Institute (*CESMII*). *SMIP* was used for data storage and monitoring and to facilitate data accessibility for all researchers when building models on site.

2.2.1 Electrochemical Reactor Setup

We have recently described the construction of an RCE reactor setup [73] which can be used to produce over 16 different gas and liquid products during the electrochemical reduction of CO_2 on a copper electrode. As shown in Figure 2.1, this electrochemical reactor setup consists of six major components. These are: the reactor with its two chambers containing respectively the working (cathode) and counter (anode) electrodes separated by an ion-exchange membrane that prevents product cross-over between the two chambers, a potentiostat that regulates the potential applied to the working electrode, a mass flow controller to adjust the mass flow rate of the feed gas (CO_2), a modulated speed rotator (MSR) to adjust the rotation speed of the electrode, a cooling/heating block to control the temperature of the cathode compartment, and a computer. In addition to these six major components, a gas chromatograph (GC) and a nuclear magnetic resonance (NMR) spectrometer are utilized to quantify the chemical species in the gas and liquid-phase products, respectively. In the work presented here, the GC has been automated for on-line gas product

quantification.

During the experiment, inlet flow rate of feed gas is fixed to 20 $mL \cdot min^{-1}$ by the mass flow controller and is bubbled directly through a 0.2 M KHCO$_3$ electrolyte solution in the cathode chamber at room temperature and 1 bar. Subsequently, CO$_2$ is reduced to synthetic fuel and chemical products which result from proton-electron transfer processes occurring on the surface of the rotating cylinder electrode, which serves as the cathode and working electrode. In addition, the potential applied to the working electrode is continuously measured against the reference electrode, while the MSR modulates the rotation speed of the RCE through magnetic coupling and the reactor temperature is controlled through a bath circulator that flows coolant at a specified temperature through the cooling/heating block. Eventually, gas products leave the reactor headspace and travel through polytetrafluoroethylene (PTFE) tubing to be quantified by the GC once every sampling period. The liquid products accumulate in the electrolyte solution and are then quantified by NMR after electrolysis [73]. Since the concentrations of the liquid products cannot be measured while the reactor is operating (NMR sample preparation, analysis and quantification take long and are infeasible to implement feedback control on this data), only the gas products are considered as the outputs of the process for the proposed control scheme.

Remark 1 *In addition to working as an actuator, the potentiostat also functions as a sensor that measures and records the current density flowing between the working electrode and the counter electrode in real-time. Based on this measurement, the potential on the surface of the electrode, which is known to directly affect the electrochemical reaction, can be calculated with reference to*

the standard hydrogen electrode (SHE) by Eq. 2.1.

$$E_{surface} = E_{applied} - i \times R + E_0 \qquad (2.1)$$

where $E_{surface}$ is the surface potential, $E_{applied}$ is the applied potential measured against the reference electrode, i is the current density (negative value for reductive currents), and R is the resistance of the solution between the working electrode and the reference electrode measured using electrochemical impedance spectroscopy (EIS) [73]. E_0 represents the standard electrode potential of the reference electrode (Ag / AgCl / 1 M KCl), so that $E_{surface}$ is referenced against the standard hydrogen electrode (V vs. SHE) after correction.

2.2.2   Identification and Quantification of the Process Output

Among all products generated by this electrochemical reactor, 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:

$$2CO_2 + 8H_2O + 12e^- \rightarrow C_2H_4 + 12OH^- \qquad (2.2a)$$

$$CO_2 + 6H_2O + 8e^- \rightarrow CH_4 + 8OH^- \qquad (2.2b)$$

$$CO_2 + H_2O + 2e^- \rightarrow CO + 2OH^- \qquad (2.2c)$$

$$2H_2O + 2e^- \rightarrow H_2 + 2OH^- \qquad (2.2d)$$
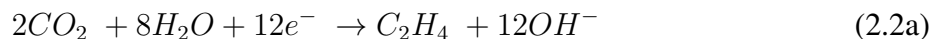
The measured production of gas products (C_2H_4, CH_4, CO, and H_2) by the GC is the target component of the feedback control scheme, which is implemented by the following procedure. Firstly, the temperature-programmed GC separates molecules injected onto the columns, which have different elution times. Subsequently, separated molecules are detected using a thermal conductivity detector (TCD) and a flame ionization detector (FID) in the GC system. In this study, the temperature program runs for 14.33 minutes, after which the heated column oven requires 6 minutes of cooling time before the next injection. This means that the GC has a sampling period of 20.33 minutes. The delay in the GC analysis limits the possibilities for real-time control, but this can be overcome to some extent through the development of a ML-based estimator which allows for the calculation of real-time gas compositions in the cell overhead. This estimator is described in Section 2.3.

Remark 2 *Because of the strong correlation of the ethylene productivity with respect to the surface potential and the electric current, it is chosen as the initial component of interest for modeling and control. In addition, ethylene has numerous industrial applications and is the most produced organic molecule worldwide, with a current installed production capacity of over 200 million metric tons per year.*

2.2.3 Catalyst Deactivation

Catalysts play a central role in electrochemical reactors. Our experimental process uses a smooth, cylindrical, polycrystalline copper electrode shown in Figure 2.4a. Copper has previously been shown to have the highest selectivity for C_2 products, making it the catalyst of choice in CO_2 electrolysis [84]. However, the copper catalyst continuously degrades as the reactions proceed,

and this causes a drift in steady-state, even when the input (typically applied potential which is the driving force for electrochemical reactions) 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 [121]. The deactivation is particularly fast on flat, non-porous catalysts such as the one used here. Electrodes with a higher porosity have a higher density of active sites and can be operated at lower overpotentials, so the deactivation processes appear to occur over longer time periods [80].

The deactivation mechanism of copper under electrochemical CO_2 reduction environments is complicated, and different potential sources of deactivation have been pointed out over the last decades. [67] showed that metal impurities, mainly Zn^{+2} and Fe^{+2}, and trimethylamine existing in a 0.5 M $KHCO_3$ electrolyte accumulated on the copper electrode during testing, affecting product selectivity and current densities. 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 [67]. Another source of deactivation that has been proposed is formation of graphitic carbon species from decomposition of reactive intermediates [34, 162, 212]. Such reaction intermediates could block catalytic sites and poison the electrode surface. On the other hand, Kim et al. observed surface reconstruction of polycrystalline copper to Cu(100) facet both in KOH [84] and $KHCO_3$ [85] 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 copper under operation [162, 41].

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 this work, we have introduced an additional parameter of the cumulative integral of current to address the deactivation issue. As a general trend, 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 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 and will be discussed in detail in Section 2.3.4.

Remark 3 *A different catalyst morphology may considerably delay the deactivation. Specifically, for electrochemical CO_2 reduction reactions, another available catalyst morphology is copper cubes that have pores on the electrode surface, allowing both internal and external mass transfer to play a role [149]. However, this study is only focused on the reactions with flat copper and compensation of the catalyst deactivation via introduction of the cumulative integral of current parameter and feedback control. Future work will investigate control with the copper cube catalyst.*

2.3　Machine Learning Modeling

One of the main objectives of this research is to control the experimental electrochemical reactor. In the absence of a first-principles model for the electrochemical system, a simple PI controller could be implemented and tuned on the basis of direct measurement feedback without the use of any model. However, this scenario exhibits many limitations in terms of control performance. To construct a closed-loop control system, the frequency of feedback from GC measurements (20.33 minutes) would be inadequate to provide accurate and reliable control of the reactor. An effective PI control in the presence of catalyst deactivation would benefit from feedback on a per-second basis rather than once every 20.33 minutes from the GC. The use of a Fourier Transform Infrared Spectrometer (FTIR), a device that can detect the concentration of gas products every second with high reliability, was considered as an alternative method of measuring and providing feedback to the control system [81]. However, because of the higher volume flow rate required for FTIR, the product concentrations from the reactor would have been reduced by an order of magnitude. This reduction in concentration would have placed the product concentrations near the limit of detection, making FTIR inadequate for the reactor used in this investigation. Due to this limitation, building a dynamic machine-learning model is deemed to be a necessary step in controlling the electrochemical system.

2.3.1　Data Collection and Pre-processing

The data used to develop the ML model is collected from open-loop experiments performed by the following procedure. Mass flow controllers are arranged to maintain $20\ mL \cdot min^{-1}$ of CO_2

gas flow to the reactor. Specifically, before each experiment, the resistance of the buffer solution is measured, and the applied potential 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 minutes, and 4 GC measurements are taken during the experiment at 15, 35, 55 and 75 minute mark. The corresponding real-time current and applied potential values are recorded simultaneously. It is known that concentration of the gases in the reactor overhead is not equilibrated by the first GC injection, so this data point is ignored for the purposes 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. It is assumed that the average residence time of this electrochemical reactor is around 5–8 minutes under conditions operating close to steady-state. Therefore, the surface potential and current values are averaged in a 3-minute time window, from 8 to 5 minutes prior to the GC injection, to best represent the current density and surface potential corresponding to each GC result.

The experiments with GC measurements below the detection limit or with unusual increases in the electrolyte resistance were marked as outliers. After these data points have been eliminated, the results of 48 experiments are considered for modeling. The resulting database includes 144 data points of surface potential, current, rotation, and gas product concentrations.

Remark 4 *Using small-scale experimental data to develop an ML model raises additional unique challenges in comparison to using simulation or well-structured industrial data, since it contains more experimental uncertainty. The measured resistance of the $KHCO_3$ solution can vary for each trial depending on the preparation of stock solution, temperature, conductivity of the inner electrical circuit, etc., although the values were kept as consistent as possible ($7 \pm 0.2\ \Omega$). Similarly, the electrodes are both mechanically and electrochemically polished prior to each experiment, but even the standardized polishing process can lead to different catalyst activities at low overpotentials. Furthermore, data measurements are limited by the detection range of the sensors, since the production scale is small with this experimental setup. As a result, it is possible to obtain different observations from the same input conditions. Thus, averages and standard deviations of some of the data from open-loop experiments, presented in Table 2.1, are utilized in the ML modeling.*

Table 2.1: Variations in surface potential, current, and C_2H_4 concentrations in repeated open-loop experiments at certain operating conditions.

Average		
Potential	**Total Current Density**	**C_2H_4 Concentration**
($V\ vs\ SHE$)	(mA/cm^2)	(ppm)
-1.447	11.25	140.61
-1.426	9.6	150.71
-1.407	6.09	139.01
-1.365	3.87	47.81
-1.318	1.7	9.62
-1.262	1.19	1.28

(a) Averages of experimental inputs and outputs in various ranges.

Standard Deviation		
Potential	**Total Current Density**	**C_2H_4 Concentration**
($V\ vs\ SHE$)	(mA/cm^2)	(ppm)
0.010	2.360	64.38
0.001	1.687	42.40
0.005	0.960	23.397
0.013	1.059	13.93
0.010	0.214	2.64
0.003	0.082	1.82

(b) Standard deviations of experimental inputs and outputs in various ranges.

2.3.2 Model Selection and Evaluation

[105] have modeled the reactor setup for the estimation of the steady-state production rate using an ANN. However, this ANN model only gives the average estimation of what the reactor would produce throughout the 80-minute experiment and cannot be used for dynamic data modelling. A very effective method of constructing a dynamic data driven model involves using Recurrent Neural Networks (RNN) as this architecture accounts for the history of the process. [208] examined in detail the use of RNNs in process control. One of the dynamic ML modeling techniques is Long Term Short Term (LSTM) RNNs. LSTM networks have an underlying architecture that processes a time window and keeps the necessary information at specific time steps. Therefore, the use of LSTM to store these dependencies is justified. However, neural network architectures are very data-dependent, and 3 data points with 20-minute intervals per experiment is insufficient to capture the general behavior of an 80-minute experiment. Another example is using ML techniques for parameter-based modeling (such as reaction rate at specific temperatures) and combining this ML model with first-principle time-dependent equations.

Based on the work of [105], the applied potential and rotation speed are two important inputs for characterizing the reactor operation. However, these two inputs are held constant throughout the experiment. To create a dynamic state estimator, the model must be trained with inputs that vary during the experiment. Thus, the state estimator model should also incorporate other inputs, such as the current and surface potential, which vary throughout the experiment with the catalyst degradation.

The available data is separated into training and testing sets. 80 % of the data is destined

for training and used to train various machine learning methods, including linear, ridge, lasso, polynomial regressions, support vector regression, and gradient-boosted decision tree algorithms. Five-fold cross-validation (CV) and mean square error (MSE) were used for model selection. The MSE of the different models are plotted against each other in Figure 2.2.
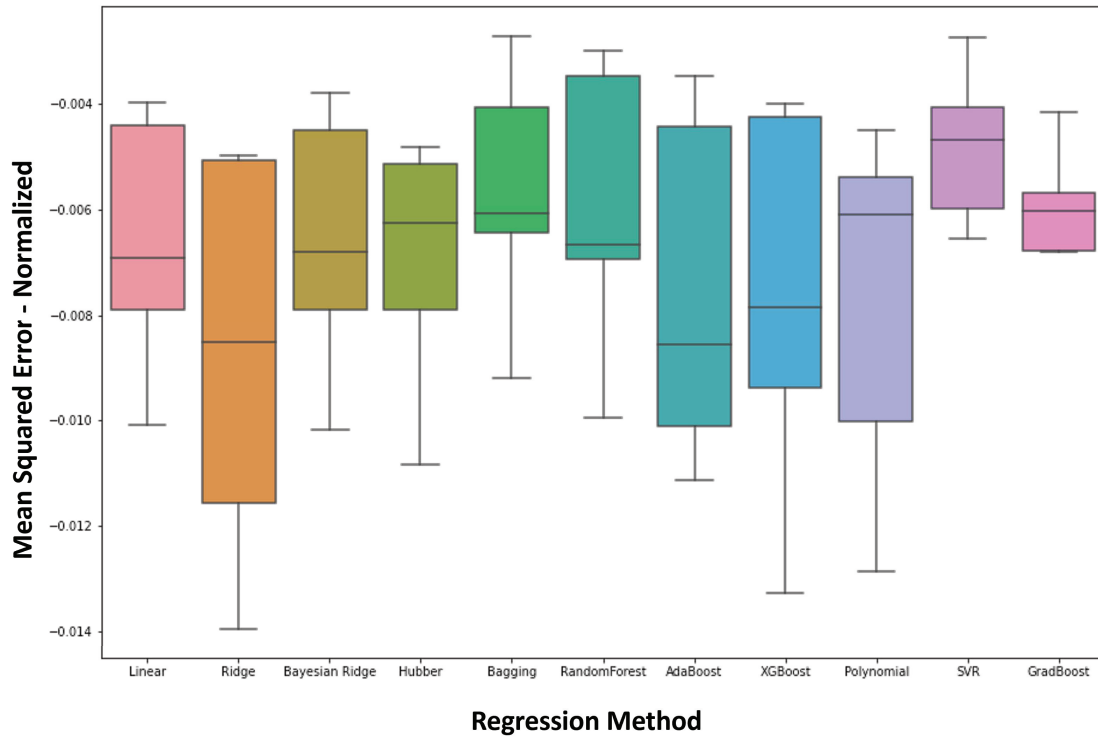
Figure 2.2: Box plots for five cross-validation mean squared errors on a normalized scale for multiple candidate ML models.

Among the tested regression methods, two of them had remarkably higher training and testing performance, which were gradient boosting and support vector regression (SVR) with a polynomial kernel. Since decision tree-based algorithms, such as gradient boosting, do not produce a smooth output [53], they present challenges when combined with control. Boosting methods are very successful at predicting the GC measurement points but can be very misleading for the areas in-between two consecutive GC points. In contrast, SVR predictions are more realistic, as they

yield predictions for the operating conditions between GC points that align closely with the expected behavior of the system. Thus, the polynomial kernel SVR is chosen because of its smooth, continuous properties and ability to predict intermediate points more accurately.

Remark 5 *It is important to note that the SVR model is trained only from GC measurements as described in Section 2.3.1, so the SVR model cannot capture the gas concentration prior to the GC measurement at 35 minutes. This requires the gas production to reach a pseudo steady-state such that the production rate at the catalyst surface is equal to that of the gas products entering the GC. In the beginning of the experiment, the electrolyte is not saturated with the product gases such as hydrogen, ethylene, methane and carbon monoxide. As a result, the initial gases produced must dissolve in the electrolyte until the electrolyte is saturated. Then, the gases begin to accumulate in the reactor overhead and become detectable by the GC. Therefore, until equilibrium is reached, the SVR predictions are different from the actual GC readings. After the time necessary for equilibrium passes, the SVR predictions and GC measurements start to converge.*

2.3.3   Support Vector Regression (SVR)

Support Vector Machine algorithms are similar to least-squares-based regression methods. However, instead of minimizing the residuals, SVR aims to optimize the generalization error bound; thus, it tries to come up with a generalized regression equation [8]. SVR assumes that the support vectors have at most a deviation of ε from the proposed function, and ideally intends to keep all data points within this margin. The SVR algorithm only penalizes points that are outside the support vectors [8]. A visualization of non-linear regression parameters of SVR is shown in

Figure 2.3.
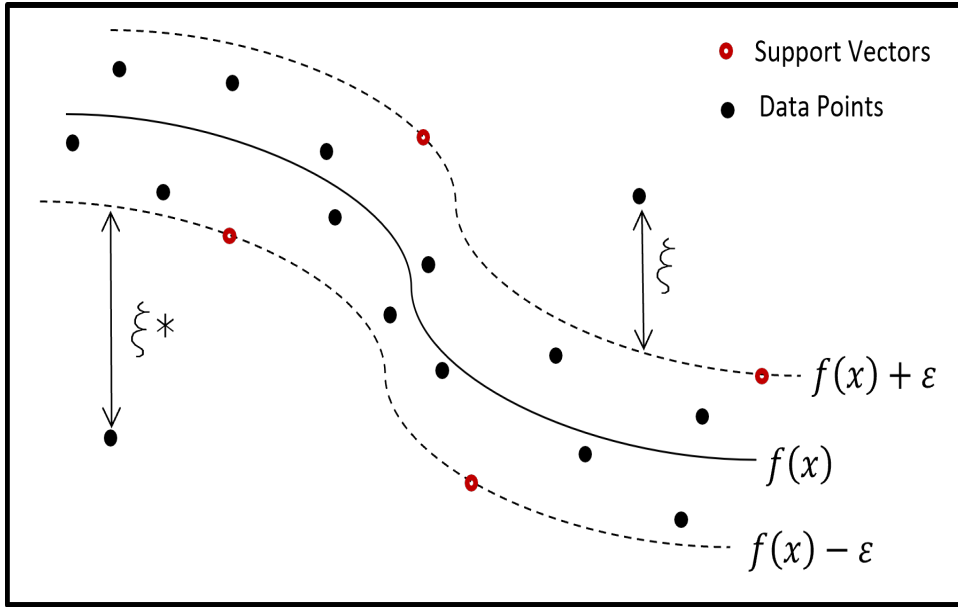
Figure 2.3: The proposed hyperplane f(x) and the margins f(x) $\pm \varepsilon$.

We note that ξ, ξ^* are slack variables for opposite sides of the support vectors and represent the distance between the outside points and the support vectors. For a linear SVR, the optimization problem is as follows [8]:

$$\min_{w} \ \frac{1}{2} \left\| w^2 \right\| + C \sum_{i=1}^{l} (\xi_i + \xi_i^*)$$

$$st. \begin{cases} y_i - \langle w, x \rangle - b \le \varepsilon + \xi_i \\ \langle w, x \rangle + b - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \tag{2.3}$$

where $\langle \cdot \, , \, \cdot \rangle$ is the inner product, w is the weight matrix, C is a parameter that decides the tolerance limit for divergences greater than ε, and l is the number of data points beyond the support vectors. The first two optimization constraints represent the support vectors. Solving the above convex

optimization problem yields the linear regression function. The same procedure can be applied to a kernel function, such as a polynomial. The degree of the polynomial is selected prior to the optimization.

The polynomial kernel $k(x_j, x_k)$ can be defined as follows:

$$k(x_j, x_k) = (1 + x_j^T x_k)^n \tag{2.4}$$

where x_j and x_k are instances of input vectors, x_j^T is the transpose of x_j, and n is the order of the polynomial.

$$k(x_j, x_k) = \langle \varphi(x_j), \varphi(x_k) \rangle \tag{2.5}$$

where $\varphi(x)$ is a non-linear kernel function and $\langle \cdot, \cdot \rangle$ denotes the inner dot product between the values. While our training data has 4 features, for simplicity, Eq. 2.6 below only illustrates 2 features with a 2nd degree polynomials for the two instances of the input vectors x_1 and x_2. Each feature contains the current, i_i, and surface potential, e_i, data of two experiments as shown in the following example:

$$x_1 = [e_1, i_1] \tag{2.6a}$$

$$x_2 = [e_2, i_2] \tag{2.6b}$$

$$\varphi(x_1) = [1, e_1, i_1, e_1^2, i_1^2, e_1 i_1] \tag{2.6c}$$

$$\varphi(x_2) = [1, e_2, i_2, e_2^2, i_2^2, e_2 i_2] \tag{2.6d}$$

$$k(x_1, x_2) = 1 + e_1 e_2 + i_1 i_2 + e_1^2 e_2^2 + i_1^2 i_2^2 + e_1 i_1 e_2 i_2 \tag{2.6e}$$

where e_1 and e_2 are surface potentials, and i_1 and i_2 are the current values of input the vectors x_1 and x_2, respectively.

The objective function would then take the following form:

$$f(x) = \sum_{j=1}^{N} (\alpha_j^* - \alpha_j) k(x_j, x) + b \tag{2.7}$$

where α_j^* and α_j are the Lagrange multipliers of the optimization problem. The convex optimization problem is solved to find the coefficients of the polynomial kernel [8].

2.3.4   Model Training and Feature Engineering

The Python package Scikit-Learn is used to fit the data to a support vector regression model. Grid search is implemented to find the best hyperparameters, by training the proposed model with all combinations of predefined sets of hyperparameters. The hyperparameters yielding the highest cross validation performance are selected to fit the final model. After grid search, the data are fit to a polynomial kernel of 5th order consisting of the surface potential, rotation speed and current as input parameters and ethylene concentration as an output. Since this model has a relatively small training set, feature engineering is applied to improve the model performance.

Figure 2.4 is useful for interpreting general trends in the data. Throughout the open-loop experiments, although the manipulated input variables (e.g., applied potential and rotation speed) are fixed, the ethylene concentration in the product is continuously decreasing. As shown in Figure 2.4b, the current density generally decreases over the duration of the experiments, and therefore, it increases the surface potential according to the relationship described by Eq. 1 (since the sign

in current and potential signifies the direction, increasing or decreasing remarks are made with respect to absolute values). The general decreasing trend of current is due to catalyst deactivation, which will be represented better in the model by the integral of the current as explained in the following paragraphs. The model is first fit to the data set, and some preliminary fitting results are presented in Figure 2.5.

Despite having a satisfactory approximation to the GC results and rational predictions in between two consecutive GC points, the model fails to catch some behaviors in the current and surface potential. In the experimental data, there exists a natural deviation in ethylene production rates measured under the same operating conditions. Also, the SVR model predictions have a strong correlation with the current input. However, mimicking the trend in the current could result in missing some inherent behavior in ethylene concentration, particularly when the experiment is run for longer than the 80 minutes used for the collection of the open-loop data used in the initial SVR model training. This motivates the pursuit of an additional input that can represent the catalyst degradation and account for the historical effect of the inputs throughout the experiment. Since the catalyst deactivates continuously, the ethylene concentration is expected to decrease monotonically over time under a constant applied potential and rotation as shown in Figure 2.5. Thus, the new input must be increasing or decreasing monotonically and must be derived from the available system parameters.

A parameter that is compatible with these prerequisites is the cumulative integral of the current. The current is always negative, as defined by the convention, since a cathodic potential is applied. Therefore, the cumulative area under the current is increasing at each time step even if the current is decreasing in magnitude. Thus, a very high integrated value of the current hints
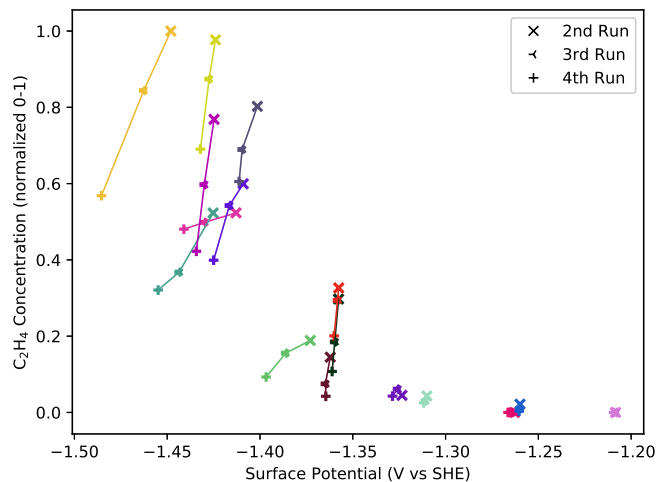
(a) C$_2$H$_4$ trend against current at 100 RPM in 0.2 M KHCO$_3$.
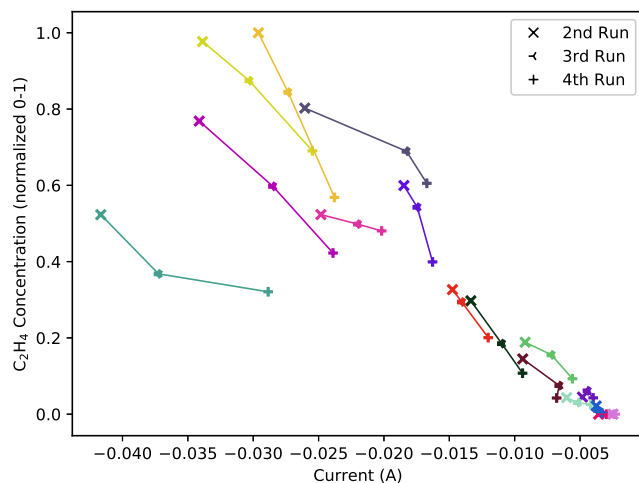
(b) C$_2$H$_4$ trend against current at 100 RPM in 0.2 M KHCO$_3$.

Figure 2.4: The trends in ethylene concentrations with increasing current and surface potential, illustrated with results from various experiments. Same colors show 3 GC results from the same experiment. Ethylene concentrations are normalized between 0 and 1.
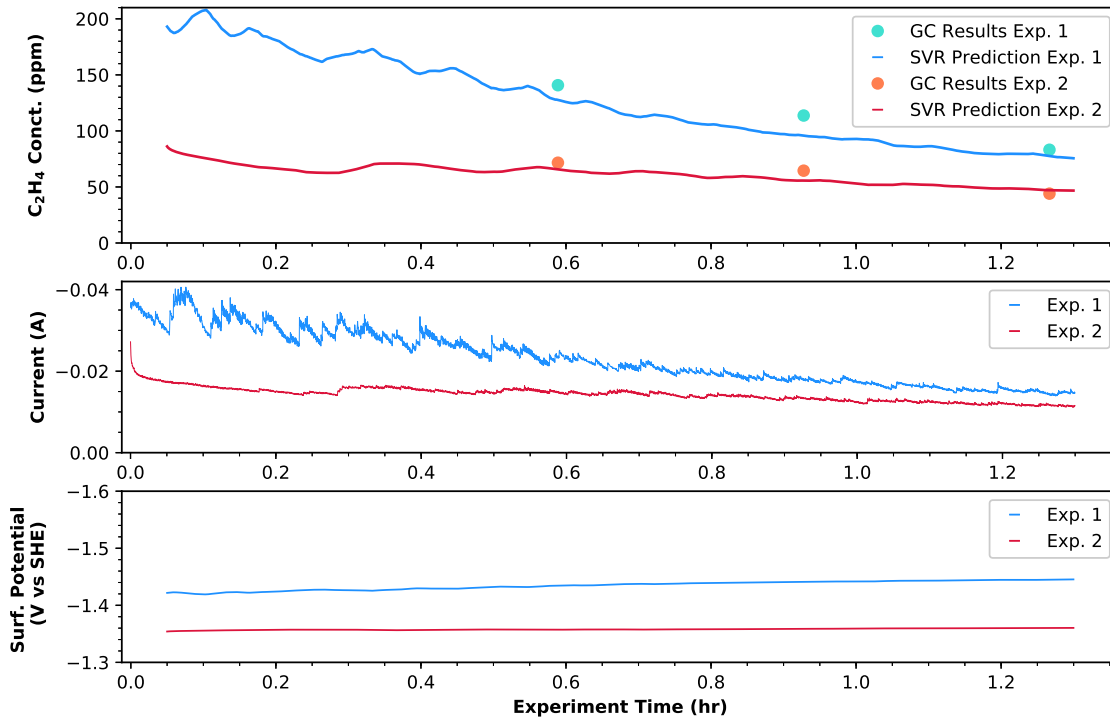
Figure 2.5: Performance of the SVR model in open-loop experiments. The concentration and surface potential plots do not start from time zero because those values are averaged over a 3 minute time window in open-loop experiments.

that the degradation has reached a very high level and the reaction is slowing down accordingly. This is consistent with the experimental trend that experiments at higher potentials show a more significant overall catalyst deactivation, because a higher applied potential leads to a larger flow of current. Additionally, the historical effect of the current in the experiment is taken into account through integration. The integral of the current can be defined as follows:

$$P(t) = \int_0^t i_{av}(\zeta) d\zeta \qquad (2.8)$$

where $P(t)$ represents the integral of the current term, t is the time, and $i_{av}(t)$ is the averaged

current values that the GC measurements are based on. The integral is calculated numerically with the trapezoidal rule based on per-second current data. The impact that the integral of the current parameter makes is shown on Figure 2.6. In the concentration plot of Figure 2.6, the large fluctuation in SVR predictions between the second and third GC points is due to fluctuations in the surface potential, but this is reduced to a more rational trend and the third GC point is captured after the integral of the current is introduced.

Since the data set is relatively small for dynamic modelling, auxiliary performance boosting methods are explored to increase the model accuracy. One of the beneficial methods is feature engineering [63]. The input parameters can be augmented by applying some mathematical transformations to the existing inputs. Some of the common feature engineering methods are polynomial, logarithmic and reciprocal transformations [63]. Since first-principles models are in the development phase, the mathematical forms of all parameters are not exactly known. For example, in the case of a polynomial transformation, $x_1^2 x_2$ might be a more efficient parameter than just x_2, where x_i is an arbitrary input. Similarly, feeding the transformation $log(x_1)$ might be a better input than x_1 if there is a possible logarithmic correlation in the nature of the phenomenon. Feature engineering can increase the number of inputs in the model, and this causes a trade-off between computation time and accuracy. The performance of the feature engineered models is tested under polynomial degrees 2, 3, and 4, logarithmic, and reciprocal transforms, in addition to regression tree and lasso transformations. The mathematical backgrounds for decision tree and lasso can be found in [117] and [126]. The results are presented in Table 2.2.

Feature engineering has the potency to improve the R^2 of the model by 8% and reduce the absolute error in the test set. The best-performing feature transformation is polynomial, and the
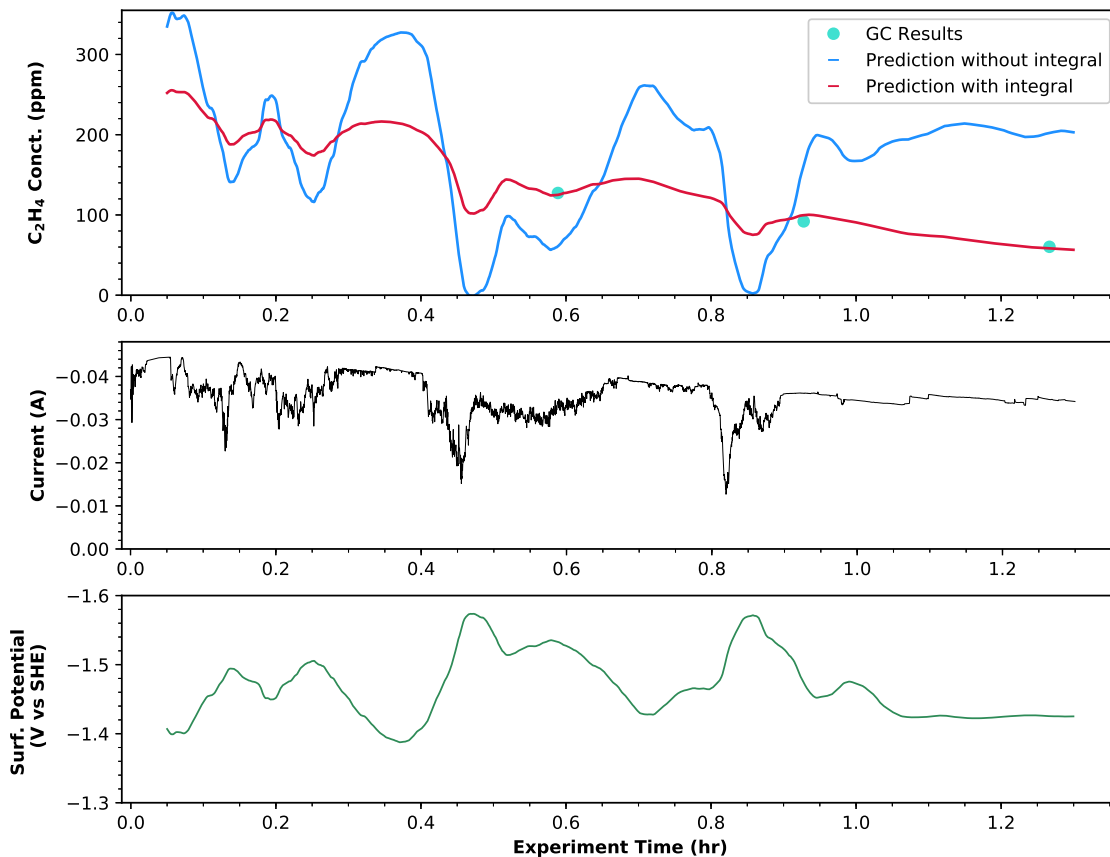
Figure 2.6: Improvement of the predictions after adding the integral of the current parameter as input. The concentration and surface potential plots do not start from time zero because those values are averaged over a 3-minute time window in open-loop experiments.

Table 2.2: Performance of the model using various feature engineering methods.

Feature Engineering Method	R^2	Mean Abs. Error
No Feature Engineering	0.842	13.24
Polynomial degree 2	0.894	10.39
Polynomial degree 3	0.922	11.31
Polynomial degree 4	0.924	10.31
Logaritmic + Reciprocal	0.853	14.59
Regression Tree	0.847	11.70
Lasso	0.841	13.24

R^2 metric continues to improve as its degree increases. The best polynomial is the fourth degree polynomial, but this transformation greatly increases the number of inputs. The third degree polynomial also appears to have a good performance, and it achieves this performance with fewer inputs. This is expected, as the third degree polynomial captures both the exponential increase in partial current density as a function of the applied potential and the fact that ethylene production becomes limited by mass transport at the highest overpotentials. Thus, a third degree polynomial feature transformation was selected to retrain the model.

Training and testing performance of the improved SVR model, after including the integral of the current as a new parameter and implementing the feature engineering, is shown in Figure 2.7. The mean absolute error is 11.3 ppm and the R^2 score is 0.92.

2.4   Real-Time Optimization

This section demonstrates a real-time optimization strategy for the electrochemical reactor. Specifically, we first approximate the cost and revenue of the reactor at various setpoints (i.e., ethy-
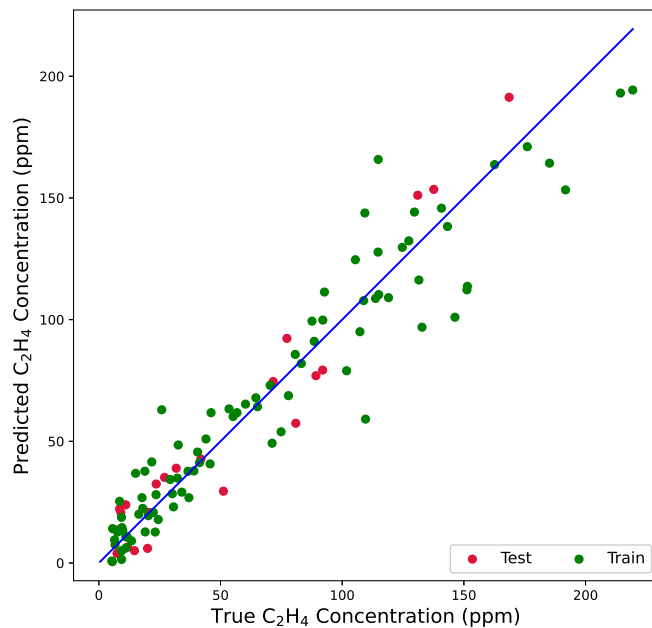
Figure 2.7: Testing and training performance of improved SVR model.

lene concentration). Subsequently, the optimization process is developed based on the economic

evaluation result of the reactor. Lastly, open source software IPOPT is introduced to perform the

real-time optimization for this work.

2.4.1 Setpoint Optimization

Setpoint optimization is critical for process operation; however, the location of the optimum

varies with different practical considerations. In this work, we develop a framework to decide the

optimum operating conditions for the electrochemical reactor automatically, by considering the

result from the economic evaluation based on the operational neural network model from [105].

Specifically, we approximate the energy consumption to maintain each ethylene concentration set-

point at steady-state by using the following equations:

$$I = \sum_{i=1}^{m} P_i e_i AF \tag{2.9a}$$

$$M_i = \frac{P_i AV_g}{F_0} \tag{2.9b}$$

$$E = IVt \tag{2.9c}$$

where Eq. 2.9a approximates the current density of the reactive surface by calculating the amount of transferring electrons, which is proportional to the overall production rate of the reactor. A is the area of the reactive surface, F is the Faraday constant, V_g is the standard molar volume of gases, F_0 is the feed flow rate of the CO_2 gas and e_i is the number of electrons transferred to form a molecule of the ith product. The values of these parameters are listed in Table 2.3. Additionally, P_i, $i = 1, \ldots, m$ is the molar production rate of the ith product, which is predicted by a statistical feed-forward neural network (FNN) model described in [105]. The FNN model is developed to take the surface potential (V) and rotation speed (r) of the electrode, which was fixed at 100 rpm for the control experiments in this study, as inputs to predict the production rates for all products. Figure 2.8 shows the comprehensive profile of the ethylene production rate predicted by the FNN model, which implies a rotational relationship between the surface potential and the ethylene production rate, and an inversely proportional relationship with respect to the proportional speed. The reduction of ethylene production at higher rotations speeds was determined to be caused by the decrease in the residence time of carbon monoxide at the electrode/electrolyte interface by [73]. Furthermore, according to the balance-based equation for this reactor (Eq. 2.9b), a unique

concentration (M_i) for each product can be found from the corresponding production rate P_i. Therefore, the energy consumption (E) to operate the reactor at a specific setpoint of ethylene concentration is calculated by adopting the equation of electric energy (Eq. 2.9c).

Table 2.3: Parameters of economic evaluation.

notations	value	unit
A	3	cm^2
F	96485.3	$C \cdot mol^{-1}$
V_g	22.4	$L \cdot mol^{-1}$
F_0	0.02	$L \cdot min^{-1}$
V	variable	voltage
I	variable	amp
E	variable	watt
M	variable	ppm

We assume the electric energy consumption is the only type of cost to operate this reactor, and we approximate the revenue of the reactor on the basis of the sale price and production rate of each product. Therefore, the optimum ethylene concentration setpoint to operate the electrochemical reactor can be determined by solving the following optimization problem:
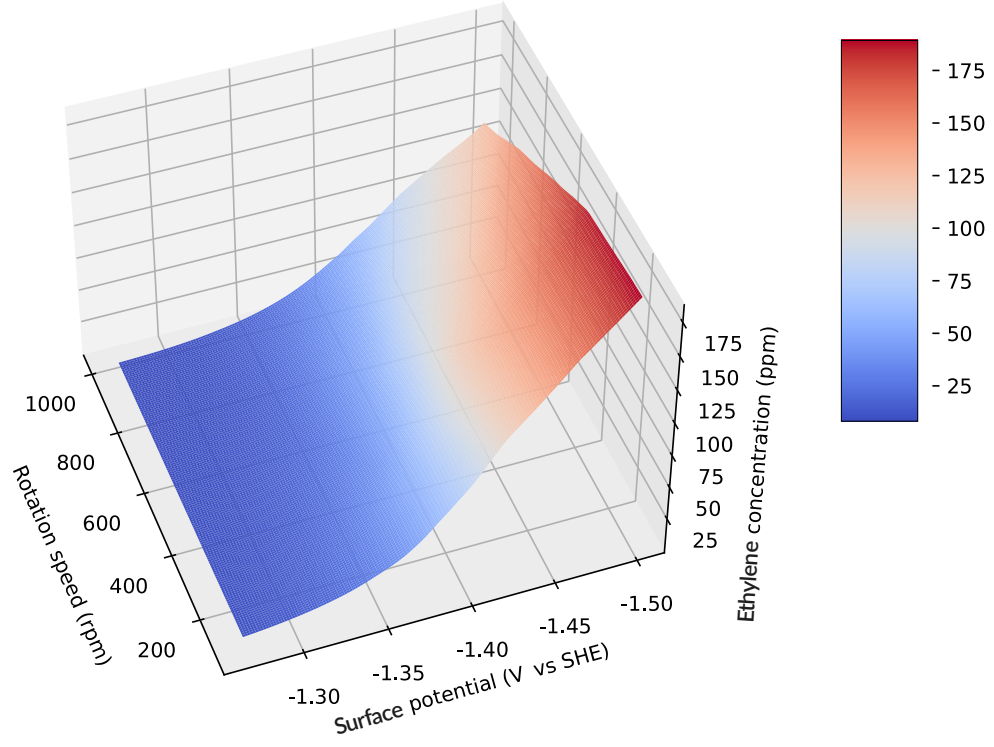
Figure 2.8: The map of FNN prediction for C_2H_4 production rates.

$$\mathcal{J} = \underset{\hat{x} \in \mathbf{D}}{\arg\max} \ R(\hat{x}, V) - C(I, V) \tag{2.10a}$$

$$s.t. \quad F_{nn}(V, r) \approx x \tag{2.10b}$$

$$C(V, I) = c_e \times E(V, I) \tag{2.10c}$$

$$R(\hat{x}, V) = \sum_{i=1}^{m} c_i \times \hat{x}_i \tag{2.10d}$$

$$r = 100 \tag{2.10e}$$

$$-1.5 \leq V \leq -1.27 \tag{2.10f}$$

44

Table 2.4: Chemical information of products.

Index	Products	Number of Transferred Electrons	Chemical Formula
1	methane	8	CH_4
2	ethylene	12	C_2H_4
3	methanol	6	CH_3OH
4	ethanol	12	C_2H_5OH
5	acetate	8	CH_3COO^-
6	ethylene glycol	10	$(CH_2OH)_2$
7	glycolaldehyde	8	$HOCH_2CHO$
8	acetaldehyde	10	CH_3CHO
9	n-propanol	18	C_3H_7OH
10	allyl alcohol	16	C_3H_5OH
11	acetone	16	CH_3COCH_3
12	propionaldehyde	16	C_2H_5CHO
13	carbon monoxide	2	CO

where $\mathbf{D}$ in $\mathbb{R}^m$ is the bounded state space of the production rates, and the boundary of the space $\mathbf{D}$ is determined from the training data set used to develop the FNN model [105]. The vector $x = [P_1, P_2, \ldots, P_m]$ contains the actual steady-state production rate of each product during the real-time reaction, and $\hat{x}$ denotes the predicted production rates given by the FNN model. c_e and c_i, $(i = 1, \ldots, m)$ are the price of electricity and the sale price for the ith product listed in Table 2.4.

In this study, the rotation speed of the working electrode is set to be constant at 100 revolutions per minute (rpm), and the surface potential is bounded from $-1.5V$ to $-1.27V$ vs SHE shown in Eq. 2.10e and Eq. 2.10f. The FNN prediction is used in Eq. 2.10c and 2.10d to approximate the revenue and cost of operating this reactor, and the constraint shown in Eq. 2.10b ensures the FNN prediction is accurate and reliable. It is noted that the bench-scale reactor used here is 8 to 10 orders of magnitude smaller in ethylene production rates compared to existing commercial

ethylene plants. Thus, the setpoint optimization is an interesting conceptual experiment as it includes operating constraints, but it does not capture the complexity of future electrified large scale production systems. This approach remains valuable as it is based on experimental data and should be translatable to increasingly larger systems.

Open-source software for large-scale optimization problems, IPOPT, is utilized to address the setpoint optimization problem. In this work, we use the forward finite difference method by adding small steps Δu on the optimized variables (i.e., potential and rotation speed), to approximate the first-order derivatives of the optimization problems. Additionally, second-order derivatives are approximated with the Quasi-Newton method to provide information for the calculation of search directions [188]. The derivatives and constant parameters (e.g., products and electricity prices) are provided to IPOPT to optimize the operating conditions in terms of surface potential and rotation speed. Finally, the results are converted to the corresponding ethylene concentration setpoint ($\mathbf{c}$) by using Eqs. 2.10b and 2.11, where F_o and a are the gas inlet flow rate (0.2 $L \cdot min^{-1}$) and a constant (1,000,000) for the unit conversion, as follows:

$$\mathbf{c} = \frac{aP_iAV_g}{F_o} \tag{2.11}$$

The optimization is performed for various electricity costs ranging from \$0.023-\$0.03 per kWh to provide optimized results over this range [54]. As shown in Figure 2.9, subfigures 2.9a and 2.9b demonstrate the approximate daily profit to operate the electrochemical reactor and the ethylene concentration profile at a rotation speed of 100 rpm. In the plot, the optimal setpoint shifts to a lower concentration of ethylene with increasing electricity price, implying that the optimizer

46

is capable of making intelligent decisions to reduce the production rate as operating costs increase. As a result, the optimum operating conditions for electricity price at 0.023 and 0.03 dollars per kWh are utilized for dynamic control experiments to demonstrate the controller performance in a cost changing scenario, such as the electricity Time-of-Use (TOU) rate plan in actual operation.

Remark 6 *The model is based on FNN calculations built using a statistical ML model, which generates averaged steady-state calculations taking catalyst deactivation into account. Since the catalyst deactivation cannot be defined with first-principle models in this study, the representation of deactivation is embedded into the FNN model with the statistical ML method. The FNN model is trained based on open-loop experiments and 3 data points were taken with equal intervals of 20 minutes, which shows the concentration decrease due to the catalyst decay. These results are averaged to give the pseudo steady-state concentration under a fixed applied potential and rotation speed. This model is optimized in our study to find economically optimal setpoints while accounting for the catalyst deactivation [105].*

Remark 7 *The catalyst needs to be regenerated once the selectivity shift happens. Catalyst regeneration is the only way to reverse the selectivity shift. However, this study does not explore catalyst regeneration. A future process operation and control study for this reactor will be based on a more stable catalyst, which does not exhibit severe deactivation.*

2.5   Feedback Control

This section demonstrates the application of the experimental control schemes to the electrochemical reactor. Driven by the motivation mentioned in section 2.2.2, the control objective is to
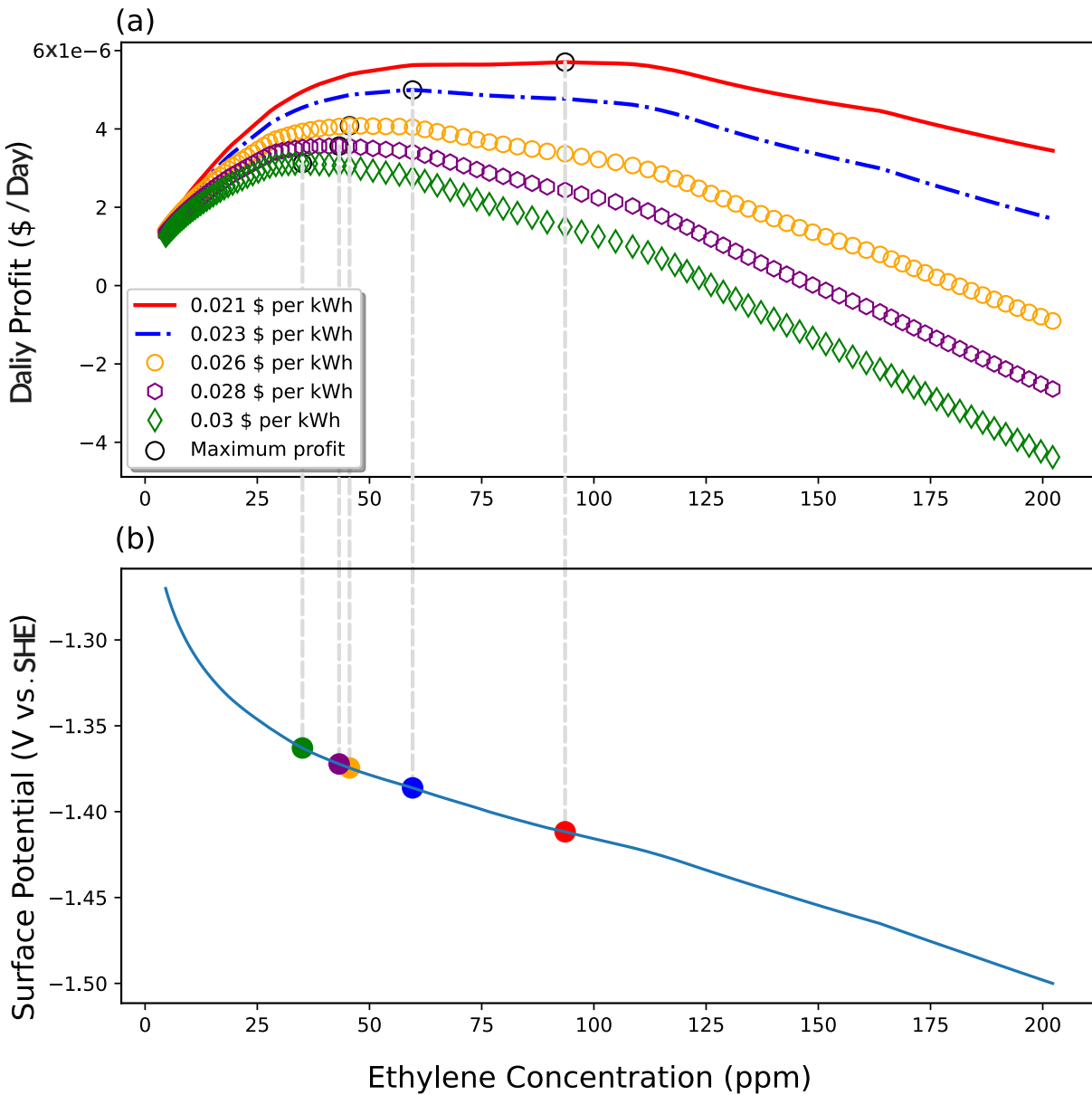
Figure 2.9: Approximated daily profit to operate the electrochemical reactor under various ethylene setpoints with changing electricity price.(a) The approximated daily profit profile to operate the reactor under different electricity cost. The open black points are the maximum profits that can be obtained by operating the reactor under the respective electricity prices. (b) The ethylene concentration profile under various surface potential conditions. The solid colored points emphasize the optimum ethylene setpoints that give the maximum profit.

regulate the ethylene production to energy-optimal setpoint values by manipulating the input variable. Therefore, a proportional-integral (PI) control algorithm is adopted to calculate the control actions for this study, where the manipulated input is the potential applied to the working electrode and the controlled output is the ethylene concentration measured from the reactor head-space gas phase product. The PI control algorithm in discrete-time form is expressed by the following formulas [113]:

$$e(t_k) = y_{sp} - y \tag{2.12a}$$

$$u_p(t_k) = K_C e(t_k) \tag{2.12b}$$

$$u_I(t_k) = u_I(t_k - 1) + \frac{K_C}{\tau_i} \left(\frac{e(t_k) + e(t_k - 1)}{2} \right) \Delta t \tag{2.12c}$$

$$u(t_k) = u_p(t_k) + u_I(t_k) \tag{2.12d}$$

where y_{sp} is the desired setpoint and y is the estimate of the controlled variable at time t_k. The controller error $e(t_k)$ is defined as Eq. 2.12a. Eq. 2.12b and 2.12c demonstrated the controller outputs from the proportional ($u_p(t_k)$) and integral ($u_I(t_k)$) terms, respectively, where K_C denotes the controller gain, τ_I represents the integral time constant, and Δt is the controller sampling time. The final control action ($u(t_k)$) calculated by the PI controller is shown as Eq. 2.12d, which is the potential on the surface of the catalyst.

Specifically, the surface potential (E_{surface}) required to reach or maintain the desired setpoint is computed by the PI controller, and then the corresponding applied potential (E_{applied}) is back-

calculated by the actuator based on Eq. 2.1, since the $E_{surface}$ compensates for the Ohmic drop from the electrolyte solution resistance. The surface potential is more relevant than the applied potential ($E_{applied}$) in describing the Tafel kinetics of electrochemical reactions involving electron transfer. It is noted that the primary and secondary current distributions in the RCE geometry are uniform, which is not the case for most existing CO_2 electrolyzers using H-type or compression cells. To further explain the control of the production rate, it is important to demonstrate how a change in $E_{applied}$ affects the current and $E_{surface}$. Intuitively, increasing the potential applied to the electrode can increase the potential on the reacting surface. However, the current density also has a positive correlation with the applied potential, and it is competing with the potential distributed to the reacting surface. Thus, the actuator needs to adjust the applied potential with respect to the output of the PI controller and the real-time measurement of the current density.

Remark 8 *The reactor may be controlled in real-time by manipulating two input variables: applied potential and rotation speed. The present study examines the control of a single gas phase product concentration (ethylene) by using a single-loop feedback controller that manipulates the applied potential. In addition to the applied potential, the rotation speed can be effectively used to control the outlet concentration of another gas product. The CO concentration is a good candidate as a second output given its strong dependence on rotation speed. Thus, multivariable control of the electrochemical reactor is a feasible task and will be discussed in a future work.*

On the other hand, the controller output should be bounded by the highest and lowest surface potential values in the training set of the SVR model. The applied potential does not have a strict upper bound due to the variable electrolyte resistance with ranged between 6.8 and 7.2 ohms and

instant current, but the surface potential is strictly bounded by the $E_{surface}$ range in the SVR training data set. In order to stay within the model confidence, an instant conversion from surface potential to applied potential is made during the control.

Due to the presence of a mixing volume in the gas phase of the reactor, it was previously determined that the GC concentrations during open-loop experiments were actually representative of the conditions in the reactor 5–8 minutes before the GC injection was taken, as discussed in Section 2.3.1. Thus, current and potential values were averaged over this 3-minute window for open-loop experiments used to train the SVR model. Because of the dynamics of the reactor, this method provided the best means of accurately correlating the production rate of ethylene with the conditions in the reactor. However, in the case of closed-loop experiments, the SVR model is used to predict the instantaneous production rate in real time, and the gas phase ethylene concentration model discussed in Section 2.5.1 accounts for the dynamic time delay in the response of the GC to the instantaneous change on the catalyst surface under a fixed rotation speed at 100 RPM. Thus, a sampling time of two seconds was used for averaging by the SVR during closed-loop control to provide real-time estimates. The real-time production rates predicted by the SVR model were then used as inputs for the gas-phase ethylene concentration model to control the reactor with a PI controller.

2.5.1   Ethylene Concentration Estimator

As mentioned in Remark 5, the SVR model is constructed to represent the product concentration on the catalyst surface. However, this does not represent what is measured by the GC. To have efficient control, it is necessary to estimate the gas-phase ethylene concentration in real time.

It is experimentally known that the GC and reactor overhead need a certain amount of time to reach equilibrium. This equilibrium volume is not exactly known and cannot be assumed to be equal to the reactor overhead volume, since gas bubbles adhere to the inner walls of the reactor and the surface of the rotating electrode depending on the rotation speed. To address this issue, the GC and reactor overhead are approximated as a continuously stirred tank reactor (CSTR), and the equilibrium volume and residence time are calculated using the first-principle CSTR equations, as shown in Eq. 2.14. It is mentioned in the previous sections that $r_A(t)$ represent the reaction rate and is calculated by the SVR model. The GC results are equalized to Eq. 2.14d, $r_A(t)$ is calculated by the SVR model and the τ and V parameters are varied to find the best fit. τ and V are correlated by the following equation:

$$V = \frac{60 \times \dot{V}_{CO_2} \times \tau}{1000} \tag{2.13}$$

where V is the equilibrium volume in L and $\dot{V}_{CO_2}$ is the volumetric flow of CO_2 into the reactor in $mL \cdot min^{-1}$. Since CO_2 flow rate is known to be 20 $mL \cdot min^{-1}$, the fitting is made by varying only the residence time τ for multiple reaction rate calculations and corresponding GC measurements.

This set of differential equations can be solved analytically to estimate the gas-phase ethylene concentration and can be modelled as follows:

$$\frac{dN_{\mathrm{A}}}{dt} = F_{A0} - F_A + r_A(t) \times s \tag{2.14a}$$

$$F_{A0} = 0 \tag{2.14b}$$

$$\frac{dC_A}{dt} = -\frac{1}{\tau}C_A + \frac{r_A(t) \times s}{V} \tag{2.14c}$$

$$C_A(t) = e^{\frac{-t}{\tau}} \int_0^t e^{\frac{\varsigma}{\tau}} \frac{r'_A(\zeta)}{V} d\zeta + C_0 e^{\frac{-t}{\tau}} \tag{2.14d}$$

$$r'_A(t) = r_A(t) \times s \left[\frac{mol}{s}\right] \tag{2.14e}$$

where N_{A} is the number of moles, F_{A} is the molar flow rate, τ is the residence time, $r_A(t)$ is SVR model calculation for instant reaction rate for ethylene production, s is the surface area of the catalyst, V is the volume of the proposed reactor and $C_0 = 0$ ppm. Several experimental data fittings with varying residence times and volumes were tested to fit the data with the model predictions. As a result, the suitable residence time and volume are selected to be 1800 seconds and 0.6 liters, respectively, as shown in Figure 2.10. A CSTR model assumes perfect mixing; however, this reactor is neither a CSTR nor perfectly mixed. In addition, the initially produced gases dissolve into the electrolyte, so the liquid must be saturated before the gases diffuse into the vapor phase at a constant rate. The reactor headspace concentration model cannot fully capture those details, and thus it fails to capture the first GC point. However, it is seen experimentally that an ideal CSTR can satisfactorily represent the reactor overhead for the remaining GC injections.

Remark 9 *A higher order model, such as multiple CSTRs in series, can be used to model the imperfect mixing better than a single CSTR. However, the development of a residence-time dis-*
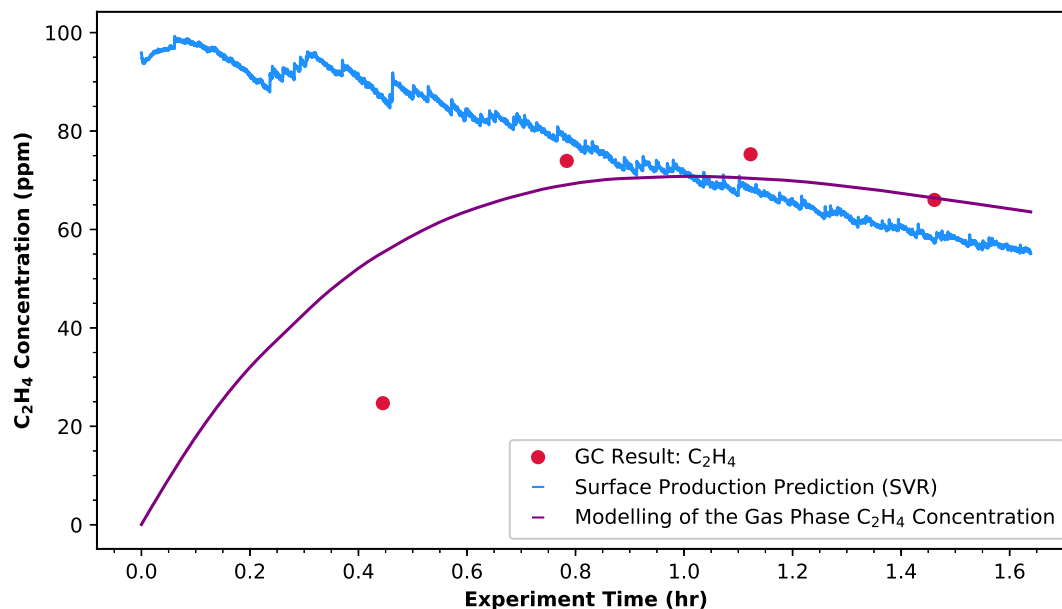
Figure 2.10: Data fitting of gas phase ethylene concentration on experimental data.

tribution model for this reactor is beyond the scope of this first work. Furthermore, this makes it

mathematically more complex to introduce the sensor feedback data discussed in Section 2.5.3.

2.5.2 System Constraints

Experimental systems are inherently vulnerable to varying conditions. In this electrochemical

setup, the initial catalyst activity can vary each trial within some error range. That is, with the same

initial fixed potential, two experiments might give different product concentrations, as illustrated

by the average rates and standard deviations summarized in Table 2.1. Although the catalysts were

cleaned with the same procedure each time, the exact catalyst activity was not known prior to the

experiment.

[68] proposed that methane and ethylene may share the same surface-bound CH_2 intermediate, with the dimerization rates of these fragments determining selectivity. In our extended feedback control experiments, after a certain surface potential threshold, it is seen that no matter how much potential is applied, there is no increase in the ethylene concentration, whereas the methane concentration keeps increasing. To prevent a selectivity shift to methane, it is crucial that the controller does not increase the surface potential to very high values (above $1.5\ V\ vs\ SHE$). This non-linear feature of the electrochemical reactor makes this control problem different from traditional reactors. A rule of thumb would suggest increasing the manipulated variable to the limit to reach the setpoint faster. However, in this electrochemical reaction case, this action would only deactivate the catalyst at such a rate that it would never reach the setpoint. Thus, the controller parameters must be selected attentively, so that the setpoint is reached slowly without shifting the reaction selectivity away from ethylene and towards methane. The anticipated selectivity shift is seen to start from the -1.45 V surface potential. The experimental outcomes are illustrated and discussed in Section 2.5.5.

The total concentration of carbon atoms going into methane or ethylene is seen to be constant when multiple dynamic experiments are compared in terms of carbon conversion to hydrocarbon gas products. The formula for this comparison is given in Eq. 2.15, and the experimental justification is shown in Figure 2.11:

$$C_C = C_{CH_4} + 2 \times C_{C_2H_4} \tag{2.15}$$

where C_c is the concentration of the relevant carbonaceous gas product. The ethylene concentration is multiplied by 2 because one molecule of ethylene contains 2 carbon atoms. The repetition of

(a) Methane and ethylene production rates   (b) Sum of carbonaceous gas products
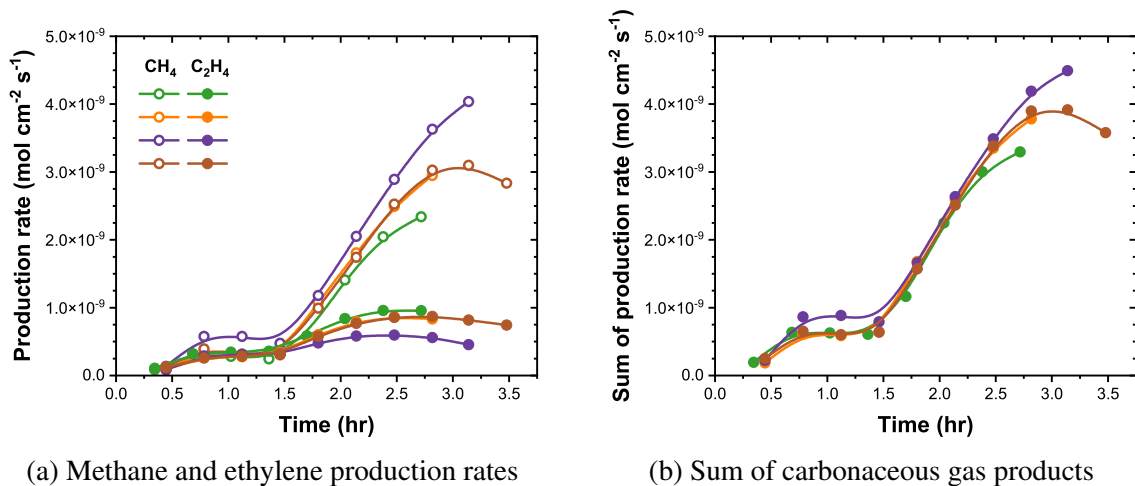
Figure 2.11: Comparison of methane and ethylene production from several experiments. Each color represents a different experiment where the setpoint for ethylene was increased to 2.7-fold at the 75th minute after being kept steady for 48 minutes.

the experiments demonstrated a constant concentration of carbon atoms at each GC measurement, regardless of the catalyst activity, when the surface potentials are constant before and after the setpoints. Although some experiments have the same potential, rotation speed, and current, this demonstrates that the difference in catalyst activity can dictate the ethylene concentration and selectivity and bring support to Hori's idea that methane and ethylene share a common surface-bound intermediate [68].

2.5.3 Feedback Control with Delayed GC Measurements

In previous sections, the modeling challenges that arose from the nature of the experimental data were emphasized. As a result of the uncertainty in experiments, the model is expected to give results within the standard deviation of repeated open-loop experiments. Implementing the control solely based on the reactor overhead estimator is expected to give close estimations to the GC results; however, this would exclude important measurement-feedback information that could be
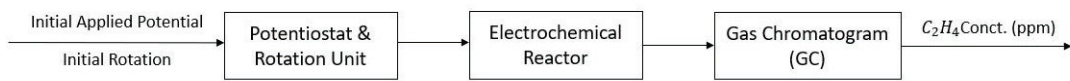
(a) Open-loop Diagram.
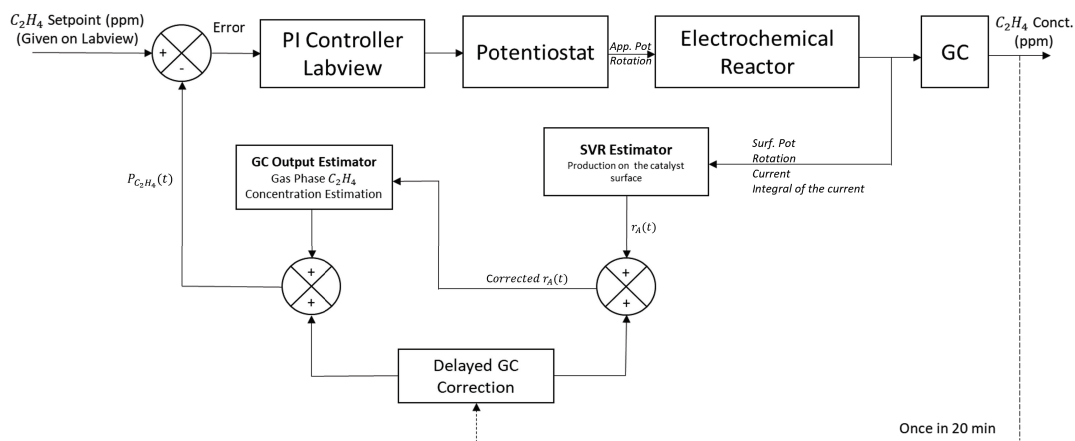
(b) Closed-loop Diagram with GC Correction.

Figure 2.12: Process control diagram for both open-loop process, and closed-loop control that incorporates the SVR model and GC feedback in the loop.

used to improve the control. Utilizing feedback from the GC measurements is a way to account for the experimental uncertainty, and this gives an opportunity for designing a complete closed-loop control, wherein the real-time sensor measurements directly impact the calculations of the manipulated inputs. Thus, the GC results obtained every 20 minutes are also used to improve the estimator. The control diagrams of the open-loop configuration and model-based, GC-incorporated, closed-loop configuration are shown in Figure 2.12.

Each component of the control system accomplishes the following: The PI controller node in LabVIEW calculates the control signal based on the PI control algorithm. The input to the control algorithm is the headspace ethylene concentration estimation (in ppm) and its output is the calculated surface potential. The potentiostat is both a sensor and an actuator. It senses the

57

real-time current and can manipulate the applied potential. It changes the applied potential based on the control signal sent by the PI controller. The reactions occur in the electrochemical reactor. The electrodes and sensors are connected to the reactor. The applied potential sensed in the reactor is converted to surface potential, the current values are measured, the rotation speed and the calculated variable of cumulative integral of the current are fed to the SVR model for reaction rate estimation, which is then used in first-principle calculations to estimate the headspace gas-phase ethylene concentration. The gas products accumulate in the reactor overhead. After the product gases equilibrate in the headspace, they travel to the GC sensor to be quantified in each 20 minutes. The quantification is processed automatically via an automated GC code and the concentration measurements (in ppm) are considered for feedback correction on the estimators. After the GC feedback correction, the error is calculated and sent to the PI controller for a new control signal.

In the closed-loop control scheme, the feedback data from the sensors is introduced to the ethylene concentration estimator to realize real-time adoption of the process control curve. However, the GC provides the concentration measurement results with a 14.33 minutes delay from the time of the reactor overhead sampling. Thus, a correction method was developed to estimate a probable past trajectory for the process control curve. The proposed correction is activated once the GC feedback data are received. The algorithm examines the past data and calculates the cumulative correction changes in the previous 20.33 minutes of the predictions (the time between 2 consecutive GC injections) to better align the predicted gas concentration with the most recent GC measurement. This correction affects both the SVR model and the gas-phase ethylene concentration estimation model. $r_A(t)$ term is the reaction rate with units $mol \cdot s^{-1}$. This correction scheme

is divided into two parts. The first correction is between $t = 0$ and $t = t_2$. The second correction is for after the second injection. The important abbreviations are presented below.

- n: the number of injections.

- t_i: i^{th} injection start time, $i = 1 \ldots n$.

- t_i^{end}: i^{th} injection end time.

- GC_i: i^{th} GC measurement result in ppm.

- $\widetilde{GC_i}$: The GC_i converted to $mol \cdot V^{-1}$.

- x_{i-1}: The correction applied to the SVR model (in ppm) after t_i^{end}, which should have been present during the previous injection time.

- $\widetilde{x}_{i-1}$: The x_{i-1} converted to rate in $mol \cdot s^{-1}$ using Eq. 2.16.

- $P_{C_2H_4}(t_i)$: Prediction of the gas phase ethylene concentration at t_i in $mol \cdot V^{-1}$.

- r_A : Prediction of SVR model (in ppm) converted to $mol \cdot s^{-1}$ using Eq. 2.16.

- $r'_{A,c}$: The cumulatively corrected SVR prediction converted to $mol \cdot s^{-1}$ using Eq. 2.16.

The equation for converting the concentration to rate is as follows:

$$r_{C_2H_4} = \frac{C_{C_2H_4}^{ppm}}{10^6} \times \frac{\dot{V}_{CO_2}}{60 \times 10^6} \times \frac{P}{RT} \tag{2.16}$$

where $r_{C_2H_4}$ is the production rate in $mol \cdot s^{-1}$, $C_{C_2H_4}^{ppm}$ is the ethylene concentration in ppm, $\dot{V}_{\text{CO}_2}$ is the volumetric flow of CO_2 into the reactor in $mL \cdot min^{-1}$, P is the pressure in Pa, R is the universal gas constant equal to 8.314 $\frac{J}{mol \cdot K}$, and T is the ambient temperature in K.

Given in Section 2.5.1, Eq. 2.14d shows the analytical solution for this reactor that is used to explain the mathematical background of the GC feedback correction. When the linear correction of the second GC injection $\widetilde{x}_1$ is applied to the prediction of the reaction rate, the analytical solution takes the form shown in 2.17a and is equal to the result of the GC measurement. 2.17b, 2.17c, 2.17d are the solution steps for 2.17a. The corrections between $t = 0$ and t_2 and the detailed solution are calculated as follows to find the linear correction term $\widetilde{x}_1$:

$$\widetilde{GC}_2 = e^{\frac{-t_2}{\tau}} \int_0^{t_2} e^{\frac{\zeta}{\tau}} \left(\frac{r_A(\zeta)}{V} + \frac{\widetilde{x}_1}{V} \right) d\zeta \tag{2.17a}$$

$$\widetilde{GC}_2 = e^{\frac{-t_2}{\tau}} \int_0^{t_2} e^{\frac{\zeta}{\tau}} \frac{r_A(\zeta)}{V} d\zeta + e^{\frac{-t_2}{\tau}} \times \frac{\widetilde{x}_1}{V} \times \int_0^{t_2} e^{\frac{\zeta}{\tau}} d\zeta \tag{2.17b}$$

$$\widetilde{GC}_2 = P_{C_2H_4}(t_2) + \frac{\widetilde{x}_1}{V} \times \tau \times \left(1 - e^{\frac{-t_2}{\tau}} \right) \tag{2.17c}$$

$$\widetilde{x}_1 = \frac{\widetilde{GC}_2 - P_{C_2H_4}(t_2)}{\tau \times \left(1 - e^{\frac{-t_2}{\tau}} \right)} \times V \tag{2.17d}$$

where $\widetilde{x}_1$ is converted to concentration by rearranging the Eq. 2.16. The correction after t_2 is

calculated as follows:

$$\widetilde{GC}_i = e^{\frac{-t_i}{\tau}} \int_0^{t_i} e^{\frac{\zeta}{\tau}} r_{A,c}(\zeta) d\zeta \tag{2.18a}$$

$$\widetilde{GC}_i = e^{\frac{-t_i}{\tau}} \left(\int_0^{t_{i-1}} e^{\frac{\zeta}{\tau}} \frac{r_{A,c}{}'(\zeta)}{V} d\zeta + \int_{t_{i-1}}^{t_i} e^{\frac{\zeta}{\tau}} \left(\frac{r_{A,c}{}'(\zeta)}{V} + \frac{\widetilde{x}_{i-1}}{V} \right) d\zeta \right) \tag{2.18b}$$

$$\widetilde{GC}_i = \frac{e^{\frac{-t_i}{\tau}}}{e^{\frac{-t_{i-1}}{\tau}}} \widetilde{GC}_{i-1} + e^{\frac{-t_i}{\tau}} \left(\int_{t_{i-1}}^{t_i} e^{\frac{\zeta}{\tau}} \frac{r_{A,c}{}'(\zeta)}{V} d\zeta + \frac{\widetilde{x}_{i-1}}{V} \int_{t_{i-1}}^{t_i} e^{\frac{\zeta}{\tau}} d\zeta \right) \tag{2.18c}$$

$$\widetilde{x}_{i-1} = \frac{\left(\widetilde{GC}_i - e^{\frac{(t_{i-1}-t_i)}{\tau}} \widetilde{GC}_{i-1} - e^{\frac{-t_i}{\tau}} T \right) \times V}{\tau \left(1 - e^{\frac{(t_{i-1}-t_i)}{\tau}} \right)} \tag{2.18d}$$

where $T = \int_{t_{i-1}}^{t_i} e^{\frac{\zeta}{\tau}} \frac{r_{A,c}{}'(\zeta)}{V} d\zeta$ is calculated numerically using the trapezoidal integration rule on

LabVIEW. We note that in Eq. 2.18a, the $r_{A,c}$ term accounts for all corrections. Eq. 2.18b divides

the calculations into two. The first integral term assumes that the previous correction was adjusted

with the linear correction so that the relevant prediction overlaps with the previous GC measure-

ment. The second integral term in Eq. 2.18b is structured to find the next correction that should

have been applied between $t = t_{i-1}$ and $t = t_i$.

It is important to note that this calculation assumes the equalities shown in Eq. 2.19 below:

$$P_{C_2H_4}(t_2) = e^{\frac{-t_2}{\tau}} \int_0^{t_2} e^{\frac{\zeta}{\tau}} \frac{r_A(\zeta)}{V} d\zeta \tag{2.19a}$$

$$\widetilde{GC}_2 = e^{\frac{-t_2}{\tau}} \int_0^{t_2} e^{\frac{\zeta}{\tau}} \frac{r_{A,c}{}'(\zeta)}{V} d\zeta \tag{2.19b}$$

The difference between the two terms is the correction in r_A.

The objective of the calculations so far is to calculate the SVR correction term $\widetilde{x}$. Following

this, the correction made on the surface production rate should be cumulatively reflected in the

headspace ethylene predictions. This requires a compensating factor for the ethylene production on the basis of the previously assumed surface production difference. Since GC corrections begin at t_i^{end}, the integral time interval is calculated between $t = t_{i-1}$ and $t = t_i^{end}$. This requires recalculation of $P_{C_2H_4}(t_i^{end})$ with the corresponding linear correction and would correspond to step changes in the headspace ethylene concentration prediction curve at $t = t_i^{end}$. The application of this correction to the reactor overhead ethylene prediction control curve and the relevant analytical solution are shown in Eq. 2.20 below:

$$P_{C_2H_4}(t_i^{end}) = e^{\frac{-t_i^{end}}{\tau}} \int_0^{t_i^{end}} e^{\frac{\varsigma}{\tau}} \left(\frac{r_{A,c}(\zeta)}{V} \right) d\zeta \tag{2.20a}$$

$$P_{C_2H_4}(t_i^{end}) = e^{\frac{-t_i^{end}}{\tau}} \left(\int_0^{t_{i-1}^{end}} e^{\frac{\varsigma}{\tau}} \frac{r_{A,c}{'}(\zeta)}{V} d\zeta + \int_{t_{i-1}}^{t_i^{end}} e^{\frac{\varsigma}{\tau}} \frac{r_{A,c}{'}(\zeta)}{V} d\zeta + \frac{\widetilde{x}_{i-1}}{V} \int_{t_{i-1}}^{t_i^{end}} e^{\frac{\varsigma}{\tau}} d\zeta \right) \tag{2.20b}$$

$$P_{C_2H_4}(t_i^{end}) = e^{\frac{-t_i^{end}}{\tau}} \left(\int_0^{t_i^{end}} e^{\frac{\varsigma}{\tau}} \frac{r_{A,c}{'}(\zeta)}{V} d\zeta + \frac{\widetilde{x}_{i-1}}{V} \times \tau \times \left(e^{\frac{t_i^{end}}{\tau}} - e^{\frac{t_{i-1}}{\tau}} \right) \right) \tag{2.20c}$$

From Eq. 2.20c, it can be seen that the concentration correction that should be made to the reactor headspace concentration estimator is proportional to $\frac{\widetilde{x}_{i-1}}{V} \times \tau \times \left(e^{\frac{t_i^{end}}{\tau}} - e^{\frac{t_{i-1}}{\tau}} \right)$ the term. This is easy to implement when integrated in LabVIEW as it is added to the integration term of the analytical solution in the real-time concentration calculations as shown in Eq. 2.20c.

The integral of time from $t = 0$ to t in the analytical solution of this headspace ethylene concentration model accounts for the past of the reaction rate. With each feedback from the GC measurement, to adjust the GC to instant production, it is assumed that the past of the experiment can be accounted for more accurately. For time interval $0 \leq t < t_{i+1}^{end}$, the SVR is also adjusted cumulatively as follows:

$$r_{A,c}{'} = r_A(t) + \sum_{k=1}^{i} x_k \tag{2.21}$$

and an example of this for $0 \leq t < t_5^{end}$ is like the following:

$$\begin{cases} 0 \leq t \leq t_2 & r_{A,c}{'} = r_A(t) + x_1 \\ t_2 < t \leq t_3 & r_{A,c}{'} = r_A(t) + x_1 + x_2 \\ t_3 < t \leq t_4 & r_{A,c}{'} = r_A(t) + x_1 + x_2 + x_3 \\ t_4 < t < t_5^{end} & r_{A,c}{'} = r_A(t) + x_1 + x_2 + x_3 + x_4 \end{cases}$$

Corrections begin at t_i^{end} and assume that this correction was applied between t_{i-1} and t_i, as shown in Figure 2.13. This methodology assumes a linear correction for a non-linear process. Thus, the correction is most effective when K_c is not high and when the concentration change between two consecutive GC measurements is not very large.

2.5.4 Simulation and Tuning

It is not practical to run experiments for several hours to determine the control parameters. Instead, a reactor simulator is built to find the controller tuning parameters. The simulator uses the data-driven SVR estimator, calculations of the first-principles modeling of the gas phase ethylene concentration, and a data-based correlation between the surface potential and current shown in Eq. 2.22. Using the available data, the optimal values i_0 and k are found from the data shown in Figure 2.14a using the following equations:

$$i = i_0 e^{k \times (E_{Surface})} \tag{2.22}$$

63
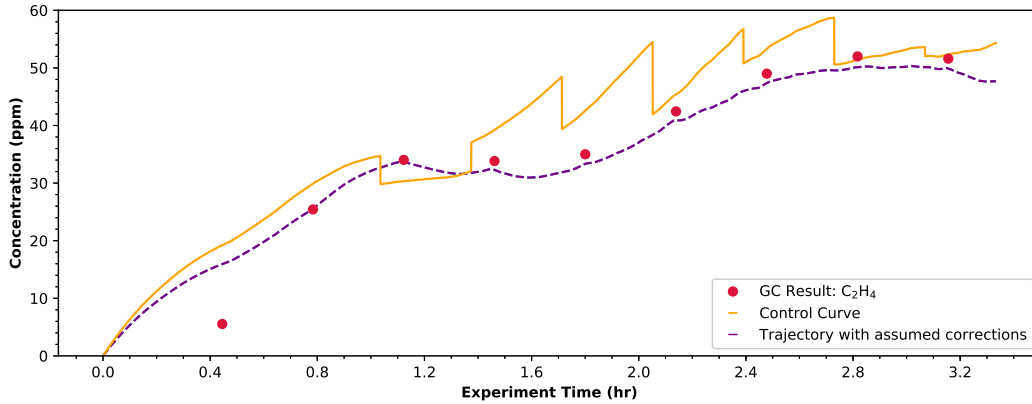
(a) The red dots represent the start of the corresponding injection t_i. The jumps in the control curve kick in at t_i^{end}. The purple dashed line is the assumed curve at the end of the 9th injection as the result of the back-calculation algorithm.
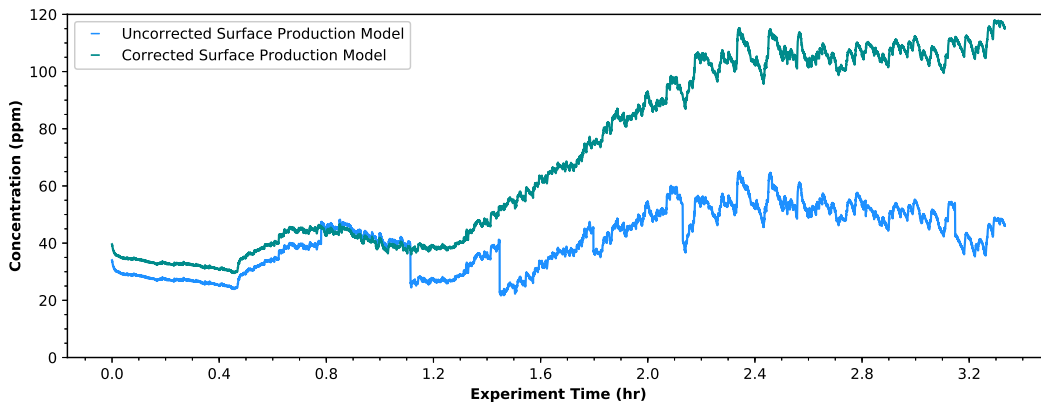
(b) The surface production correction has the same timing. The cumulative correction calculated based on GC feedback is seen when the two curves are compared, especially through the end of the experiment.

Figure 2.13: The cumulative corrections based on past data and its adjustment to real-time trajectories.

$$i = i_0 e^{k \times (E_{Applied} - i \times R + E_{Ag/AgCl})} \tag{2.23}$$

The starting fixed potential and rotation speed are defined at -1.65 V and 100 RPMs respectively, and the corresponding current is calculated by solving Eq. 2.23. Then, the PI controller code calculates the necessary surface potential, and this value is fed to the SVR model. The current at this surface potential is found using Eq. 2.22.

In addition to the current prediction at a specific surface reaction, an approximate relation for the catalyst deactivation is extracted using the available open-loop experiments. Specifically, open-loop experiments are conducted at a fixed surface potential, and catalyst deactivation manifests itself as a reduction in current throughout the experiment. The rate of reduction in current in open-loop experiments was found to be approximately linear with time at a given potential, and a correlation between applied potential and the rate of current loss was approximated as shown in Figure 2.14b by a power relation of the form:

$$\frac{di}{dt} = 3.17 \times 10^{-9} \left| E_{\text{Applied}} \right|^{10.82} \tag{2.24}$$

It is important to note that the developed simulator does not perfectly represent the reactor. Firstly, the activity of the catalyst is variable, and the simulator uses a model derived from the open-loop experiments. Due to this, experimental proof is needed to show that the controller parameters are suitable to compensate for the inherent decrease in the current. However, the simulator is expected to produce a good estimate of the controller parameters, K_c and τ_i. Second, the SVR model does not properly represent the selectivity shift that occurs at high surface potentials because it is not trained on experiments that are long enough for the selectivity shift to occur. As a result, the

65
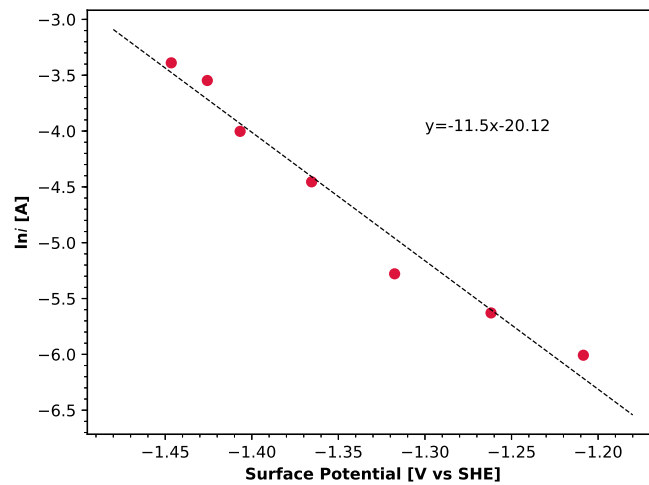
(a) Current-Surface Potential Linear Correlation
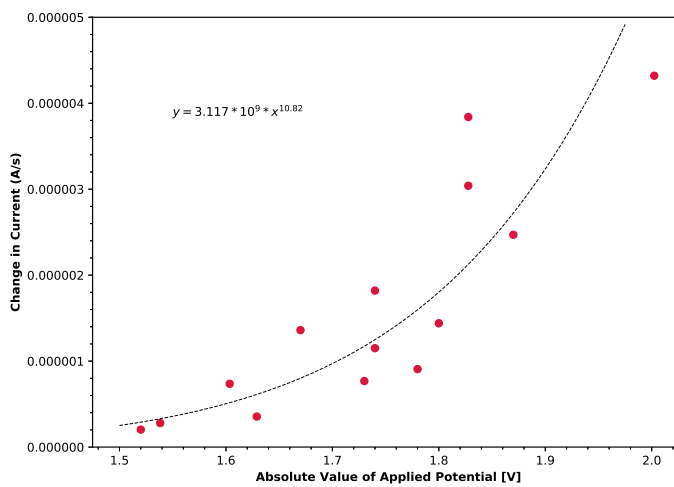
(b) Current-Applied Potential Power Correlation

Figure 2.14: Regression based on averaged experimental data to find current-potential correlation parameters.

controller has no knowledge whether the selectivity shift has occurred or not, and it assumes that increasing the potential will increase the ethylene concentration. This issue should be considered when selecting control parameters. Finally, the simulator does not use feedback data from the GC. Thus, the feedback correction mechanism for the GC discussed in Section 2.5.3 cannot be used during the simulation.

Before the simulator was built, an initial experiment was conducted using the control parameters, K_c = -2.45 $\times 10^{-5}$ and τ_i = 40 seconds. However, it was seen that the catalyst deactivates faster than the controller compensates for it. After the simulator was coded, the simulation and the experimental trajectories were compared, and they are shown in Figure 2.15. The experimental trajectory shown in Figure 2.15 is compatible with the GC results. Both the simulation and the experiment show that this proportional controller gain is not strong enough to compensate for the catalyst decay. As a result of more simulations, after the second setpoint change, a single pair of controller parameters were found not to be appropriate to control the process. This is an indication of the high non-linearity of the process and should be tackled by gain scheduling. As a result, two different gains were used for two different setpoints considered in our experiments.

Multiple combinations of controller parameters were tested with the simulator. The desired parameters should not drive the process output to the controller surface potential limits to avoid an early selectivity shift to methane. Also, the controller parameters should not lead to a very sluggish response where the catalyst deactivation overpowers the system before the setpoint is reached. The desired parameters in the simulation should allow some overshoot, since in reality the expected concentrations are lower. Figure 2.16 shows the controller simulation for two setpoints, 34 and 60 ppm, respectively. The control parameters are K_{c1} = -0.00011025, K_{c2} = -5.39 $\times 10^{-5}$ and τ_i = 40
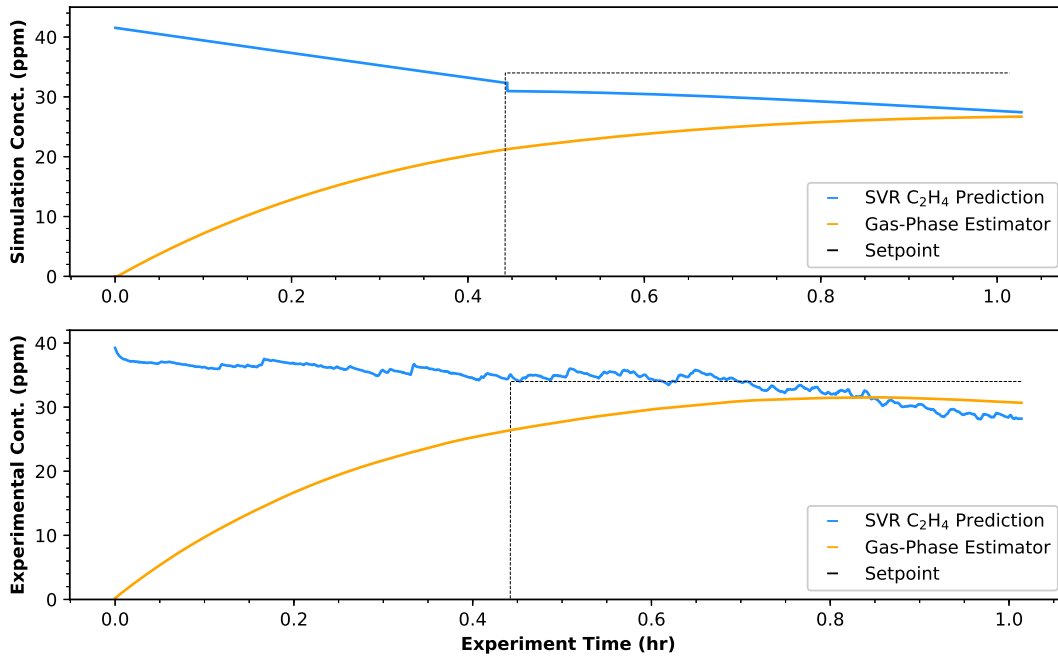
Figure 2.15: Comparison of simulated and experimental data for the setpoint of 34 ppm.

seconds.

2.5.5   Closed-loop Performance of Setpoint Tracking

In an effective control system, the controller should be able to drive the system to a new setpoint in case of a setpoint change. In a hypothetical scenario, the electricity price can change (due to TOU pricing, etc.), and the controller must adapt to keep the most economical or energy efficient production. In order to prove the efficacy of the estimator-based PI controller on this setup, a setpoint change is introduced between two economically optimal points. The setpoints are chosen to reflect a sufficiently high change in production over a reasonable period of time. Thus, the PI controller is tested for setpoints of 34 and 60 ppm.
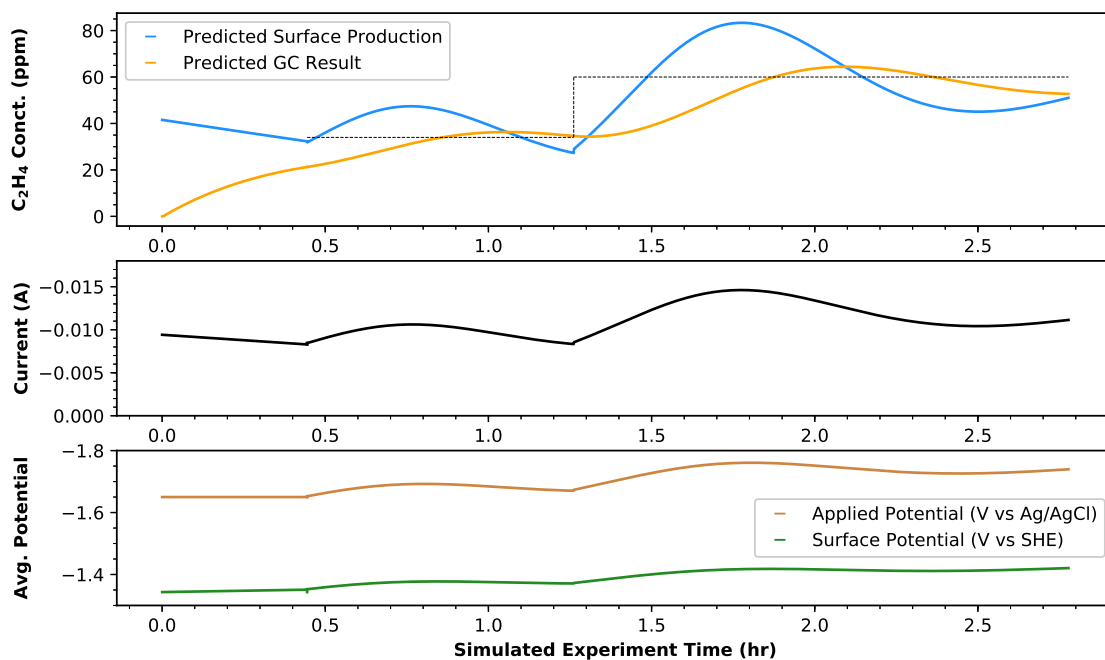
Figure 2.16: The simulation that was used to find the control parameters that were used in the closed-loop experiment.

In an open-loop setting, while the catalyst degradation decreases the number of available catalyst sites, the activation energy reduction caused by the catalyst slowly subsides and molecules need more energy to overcome the activation energy barrier. As a result, when the applied potential is kept constant, the number of molecules that can pass the activation energy barrier decreases as more reactive sites become unavailable. In a controlled experiment, the closed-loop potential is driven to more negative values by the PI controller to keep the energy of the molecules high enough to sustain production. The trends in current for the open-loop and closed-loop experiments are the opposite. In the open-loop experiment, the decreasing current is an indication that the ethylene concentration is decreasing as the catalyst deactivates over time. In the closed-loop experiment,

69

the increase in the applied potential drives the current in the opposite direction with respect to the open-loop current, indicating that the manipulated input compensates for catalyst degradation.

The experiment is run for 9 injections, and it is expected that the PI controller keeps the concentration at each setpoint. The controller starts at the 27th minute, and the setpoint change is introduced at the 76th minute, after obtaining 2 GC measurements. The experimental results are presented in Figure 2.17. The process succeeds in driving the gas-phase ethylene concentration to the first setpoint by the third injection. Then, after the second setpoint change, the controller drives the output close to 60 ppm at about 52 ppm. This experiment was repeated several times, with higher and lower feasible proportional controller gains, and with different catalysts. In each of the experiments, the final concentration was recorded to be between 52-54 ppm. One of the potential reasons that this system does not fully reach the higher setpoint is that the dynamics of the system cause the process to approach the second setpoint after 2 hours. By this time, the catalyst has largely deactivated. The surface potential needed to increase the concentration of this system causes a selectivity shift. Figure 2.18 shows this selectivity shift to methane more clearly. Especially, between injections 7 to 11, the methane concentration increases as the ethylene concentration remains constant under increasing potentials and current. This is an experimental justification that the energy given to the process was being used to produce more methane instead of producing ethylene. Thus, if the system has not reached the setpoint by the time the selectivity shifts to favor methane, it is not possible to increase the concentration of ethylene to the setpoint by increasing the potential. It takes around the time of 3 GC measurements to reach the neighborhood of a new setpoint. The results show that the controller can drive the system to the neighborhood of the setpoints within the error margins of 10% to 12%.

Remark 10 *The setpoint change would not be possible without the integral of the current input. The current and surface potential values at 100 RPM recorded through the end of the repetition experiments were seen to yield much higher concentrations in the early GC measurements of related open-loop experiments. However, the cumulative integral of the current brings the concentration closer to the GC results by capturing appropriately the catalyst deactivation over time.*
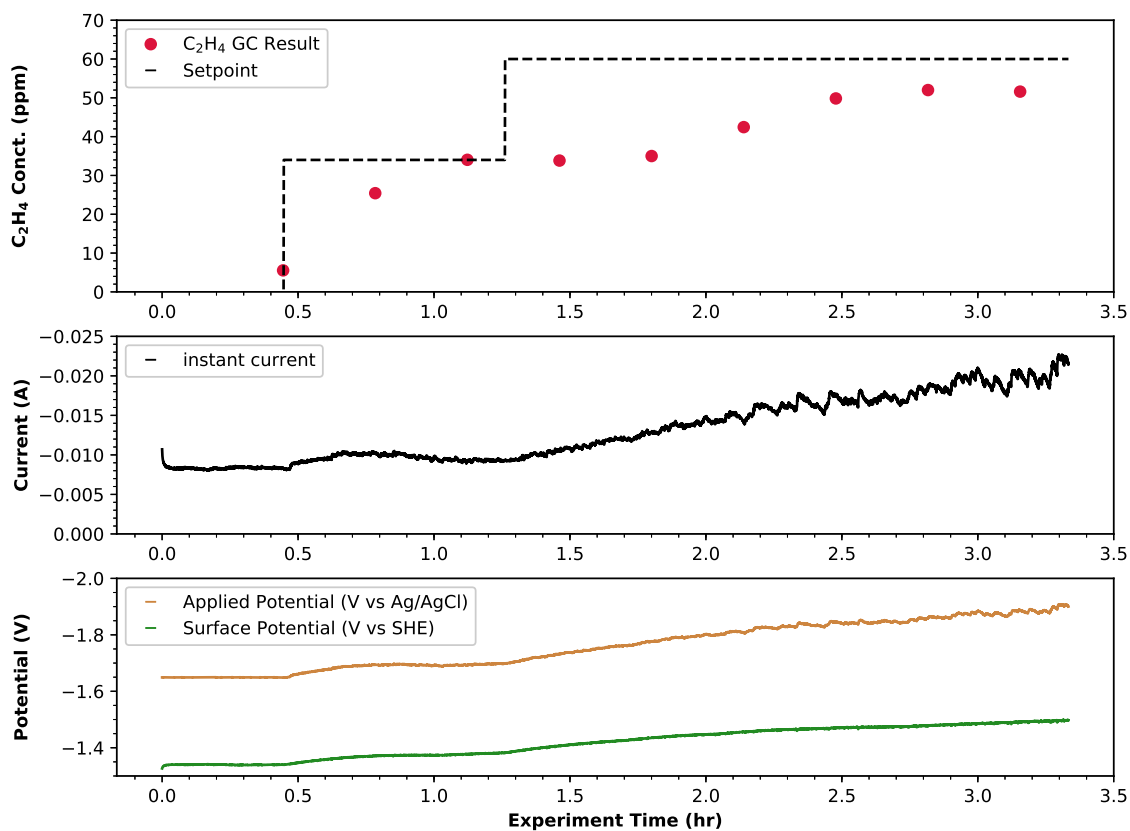
Figure 2.17: Closed-loop experiment with setpoint changes.

71
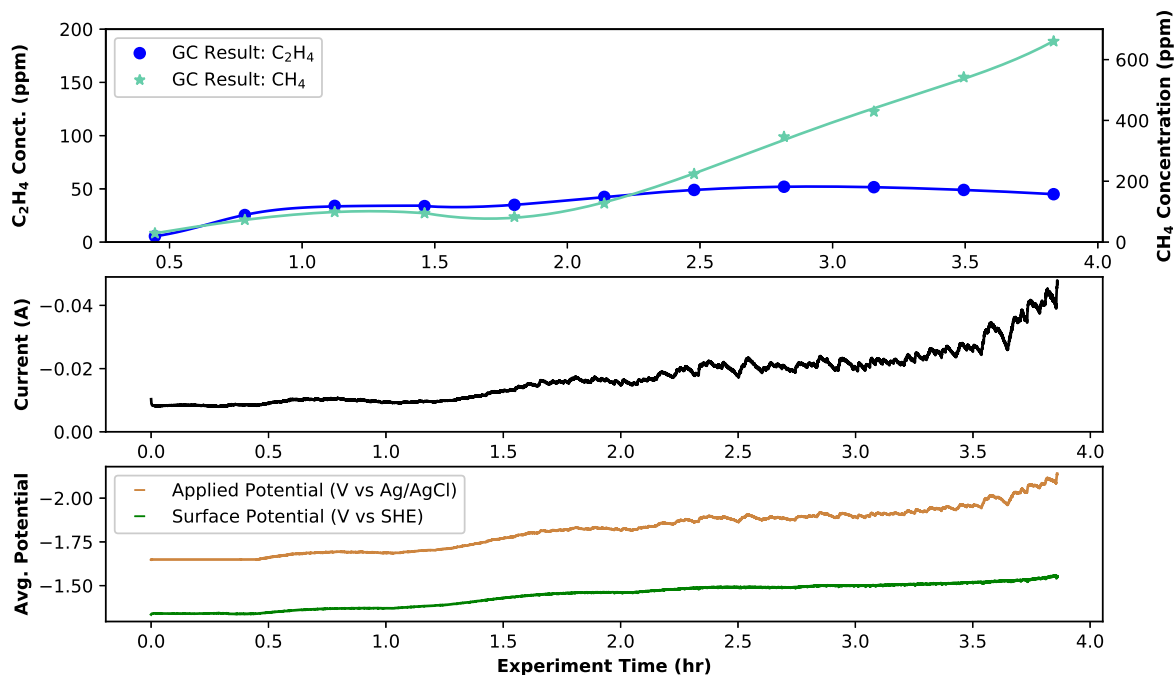
Figure 2.18: Ethylene and methane concentration comparison. The selectivity shift to methane can be seen starting from the 7th injection.The surface potential and current continue to increase, while the trend of ethylene concentration remains steady and methane production increases sharply.

Chapter 3

Digitalization of an Experimental Electrochemical Reactor via the Smart Manufacturing Innovation Platform

3.1   Introduction

The 21$^{\text{st}}$ century has witnessed a rapid increase in the amount of data produced by many industrial sectors, including chemicals manufacturing. The use of process data to improve process operation via model-based systems and control engineering has initiated the Fourth Industrial Revolution, which is described as Smart Manufacturing (SM) in the United States and Industry 4.0 in Europe (e.g., [24, 79, 216]). The embedding of digital technology into nearly all components of a manufacturing process is generating a digital transformation that enhances decision-making, increases energy efficiency, and improves productivity [153]. The digital transformation of a manufacturing process includes the seamless integration of advanced sensors with data-based machine

learning models, simulation technologies, advanced real-time optimization and control strategies, and data-driven, decision-making tools with a real-time view of the factory and supply chain. The abundance of data allows for the automation of processes and the development of intelligent systems, which is crucial to eliminating human error, to facilitating data acquisition, and to accelerating manufacturing research breakthroughs by combining operation and information technologies under the scope of smart manufacturing. In a broader perspective, smart manufacturing is concentrated on the optimization of cyber-physical components by resolving challenges associated with uncertainties including raw materials and supply chain.

Although smart manufacturing is beneficial for most industrial applications, realistic digital transformation is difficult to integrate into most industrial applications due to its time-consuming nature and the need for advanced operator training [129]. The growth of big data, catalyzed by Internet of Things (IoT) technologies, has exposed challenges in the implementation of smart manufacturing. Specifically, the ability to extract more meaningful relations and greater insights involves working with the data in a more integrated fashion with methods for large, more complex data sets. In recent years, machine learning algorithms have evolved to overcome some of these aforementioned challenges, as they can identify hidden patterns or trends in big data sets that can be used for better decision-making. Using real-time data directly from the operation increases the effectiveness of the analyses, and positions for greater understanding from the beginning and not at a later step in the analysis.

One of the main objectives of smart manufacturing is to establish relationships within large data sets in an efficient and timely manner for operational modeling and troubleshooting. As machine learning algorithms are becoming more popular, the spectrum of data-driven modeling

74

options is expanding. Regression methods such as Artificial Neural Networks (ANN), Support

Vector Machines, and Gradient boosting have been successful in modeling process operational

data. With respect to recent work in the domain and other machine learning (ML) methods, [208]

explored the use of recurrent neural networks (RNNs) for modeling time series data in chemical

processes and demonstrated that RNNs can be effectively used in nonlinear control schemes such as

model predictive control (MPC). In conjunction with ML methods, several works have successfully

integrated ML algorithms into chemical engineering processes to develop new robust predictive

models for multiphase flows and reactors. [224] have summarized recent ML applications to

hydrodynamics, heat and mass transfer, and reactions in single-phase and multiphase flow systems.

Machine learning approaches have also found use in the realm of quantum calculations to predict

chemical properties and reactivity. For example, a deep neural network (DNN) was constructed

to estimate reaction rate parameters from an extensive partition function database [86]. The use

of transfer learning in a DNN for activation energy estimation was also demonstrated, starting

from a dataset generated from Density Functional Theory (DFT) calculations [51]. In addition

to the aforementioned works, the use of AI and ML reaches beyond process data modeling in

operational contexts. For instance, [55] developed a convolutional neural network model that is

used to detect effervescence in a multiphase flow through a high-speed camera that collects images

of fluid dynamics and counts the number of bubbles in each image. In another study, the pressure

drop of a cyclone separator was modeled with a hybrid generic algorithm radial basis function

neural network (GA-RBFNN) and then this model was optimized for the most optimal separation

parameters [40]. The aforementioned works highlight the importance of ML algorithms in modern-

day processes and the diverse ways in which these can be integrated into smart manufacturing

practices.

Smart manufacturing applications can capture a complete process cycle. For example, [93] developed a smart manufacturing approach for a distributed control sequence for a steam methane reformer (SMR) that exploits a reduced-order model (ROM) based on large data sets extracted from computational fluid dynamics (CFD) simulations, which is a time-consuming task. In this work, an optimizer sets the most optimal set-points for fuel distribution in burners while a ROM was updated automatically over time with new CFD simulations. Moreover, this framework benefits from advanced sensors like infrared (IR) cameras for temperature distribution detection. Similarly, [144] designed an industry 4.0 framework for metal alloy additive manufacturing, optimized the production by employing micro-, meso- and part-scale finite element method (FEM) analysis of manufactured parts, trained convolutional neural networks (CNN) for production monitoring and defect detection, and offered a data transfer strategy between manufacturing machines, factories, and research labs to continuously develop additive recipes based on feedback. Cloud analytics was used to store the voluminous data and automatically update the CNN model with new incoming manufactured part pictures, proposing a strategy for the testing/training data set split for the auto-CNN model update. In addition to those projects demonstrating the application of the Smart Manufacturing paradigm, [89, 13] presented in detail how a common platform accelerates and facilitates their research for discrete part manufacturing systems by integrating a test-bed with the Smart Manufacturing Platform.

Meanwhile, the general trend in energy research is transitioning toward renewable resources from fossil fuels. One of the promising research areas in sustainable energy production and chemicals manufacturing is electrochemistry. Electrochemical CO_2 reduction is an attractive and emerg-

ing process that can produce valuable chemicals such as ethylene, ethanol, acetaldehyde and syngas. However, due to the complexity of the reaction mechanisms, industrial-scale first-principle models of this process have not yet been developed. As a result, data-driven models appear as an attractive alternative to represent this process and enable process optimization and control strategies [114]. For this purpose, large data sets should be generated from experimental electrochemical cells; a task that introduces significant data acquisition and data modeling challenges. Thus, in the present work, Smart Manufacturing techniques have been used to accelerate this phase through the direct connection of an experimental electrochemical reactor setup to the Smart Manufacturing Innovation Platform (SMIP) developed by the Clean Energy Smart Manufacturing Institute (CESMII), a U.S. government funded organization. The SMIP offers the necessary tools for data transmission, configurable connectors to historians and Open Platform Communication (OPC) servers, data storage, correlation extraction, and data contextualization. With automation and digitization of experimental systems, the communication between different area experts is eased and contextualized data along with controlled experiments is expected to contribute into scale-up efforts of bench-scale electrochemical reactor systems.

For our electrochemical reactor, the smart manufacturing process starts with the objective of demonstrating that the complex catalytic reaction can be controlled to optimize the conversion of CO_2 to Ethylene. There is a need to use the data to build a control model. Smart manufacturing also carries the objective of building outlines or templates for the finished models to save time and effort for recurring similar applications. The smart manufacturing process starts from the idea of using data to create process operation models, but it also aims to create reusable outlines or templates called 'Profiles' to save time and energy for the recurring similar applications. The other

objective of smart manufacturing is to simplify the instrumentation and the connection, ingestion and contextualization of the real time data from the reactor operation. This is especially difficult for legacy equipment and instrumentation and when multiple vendor products are involved. As the amount of data increases exponentially, the need for storage and easy access is becoming vital. Legacy manufacturing operations are not easily instrumented for digital data ingestion and as a result continue to push data inconsistently in time and format, making it very difficult for the data to be accessible by multiple area experts. In the last decade, many of the product vendors created proprietary software tools for their equipment, which are often not compatible with each other and/or difficult to interface with external software applications. Finally, legacy software applications themselves have tended to embed data for a particular function, essentially trapping it for other uses. The incompatibility and proprietary nature of applications make it very difficult for instrumentation equipment manufactured by different vendors to be used consistently to connect, ingest and bring data together. This is the case for systems used in electrochemical reactors research and development. It can be argued that the lack of a standard model or common platform to exchange data has led to a limited opportunity in terms of exploiting available data modeling options in the advancement of complex manufacturing systems. Cloud technologies offer a service to store voluminous data and make them accessible anywhere and anytime, but various cloud vendors still lock in data in respective platforms. For our electrochemical reactor research, there is urgency in translating greener electrified chemicals and fuels manufacturing technologies from the bench-scale to commercial scale in order to quickly decrease global carbon emissions. Thus, there is a unique opportunity to leverage cloud technologies for shared computational resource usage to significantly decrease the time needed for the analysis, optimization and scale-up of electro-

chemical systems. As a last point, the availability of cloud technologies and their interface to the physical operations opens additional cybersecurity vulnerabilities with integrating multiple cloud products and/or ingesting data from multiple vendor instrumentation. The data transfer between a cloud resource and manufacturing process equipment, sensor or machine or local processor must be encrypted (HTTPS), safe, and the data stored on the cloud must be protected. To this end, a smart manufacturing framework must address the necessary cyber-precautions to keep the user data safe, with policies and user roles that clearly indicate who can read or modify the data.

This paper presents smart manufacturing components of how smart manufacturing has been applied to the control of an electrochemical CO_2 reduction reactor and its instrumentation with the SMIP. The rest of this chapter is organized as follows. In the next section entitled "Smart Manufacturing in Experimental Electrochemical Reactor Setup," the experimental reactor setup and the key Smart Manufacturing hardware and software setups are described. In the section, entitled "Advanced Sensors," the real-time point sensor measures are described and the need for an automated gas chromatography (GC) spectra processing algorithm is explained. In the section entitled "CESMII Smart Manufacturing Innovation Platform," the SMIP involvement with examples and a novel LabVIEW interface are presented. In the section entitled "Docker Container," the advantages of the use and implementation of Docker containers are discussed. Finally, the iterative evaluation process and resulting advantages of using ML and hybrid models are addressed in the section entitled "Electrochemical Reactor Modeling Using SMIP".

3.2 Smart Manufacturing in Experimental Electrochemical Reactor Setup

Taking a smart manufacturing perspective on electrochemical research on CO_2 reduction in our experimental test bed is illustrated in Figure 3.1. The setup is fully automated, and the data collected from the test bed are used to extract insight and build operational Machine Learning (ML) models. Applied potential and current data are collected by a potentiostat, which is composed of a sensor for measuring the electrical current and potential and an actuator for tuning the applied potential.

Liquid product concentrations are measured using nuclear magnetic resonance (NMR) after the experiment is completed, and the gas concentrations are measured online through gas chromatography (GC) in intervals of 20 minutes. It is notable that the injection and processing of the GC system need to be automated to use the gas product analysis in real-time. Collected data are processed to extract relationships between inputs (e.g., applied potential, electrode rotation speed, and temperature) and outputs (gas production rates) for ML models. However, there are challenges in building models from actual experimental data. For example, the limited amount of the GC data produced per experiment limits the use of some ML modeling approaches for dynamic process modeling, for example recurrent neural networks. These algorithms require large data sets to be effectively trained in order to establish a robust predictive model. Moreover, the quick deactivation of atomically flat catalysts causes a shift in selectivity, resulting in the production of other products than the desired ones. To overcome these issues, statistical ML methods maybe enhanced with kinetic constants, which were calculated by [105], the cumulative integral of current, which

was used to set a correlation between the current transfer and catalyst deactivation, and feature engineering, which was used in Chapter 2 to increase the model performance. Finally, a feedback control strategy is implemented to drive the process to a desired, energy-optimal setpoint. Our experimental research setup is shown in Figure 3.2.
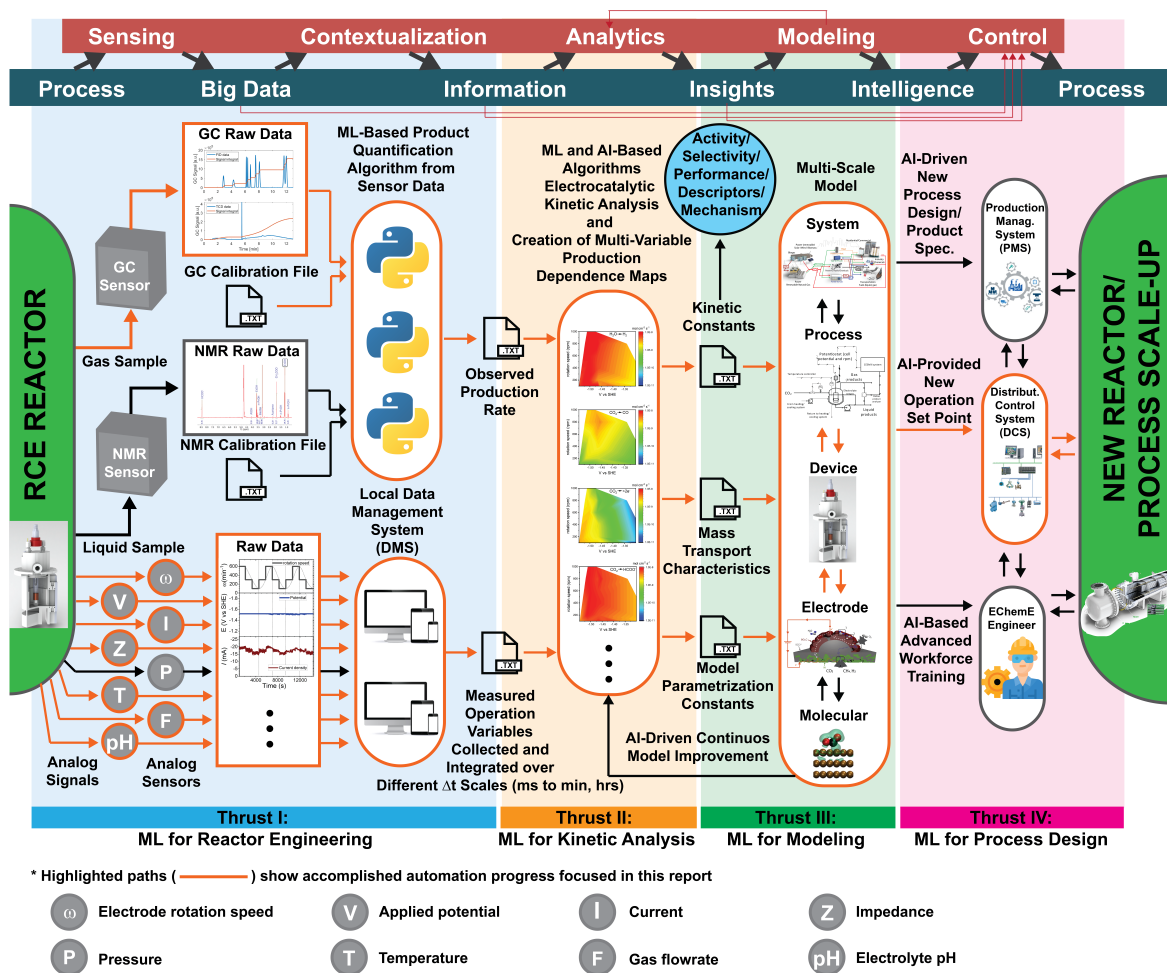
Figure 3.1: Data flow and automation strategy for the experimental setup. The tasks achieved in our research are highlighted in orange.

Figure 3.2: UCLA gas-tight RCE reactor setup.

Figure 3.3 shows the building blocks of smart manufacturing where the interactions between the building blocks are addressed with SM Profiles. Below is a summary of our research findings in mapping the different building blocks described by [31] of the smart manufacturing approach onto our experimental electrochemical reactor setup. Details are presented in greater depth in Chapter 2 and [105].

Sensing is a fundamental building block of smart manufacturing. In our research, a potentiostat is the primary instrument used to sense and record the current and applied potential values in
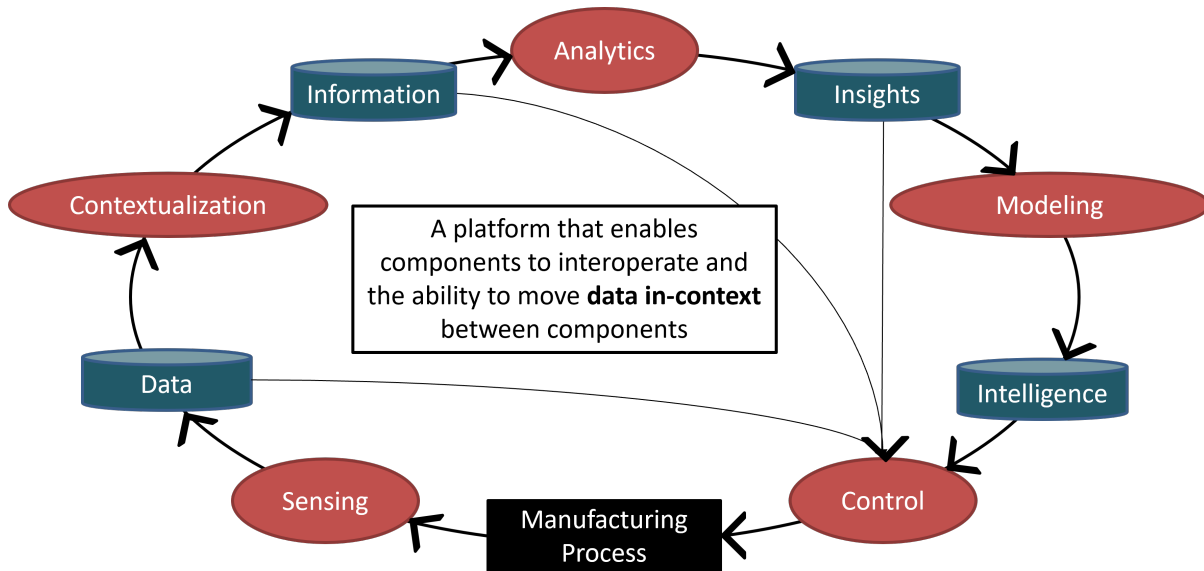
Figure 3.3: Smart manufacturing building blocks.

real-time every second. Although the raw data ingested from the sensors is fundamental for establishing big data, the data must be contextualized in order to be useful. Data contextualization is a process that converts raw data into an interpretable form, which might include organizing the data based on the units and relevant metainformation. An important example for this work is automating the GC code to mathematically processes the raw GC data, which yields gas concentrations upon the comparison of the sensor's responses against GC calibration files generated using standard gases of known concentrations. The raw GC data is simply a collection of electrical signals that are then converted into compositions over the interval of running time (14 minutes). This data is contextualized by eliminating irrelevant intervals and classifying this data as collusion intensity, which becomes available to the automated GC code processing. Thus, the raw sensor data is now meaningful to the operators and is processed to determine the correlations between the parameters. As a result of the correlation analysis, for example, the cumulative integral of the current was

identified as a metric to estimate catalyst degradation in Chapter 2. Thus, the current value was integrated over time and contextualized as a representation of catalyst decay which can now be used to plan catalyst regeneration procedures. In addition, the continuously changing experimental inputs are used to build a dynamic model that includes nonlinear relations between surface potential and current as explained in Chapter 2. The surface potential is a calculated variable that uses current, applied potential, and solution resistance values, which is a better representation of the electrocatalytic driving force compared to the applied potential. The calculated surface potential is recorded with appropriate units that is ready to be imported to the ML model.

Steady-state and dynamic machine learning models were used to generate new insights on the reaction kinetics. Firstly, it was demonstrated in Chapter 2 that the ML models based on GC data can be used to predict the reaction rates. This model was then inserted into a first-principles gas-phased mass balance model to estimate the gas phase ethylene (C_2H_4) concentration. Then, the ability to regulate the gas phase ethylene concentration by manipulating the applied potential was explored with the availability of a dynamic model. Next, the fast-decaying catalyst activity introduces additional nonlinearity and uncertainty into the process, which were addressed by modifying the feedback controller parameters in real-time in process similar to classical controller gain scheduling. Moreover, the process goes through a selectivity shift from the desired ethylene product to the undesired methane product at potentials that are more negative than a threshold potential. This leads to a control strategy in which the target setpoint is approached slowly with a small proportional controller gain, thus delaying the selectivity shift. Finally, a computational method was developed to include GC sensor feedback data to correct the gas phase C_2H_4 estimation.

The insights extracted from the models together with the specific model predictions were

combined with process optimization tools to maximize the process economics and energy savings. The latter was accomplished by an operational steady-state neural network model, which was used to determine the most profitable ethylene concentration setpoint. In order to locate this optimal setpoint, a standard Interior Point Optimizer (IPOPT) tool was used along with electricity costs and chemical sales prices. Finally, the feedback control of the electrochemical reactor was realized using a hybrid (i.e., combined machine learning/first principles) model constructed with open-loop data. A ML estimation-based feedback controller was used in Chapter 2 to control C_2H_4 concentrations by manipulating the applied potential.

Remark 11 *The development of first-principles dynamic models for electrochemical reactors is still in its infancy. Thus, there must be a dependence on data-based modeling for further process exploration and scale-up, as our work has demonstrated. Moreover, it is important to emphasize that working with experimental data has multiple challenges compared to using simulated data. Connectivity problems, uncontrollable variations in experimental conditions, and shifts may cause anomalies and excursions, and should be carefully handled at the data-based model building stage.*

3.3   Advanced Sensors

As process sensors become cheaper and smarter (in the sense of measuring complex process properties), advanced algorithms are needed to be used translate sensor primary measurement into useful information about the process that can potentially be used for modeling and control purposes. A thermocouple is a good example of a sensor frequently used in legacy engineering. The voltage measured by a thermocouple is converted to a temperature unit through a basic algorithm

with no data contextualization. On the other hand, smart sensors are heavily dependent on contextualized data. The richness of the data from the IR cameras mentioned in the work of [93] was used to map the radial and axial temperature distributions of tubular reforming reactors. The cameras were placed around and at the top of the furnace tubes; however, due to the sequence and orientation of the cameras, some of the tubes were not fully visible to cameras. An algorithm was developed to interpolate the temperature values in these areas and to convert the infrared wavelengths into temperature values, which was then used in burner fuel arrangements for a uniform temperature distribution. A similar example of a smart sensor used in our research is the automated GC processing algorithm.

Gas chromatography is an analytical technique that separates a sampled gas mixture into its components that are then quantified. A gas chromatograph sensor contains a long thin column, where the gases travel until they hit a detector. Separation occurs inside columns containing materials that serve as stationary phases while a carrier gas (mobile phase) transfers analytes toward the detectors. Differences in retention times due to different affinities of analyte molecules to the stationary phase allow the separation of gases. Eluted gas molecules are then sent to the detectors where electrical signals are generated based on their respective detection mechanism. The impact and the quantity of gas molecules on the detector are represented with peaks in height and breadth in the GC analysis software. The signals produced from the detectors lead to peaks in the spectra, and as the concentration of the gases increases, the corresponding peak area also increases. The concentrations of each gas species analyzed can be quantified by comparing the area under the peaks to that of a calibration file, which contains signals generated from a calibration using a standard gas consisting of known concentrations of all the gases of interest. The ratio between the

areas of a peak appearing during electrolysis and the calibration peak gives the concentration of the produced gas when multiplied by the known calibration concentration.

The raw data coming from the GC or GC sensors generates peaks; however, without processing the data with a baseline correction, it is not possible to obtain accurate results. Since the area under the peak needs to be calculated via numerical integration, the peaks that do not have baselines on the X axis yield misleading areas. Thus, the baselining is a vital part of the GC data analysis. Most of the proprietary GC software programs create an automatic baseline. However, in most cases, the baseline fails to bring the bases of each peak to the X axis since the algorithms calculate a best fit accounting for the entire measurement sensor data rather than calculating individual regions. It is required to manually select the bases of the peaks and arrange the sensor data baseline accordingly. This process requires continuous monitoring of the GC peaks and prevents its on-line use without monitoring. Thus, a GC measurement costs an ample amount of time and energy for supervision.

Even though GC is the most commonly used gas separation and quantification technique, it is susceptible to human errors. It is possible that the supervisor selects the peaks incorrectly, as the peak legs are decided intuitively without a mathematical method. Moreover, the GC runs must be started or scheduled manually, which slows down the experiment procedure in the long term.

Figure 3.4 illustrates the PeakSimple GC analysis software interface. One of the products of interest here is ethylene and the peak is highlighted in yellow on Figure 3.4. The peak has bases higher than the X axis and it is corrected manually by the supervisor. If the area calculations were made without manual baselining, the underlying area would be much higher than the standard expectations.
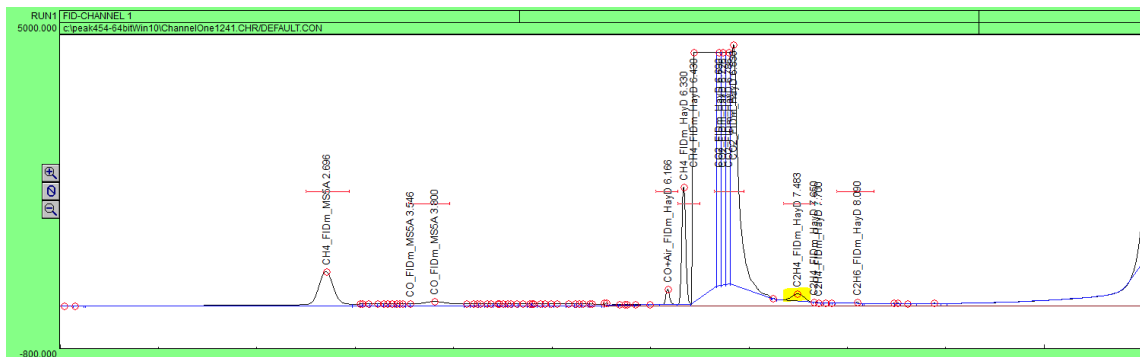
Figure 3.4: Manually corrected C_2H_4 peak baseline. The black line is the raw data coming from the detector, the blue line is the baseline, and the red circles are the peaks identified by the software.

3.3.1 Automated GC Code

Previously, Figure 3.4 shows that the quality of data coming from the GC detector is high; however, in reality, most cases might not be as stable as the black line shown. For example, an increase in column temperature and injection of water from a saturated gas can cause the baseline to drift and the signal curve (black line) might be much higher or inclined than the X axis. This type of raw GC sensor data is described in this article as noisy data, which will require manual interventions to correct the baseline. To create an autonomous intelligent system, the GC code needs to run reliably without human intervention and must be robust to implement corrections to noisy data from the GC sensor. Accordingly, an autonomous GC algorithm is developed with Python programming language that can automatically:

- Start a GC run at the desired times.

- Extract the raw GC data file in ASCII format.

- Baseline the raw GC data.

- Detect when the overall baseline is not accurate enough.

88

- Calculate various baselines for each peak and recalculate the optimal baseline.

- Calculate the area from raw data if the optimal baseline still has bases above X axis.

- Calculate the areas from the calibration files and calculate measured concentrations using calibration area.

- Send data to a database.

The plot generated by the automated GC code is displayed in Figure 3.5. The purple line is the raw data coming from the GC sensor. Compared to the black line in Figure 3.4, the raw data in Figure 3.5 is much more inclined and higher. However, the automated GC code can overcome the burden of stochastic data and calculate the area under the curves. The different colors around the peaks represent more peak-specific techniques to improve accuracy, which will be discussed in detail in the following subsection. Furthermore, the baselining problem can be seen in the raw data from the software in Figure 3.6a, where it is correctly baselined with the automated GC code for the H_2 peak.
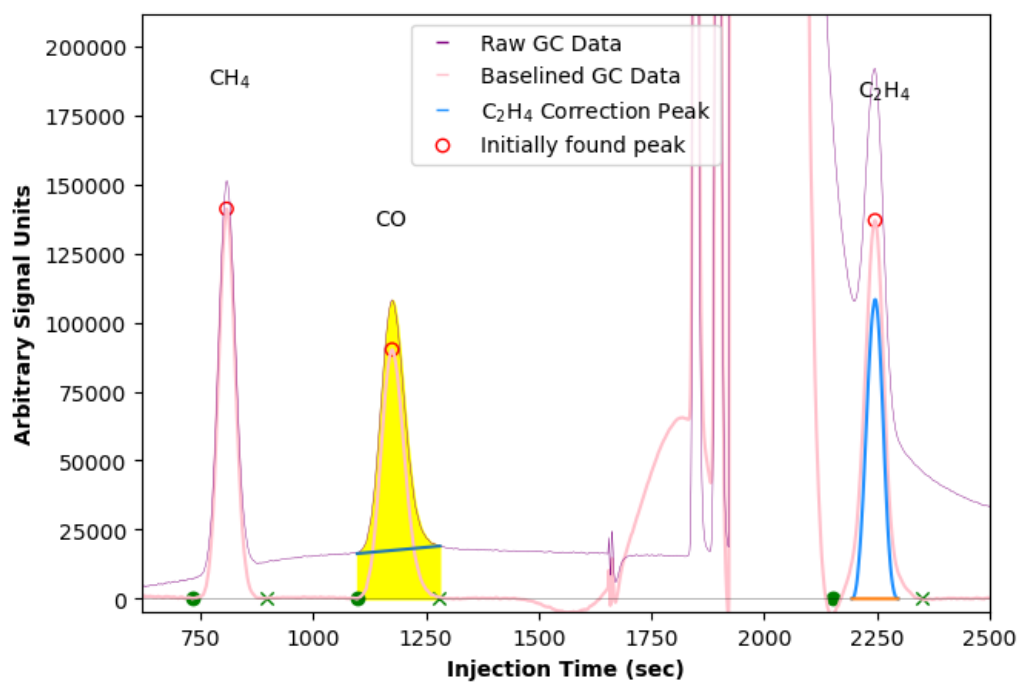
Figure 3.5: Automated GC peak and area calculation example.
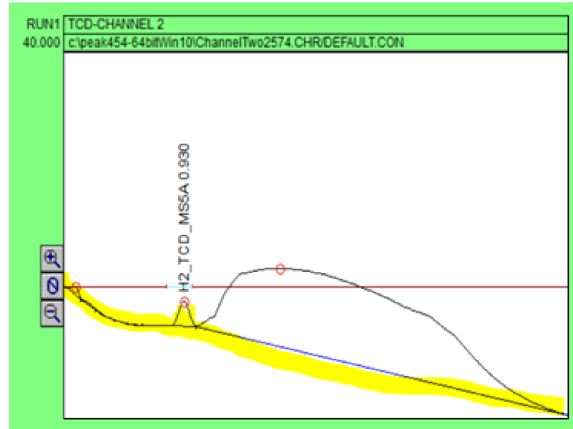
(a) Raw data that needs manual correction shown
on GC Software Interface
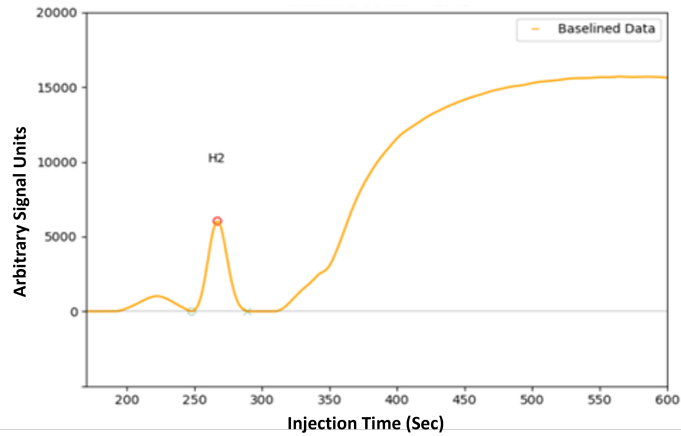
(b) Automated GC Results shows that data is automatically
processed to find accurate peaks

Figure 3.6: Automated GC and manually corrected raw data comparison from TCD channel hy-drogen data.

3.3.2 Automated GC Working Mechanism

There are four gas products coming from our electrochemical CO_2 reduction reactor that must be quantified by GC. The products are CH_4, CO, C_2H_4, which are detected via a Flame Ionization Detector (FID) equipped with a methanizer, and H_2, which is detected via a Thermal Conductivity Detector (TCD). There are two separate channels that are involved in the quantification of concentration and there is one plot generated for each. A comparison of plots of the TCD channel raw data and how it is manually baselined against the processed automated GC results are shown in Figure 3.6. TCD nondestructively senses changes in filament temperature and resistance due to thermal conductivity difference between the analytes and the reference carrier gas, while FID detects ions generated upon pyrolysis of organic analytes that can sensitively detect hydrocarbons, CO and CO_2, when equipped with a methanizer. Data generated from these detectors are contextualized to represent the intensity of the signal when sent to the platform. The code is tuned based on the gas quantification results of 43 open-loop experiments. This tuning process accounts for many various extreme cases and creates a hierarchy for peak detection and integration.

Baselining is implemented based on the asymmetric least squares smoothing suggested by [39]. This method generates the optimal baseline by minimizing the following cost function:

$$S = \sum_i w_i \left(y_i - z_i\right)^2 + \lambda \sum_i \left(\Delta^2 z_i\right)^2 \tag{3.1}$$

where S is the regression cost function, y_i is the signal that should be baselined, z_i is the smooth baseline, and $\Delta^2 z$ is equal to $z_i - 2z_{i-1} + z_{i-2}$. The first summation describes the performance of the fit and the second summation describes the smoothness of the fit. Thus, w_i is a factor related

to the asymmetry and λ is related to smoothness. In our code, the asymmetry parameter is set to a pre-determined value and λ is varied over a range to find the ideal baseline fit. When S is minimized, the corresponding z is the baseline. This is a function of raw data and λ, which is a factor that is observed to work well in the range of $[10^4, 10^9]$ [39].

The pink baseline shown in Figure 3.5 is created based on fixed parameters. Various parameter values for λ are tuned for the calibration peaks and the best performing lambda value is selected as the default value for the pink baseline. However, it does not always give accurate results for our peaks, especially when the sensor data is noisy. In this case, the code isolates the peak vicinity and tries various λ values in these specific excluded regions and selects the baseline with bases closer to zero. This methodology also contributes to the reusability of the code and ensures that the most optimal baseline is selected for the peaks.

The peaks are found with a user-defined function, that checks elevations and declination in the time-series peaks. The peak bases are selected mathematically "peak_prominences" function through the Python library, Scipy. However, the expected peak base length value should be provided to the code for a more accurate peak/base selection. The peak location is provided by the Python command. The algorithm starts extending a horizontal line to both sides of a peak until it intersects the signal of higher peaks. Then, the lowest two signal values within the range of these horizontal lines are the bases of the peak [159].

When the peak detection hierarchy is applied on the methane peak in Figure 3.7, the pink baseline is replaced by the lime peak. The CH_4 raw data in this graph is inclined, thus it is noisy and might be confused for 2 peaks. The code first considers the red circle to be the pinnacle of the peak; however, this is incorrect. The green marks on the X axis are the first predictions for the

93

peak bases that correspond to high positive values on the pink line, thus the code understands that

this peak is questionable. The code adjusts the baseline to conform to the level of inclination of

the data using various baselines that conform to the level of inclination of the data. Ultimately, the

code determines an optimal peak that represents the actual peak for the CH_4 component, which is

represented by the lime-colored peak. It is notable that all remaining peaks have an inherent peak

hierarchy despite the base code being the same. Thus, this code can be adapted for any peak for

any gas product if special cases are accurately identified in the peak detection hierarchy.
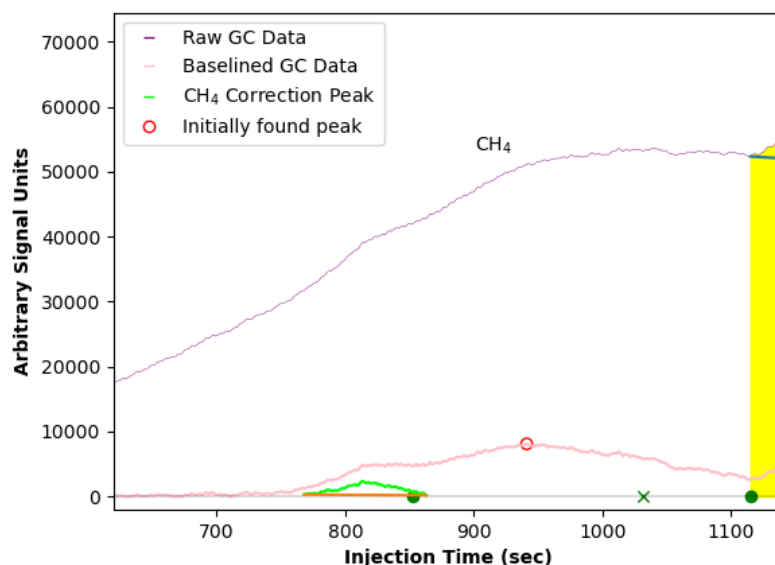
Figure 3.7: Methane peak corrected by a supervisor algorithm. The first found peak is shown with the red circle, however, the corrected peak is shown in lime.

In Figure 3.5, the CO area is highlighted in yellow, which indicates that the code is able to

provide a more robust estimation by manipulating the raw data and is able to find the bases of

the raw data peak to calculate the peak area under the curve that is subtracted by the area under

the green line. This approach typically gives better results only for CO compared to the baseline-

94

corrected area calculation. The correction peaks are shown in lime for CH_4, yellow highlight for CO, and blue for C_2H_4. It is especially difficult to correctly baseline the C_2H_4 peak, since it appears on a line that overlaps with a CO_2 peak, which is the reactant itself. Thus, in the majority of cases, the correction algorithm calculates the C_2H_4 area.

3.4 The Role of the CESMII Smart Manufacturing Innovation Platform (SMIP) in Electrochemical Operation Research

As the use of data is proven to increase the profitability and efficiency, more effort is put into developing model-based software solutions. One interesting framework is the Parametric Optimization and Control (PAROC) Platform developed by [130]. PAROC is a computational tool for creating high-fidelity models for control and optimization strategies. [165] have built a platform service that uses an optimization and compatibility engine that provides decision-making support between the customer and suppliers based on the equipment of the supplier such as raw material and energy consumption options. The use case scenario of this work supports the selection of distribution for most compatible resources from discrete suppliers for a metal cutting process by considering energy efficiency, machine geometrical compatibility, and precise number of metal sheets thus optimizing the supplier data available on the platform.

The Smart Manufacturing platform is standard based infrasturcture for connection, ingestion and contextualization of data to be used for building applications. The platform makes contextualized data available for machines and process components, line operations and process and plant operations between the chain starting from machines, process, and plant enterprise to the supply

chain [38]. Thus, it is the infrastructure of information and operational technologies (IT and OT) of a process that enables the implementation and collaboration of auxiliary smart applications. The use of terminology OT here comprises all the hardware and software that is used to monitor and control process or events related to industrial equipment. Moreover, companies can exchange data and use apps developed by others [38, 31]. In addition to accessibility of data and applications from various sources, the SM Platforms also provide interoperability among different vendor equipment through the use of SM Profile. For instance, SMIP is an integral part of the digitization of the electrochemical reactor. The data collection, contextualization, and transmission of the electrochemical reactor is conducted through a Laboratory Virtual Instrument Engineering Workbench (LabVIEW) application running on an edge device that controls the operation of the electrochemical reactor. There are three major components that make up an operational SMIP:

1. An edge device that can collect, contextualize, and transmit the data to the core services.

2. The platform core service that receives, stores and exposes data to the application developers and services.

3. Integrated applications that can consume the data to build data-driven models or workflows that can improve manufacturing operations.

The Hypertext Transfer Protocol Secure (HTTPS) transfer of reactor operational data to the SMIP core services is performed through GraphQL in real time. GraphQL commands are similar to the one used in Structured Query Languages (SQL) except that these commands are initiated over a Hypertext Transfer Protocol (HTTP) connection with a web API. GraphQL can perform typical

96

CRUD (Create, Read, Update, Delete) queries similar to a REST command operation except that the program is constrained to the transmission of necessary data due to the limit of the network bandwidth [60]. With the accessibility of various programs, [15] have conducted a social experiment on participants to determine which program was more beneficial for their use when asked to conduct several data transfer tasks. Ultimately, the majority of participants reported the GraphQL to be superior to REST because of its user-friendly Application Programming Interface (API) to query data. The back-end storage of the CESMII SMIP is provided by a PostgreSQL database. The operational data that is stored on this database is archived using the timestamp at which the data is transmitted. The data that requires no timestamp such as equipment model number and serial number are stored as config data, which are stored as attributes inside the SM profile. The attributes contain data name, data type (float, int, string, etc.) and data units (seconds, volts, amps, etc.). SM Profiles are structured as data-in-context information that is specific to an equipment, domain or platform. The idea is that all end users of the data from the same type of equipment can expect to receive the same information model. In many ways, the SM Profile is an extension to the Open Process Communications Unified Architecture (OPC UA) information model that specifies the interoperability standards for structured data communication among producers and consumers of data regardless of who manufactured that equipment. This guarantees interoperability between data producers and consumers, regardless of the manufacturer of the equipment or the operating system (OS) where the data is generated. Since GraphQL mutation commands can overwrite the existing data on SMIP, a role-based authorization process is employed in determining which user can update the data and which user can read the data through GraphQL query commands. SMIP has an Integrated Development Environment (IDE) for Python, PHP, and SQL that can be used in

developing a machine learning based data-driven model using the data that is already stored inside the SMIP. Additionally, SMIP offers data visualization tools, such as a trend analyzer, to visualize time series data or to compute data correlations. In addition to the GraphQL mutation, CESMII SMIP also offers custom gateway connectors that facilitate the high-speed entering of data into the SM Platform. Currently, operational databases and historians such as OSI PI and Wonderware historian as well as live data sources such as OPC DA (Data Access) and OPC UA are supported. The SMIP architecture used in this electrochemical reactor research is shown in Figure 3.8. SMIP has standardized on GraphQL as the protocol API for entering the data. The SMIP has an extensive library of SM profiles, which can be easily exported along with all previously set up features and contextualization and can be accessed from an SM Marketplace. Finally, apps developed by the other SMIP users can be accessed and combined for use in your manufacturing operation. Several of these SMIP features used in this research were used to advantage in this research and are discussed in the following sections.
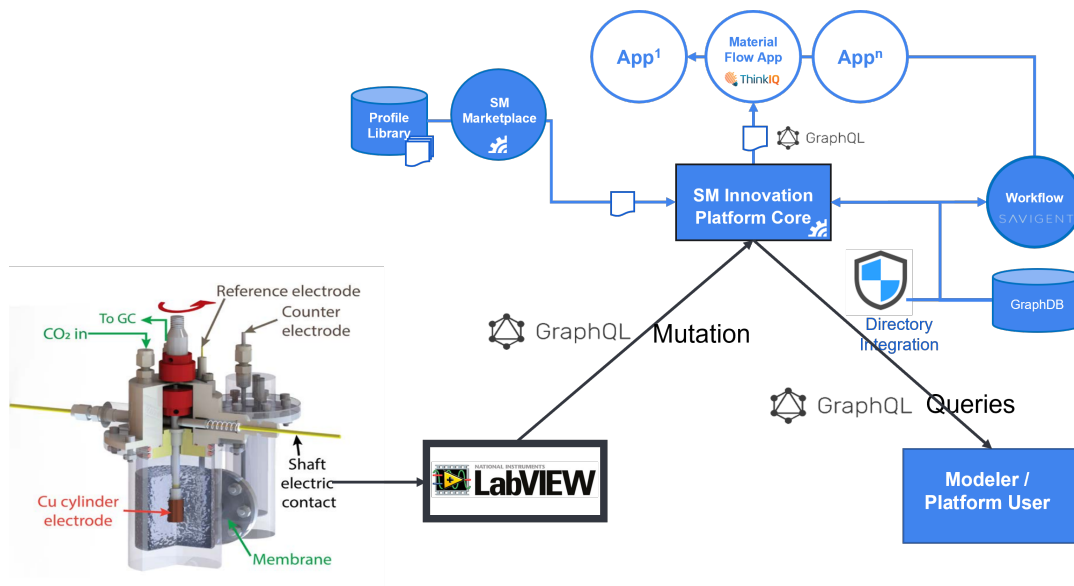
Figure 3.8: SMIP architecture.

Sensors collect data from the reactor through a LabVIEW interface. Sensor data is transmitted each second to the SMIP securely through GraphQL commands. In GraphQL terminology, a query that alters the data at the URL endpoint is called mutation. A mutation is a form of query where GraphQL commands create, update or delete data. Use of a timestamp along with the data value will make sure that data is not overwritten, as mutation commands are not reversible. The associated timestamp will also help users who want to query the data select only the data belonging to the desired time intervals. GraphQL API commands may be issued in several programming languages such as Python, JavaScript, Curl, etc. We have chosen Python script for the easy implementation with LabVIEW interface through a LabVIEW code mentioned in detail in Section 3.4.2. The current version of SMIP allows any authorized users to readily select and download the data using the GraphQL query commands to their local compute environment to more easily evaluate different modeling approaches as well as different data sets. Once a satisfactory modeling approach is worked out locally, the model can be implemented directly within the SMIP. The platform provides IDE in several languages such as PhP, Python to build the model. We used the Python IDE on the platform because that is what we are using locally to develop our ML model. The default library already contains commonly used ML modeling tools, but it is also designed for users to contribute to and build the tool base it contains to build ML models with new ML tools. The choice to download the data to locally or to use the platform resource is entirely dependent on the network bandwidth and computational resource. The computational resource power on the platform is very limited and there are no co-processors such as GPU to accelerate model building. So, for initial modeling purposes, using the local resource makes better sense for most users. The platform also provides tools to monitor the real-time data flow through trend charts and process layout diagrams that show

a hierarchical organization of reactor equipment and display the data values for each equipment piece.

3.4.1 Reusable Profile for Electrochemical Reactors

Every day, engineers are building models to improve the efficiency of manufacturing, but the size of the data and lack of commonly agreeable formats have become a major impedance in collaborating with other model builders. Before the implementation of SMIP, we are not aware of any other common platform or medium where data producers and consumers can easily exchange information in a common format. Through the use of SM Profiles described by [31] SMIP has exposed a common information model for commonly used equipment that producers of data can use to deliver the data to consumers to build data-driven models. The SM profile is available not only for major equipment but also for the sensors, actuators or any device used in the operational technology. Currently, SMIP does not have all the SM Profiles, but projects like ours are contributing new SM Profiles every day to SMIP to enrich its portfolio.

Figure 3.9: Hierarchical equipment profile interface on SMIP for electrochemical reactor.

Every piece of equipment in SMIP has its own Smart Manufacturing profile so that producers

and consumers of the data have a clear understanding of the expected data structure and contents. SM Profiles can be created on the SMIP itself or can be imported (crowdsourcing). A key objective of the SM profiles is reusability for similar equipment and equipment types. Once a profile is built, the manufacturers/researchers can use the previously built profiles to easily start data collection and modeling. The SM profile also adopts many of the concepts from object oriented programming with features such as inheritance and data abstraction.

The profile built on the SMIP for the Electrochemical Reduction Reactor is shown in Figure 3.9. The "CO_2 Reduction Reactor" is the top level profile and the "Gas Chromatograph," "Potentiostat," and "Modeling" are the subprofiles. Each operational data is sent to the SMIP and stored in the corresponding attribute in the profile and is organized in a timestamped historically contextualized format. Each attribute in a profile, for example, temperature from the sensor at a certain point in time is stored in the appropriate data endpoints. Each attribute in a profile is assigned a tag ID number automatically to define the data storage location during its creation. Each attribute also has the relevant datatype information (float, int, string, etc.) and the units (Amps, inches, etc.). The tag ID and timestamp are the two required parameters to store or retrieve the historized data from the SMIP. The modelers, who are the consumers of the data, can use GraphQL queries to discover the equipment and all the associated tag ID of each attribute independently and download only the desired time interval data from the SMIP.

When a new user wants to build a profile for an electrochemical CO_2 reduction setup, the user can take the one constructed by us as a basis. This would also bring the sub-equipment such as the potentiostat, GC, and rotation unit. If the user is using a Fourier transform infrared spectroscopy (FTIR) sensor that can quantify high-volume gas products on a per-second basis, the GC

101

sub-profile can be replaced by a FTIR sub-profile. All sub-equipment would include the features created in this project, and new users can reuse the profile in general to start data transfer to the platform without extra effort to define system components and data storage end points. The only thing that will change will be the tag ID because a new tag ID will be generated for corresponding attributes in the new equipment. When the proprietary data is removed from the hierarchical equipment model, this equipment model becomes a profile. Additionally, this profile may include an automated GC algorithm feature. It would be a time-consuming process for engineers to develop an automated GC code from scratch and embed it into an experimental system operation ready to be used in a process control scheme. Instead, SMIP renders this phase fast, reusable, and user-friendly.

Another example of reusability is the infrared (IR) camera model used in [93], which was mentioned in Section 3.3. When the steam methane reforming specific data of the plant is removed from this IR camera model, the remaining model can be used in different contexts to measure temperature distributions. Similarly, in our project, an automated gas chromatograph code, mentioned in Section 3.3.1, was developed to quantify H_2, CO, CH_4, and C_2H_4 gases in ppm. However, this code can be reused when, for example, specific C_2H_4 inputs are replaced with another gas input, such as where peaks are expected to appear. Moreover, this code processes the ASCII format, which can be replaced by another GC output file format. Similarly, the automated GC code triggers PeakSimple software to start the GC run, which can be adjusted for other GC vendor software.

In addition to the reusability feature, SM Profiles offer easy real-time/offline monitoring of the process data, process reports, and trend analyzer. One of the key parameters in the electrochemical reactor research is the real-time current. Figure 3.10 depicts the trend analyzer interface

for real-time electric current. It is possible to change the data of different equipment or display them on the same interface. In traditional manufacturing, all process equipment operates on their vendor-specific software, and the user needs to switch between many different vendor-specific software tools to see the real-time data. Moreover, it would not be possible to see the data coming from different vendor sensors on the same plot. Smart manufacturing aims to eliminate this interoperability issue by bringing vendor-dependent sensors together in a common medium, such as SMIP.

Figure 3.10: Trend visualization tool on SMIP demonstrated on real-time current data.

3.4.2 Data Flow

As information technologies are increasingly used in data-based services, the importance of cloud-based systems has increased. The SMIP has been built on infrastructure cloud services and uses the PostgreSQL database, which is currently hosted on the MS Azure cloud platform. SMIP's cloud-based database enables cybersecure data storage. It is easy to mutate (create, update, delete) the data or query (read) the data in real-time using GraphQL commands. Sensor data are transferred through GraphQL commands from the local machine (edge device) to the SMIP. Data is stored at endpoints described as tags or attributes, while the equipment profiles were being generated. For example, when the potentiostat sub-equipment was built in Figure 3.9 and attributes such as electric current were created, a tag ID was assigned to this feature. Thus, any electric current data mutated to PostgreSQL database can be reached and viewed using this tag. A timestamp is used to locate the data in the database rather than sequential numbers. For real-time data transfer, timestamps reflect the time of data collection. For this data mutation process, users need security tokens called Bearer tokens, which can be obtained from the SMIP website. A Bearer token is a HTTPS authentication and contains a long random string which can be understood by machines and gives authentication to the user for data transfer. They are also short-lived and need to be renewed from time to time.

Many traditional databases require users to generate data containers or tables before starting data collection. Moreover, these data containers need to be configured and connected to specified servers. For example, in order to create a database, the creator of the database first needs to compose a database schema that indicates how the elements make up the database, how they are related to one another including all the tables and keys needed to store and retrieve the data. If

those data tables are to be queried by an external application such as Python, an administrator must make the relevant settings available. Although this approach has been shown to be useful in many industrial applications, it requires domain specific knowledge. The use of SMIP simplifies this process, as users only need to know the SMIP endpoint URL, and have the credentials to create a Bearer token for secure transfer of the data and transmit the data into the previously created attributes corresponding to the equipment. The consumer of the data also just needs to know the SMIP endpoint URL and appropriate credentials for the creation of a Bearer token, and it also uses GraphQL queries to download the data. Since data is stored at endpoints rather than in tables locally, there is no need to spend time to separate the necessary columns from data tables. Another advantage of the end-point data storage is that it is easy to store features that have different sampling periods. For example, the electric current from the electrochemical reactor is recorded on a per-second basis, while the GC measurements of different gases are recorded at every 20 minutes, and all the data sent to the platform is contextualized. Therefore, there is no need to create separate data storage tables for these features.

The importance of real-time data collection was emphasized in the previous sections. However, for a new user who has no knowledge of Python or GraphQL, there is going to be a learning phase to move the generated data to SMIP's database since this process is different from real-time data mutation. To this end, we have developed a script and a Django-based web interface to select the relevant columns from the data sheets and send them to the platform. The interface is shown in Figure 3.11. This has been particularly useful in our project when we uploaded the previous (legacy) experimental data obtained before the integration with the SMIP was pursued. This feature was planned to be easy, specifically for equipment operators who do not have programming

expertise.

Figure 3.11: The interface of data upload tool. The user needs to enter SMIP credentials, path to the spreadsheet file that will be uploaded to the platform, and columns/rows within the spreadsheet to upload to the platform.

3.4.3 Process Equipment and Data Connectivity

In smart manufacturing, combining process equipment with a common digital interface is a good starting point. This task can be achieved through the use of connectors. Most of the time, sensors and local machines are written in different protocols. Connectors behave like translators, enabling communication between sensors and machines, thus making sensor data available. The SMIP offers a variety of connectors that would enable sensors to transfer data to the platform, such as a programming environment that sends the data to the cloud. LabVIEW is a coding environment that is used for automating test beds for measurement and control. It has numerous drivers that are compatible with commonly used experimental or industrial equipment. In addition, some software development kits (SDK) have been developed to make novel and complex experimental equipment compatible with LabVIEW. Moreover, LabVIEW can communicate with external programming scripts, such as those written in Python or Matlab. Thus, LabVIEW appears as an

attractive alternative for experimentalists for data collection, process automation and control.

In our project, a Metrohm Autolab Model 302N is connected to LabVIEW through the Autolab Software Development Kit 1.10 [7]. Without the SDK, the potentiostat runs on the vendor software Autolab NOVA. The experimental actions called procedures are predefined in NOVA and the experiment runs accordingly. Also, since it is very difficult for a feedback controller to change the input parameters in real-time, thus, using the SDK enables experimental researchers to go beyond vendor-specific software. However, the gas chromatograph does not have a driver available in LabVIEW. To overcome this issue, an automated Python script was written to externally trigger the GC measurements at predefined times by opening the PeakSimple software and initiating the run. When the GC run is finished, the Python code sends the generated raw data to SMIP to be processed and quantified. Then, the produced results are queried by the LabVIEW interface. The LabVIEW interface also shows a plot of the processed GC data and the relevant peaks.

The rotation unit also does not have drivers installed in LabVIEW, and it is connected by a Compact Reconfigurable Input Output (CompactRIO) system, a National Instruments product that enables engineers to connect input/output modules without drivers. Even though the rotation speed is kept constant throughout the experiments, CompactRio can adjust the rotation speed in real time. Finally, the mass-flow controllers (MFCs) are connected to the LabVIEW interface via a VISA node, a function that enables pinhead cables to send commands specified in the user manuals of the device. Thus, MFCs can be set to a specific flowrate at the beginning of the experiment. A portion of the LabVIEW interface is shown in Figure 3.12.

The LabVIEW interface can control and acquire real-time data from the potentiostat and the gas chromatograph, and it can communicate with the platform for real-time mutation and
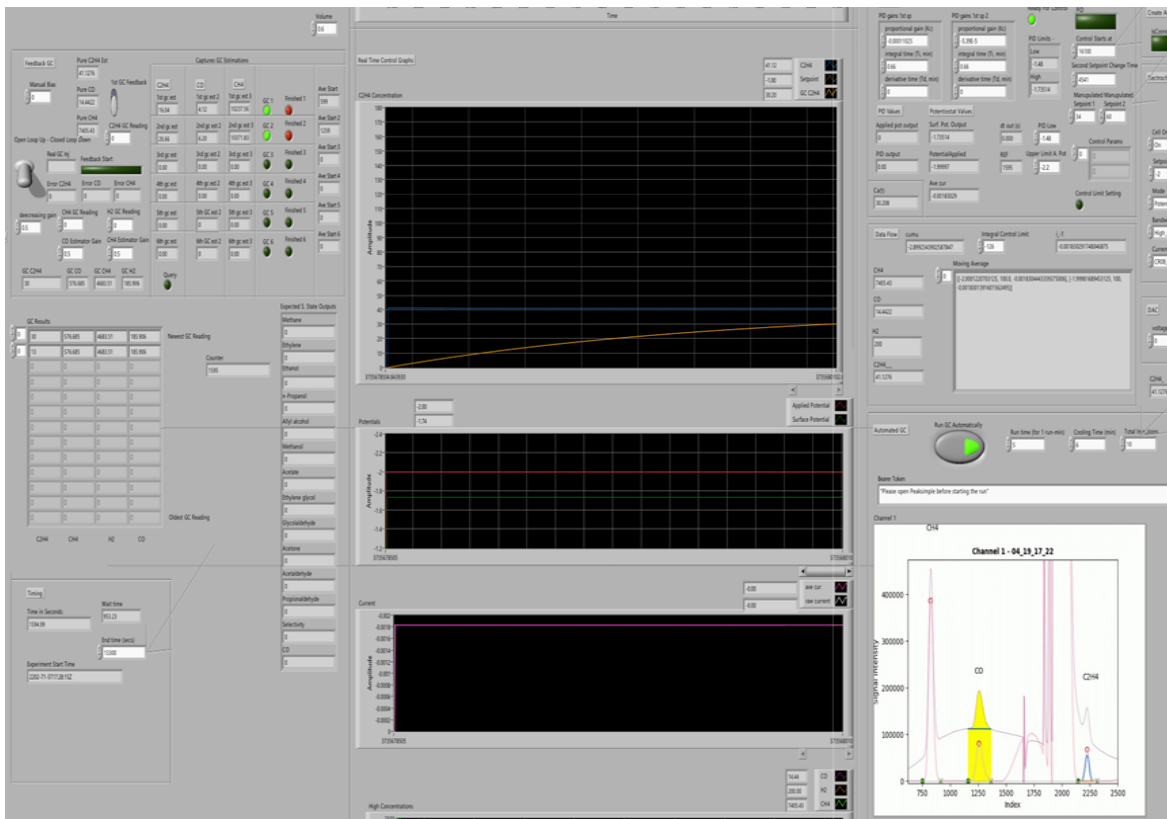
Figure 3.12: LabVIEW interface and representative real-time data plots.

query through GraphQL. LabVIEW has a feedback control feature which is used for the control of

the gas-phase ethylene product concentration by manipulating the applied potential. A real-time

change of applied potential is made by the potentiostat based on the feedback value calculated

by a proportional-integral (PI) feedback controller (other control methods can also be used in this

framework) on LabVIEW. In order to send data to the platform, a LabVIEW script is developed,

as shown in Figure 3.13.

The script on the right side is written in json format and is sent to `https://uc.cesmii.`

`net/graphql` for query. The json script needs the timestamp, value, tag, start, and end times.

Time-stamps are obtained on LabVIEW in real time (the middle functions convert the time-stamps
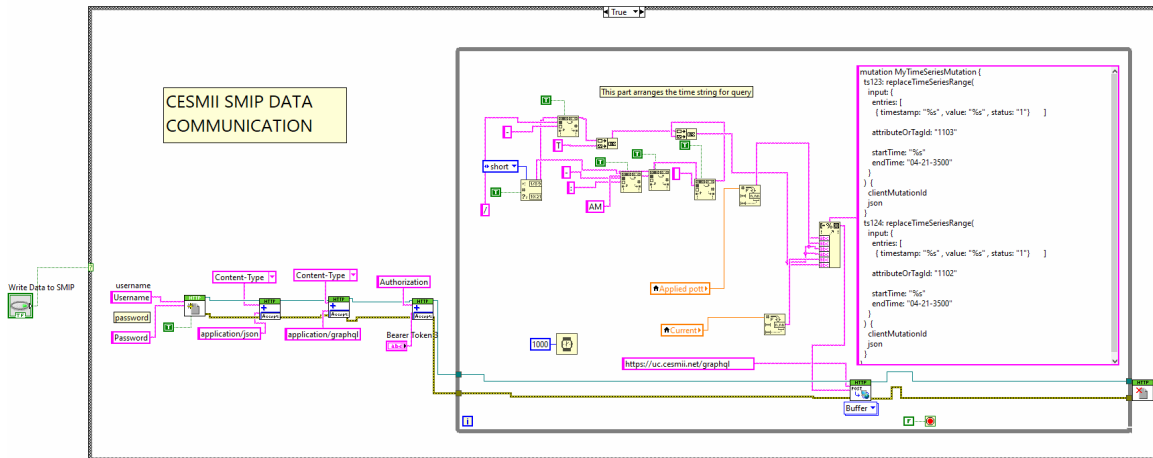
Figure 3.13: Data query script on LabVIEW.

to the required string format), where the values are obtained from sensors and the tags are prede-fined depending on the profiles mentioned in Section 3.4.1. The start time can be the same as the timestamp so that this process is established in real-time and does not intersect with previous data. The end time is defined in a distant feature to prevent a timing conflict. HTTP client nodes are used from the Data Communication - Protocols section in the LabVIEW functions palette. This is similar to the Python '"Requests" library. A bearer token, generated by the SMIP, is entered on the LabVIEW interface for authorization before the experiment starts. The open handle function defines the SMIP username and password. This script sends data to the SMIP on a per second basis.

3.5 Virtualization

Up to this point, the major functions that the SMIP provides to integrate advanced computing and data science technologies into industrial process modeling and control systems were demon-

strated. However, there are two limitations of using the SMIP in daily process operation. Firstly, at this point, the SMIP cannot provide enough free computational power that can perform very expensive computing tasks (e.g., training ML models and/or solving complicated optimization problems, particularly the ones arising in the context of real-time model predictive control) to the users. Therefore, users will need to download tools from the SMIP and run them on their local computer or in a third-party cloud computing service, such as Amazon Web Services (AWS). Second, the client software to access the SMIP is in development, so users need good programming skills to interact with the SMIP. This section introduces the application of Docker technology to bypass these limitations. Using the Docker to communicate with the SMIP as a new type of computer virtualization method can be applied to simplify programming tasks in daily research, especially for setting up virtual working environments and programs management.

3.5.1 Docker Overview

Docker is an open source application for computer virtualization at operating system (OS) level, which is proposed as a lightweight alternative of the traditional virtual machine technology. The Docker application is composed by four main components: Docker Client and Server, Docker Images, Docker Registries, and Docker Containers [136]. When using Docker, users give command lines to the Docker Client, which then converts those commands into a request form and sends it to the Docker Server. The Docker Server can be understood as the background script that is running behind the screen. Usually, the Docker Client and Server are installed on the same machine, but they can also be installed separately.

Images and containers are the basis of Docker virtualization. Specifically, a Docker image is

a read-only file that contains OS (e.g., Ubuntu for Linux-based application), libraries (e.g., Tensor-Flow, Numpy, Pandas for machine learning programs) and tools (e.g., Jupyter Notebook), which can be shared by different containers. Containers can be understood as writable layers built on top of images so that users can make changes and run applications on containers, such as developing a new program or generating data by running applications built in the images. In short, by using Docker, users can create and run isolated applications with various virtual OS on the same machine.

3.5.2 Docker-SMIP Synchronization

We consider Docker as the appropriate method to interact with the SMIP, because it can build images from the base OS, which is analogous to setting up a new computer that provides a potential to design any suitable function if it is programmed properly. On the other hand, Docker images are lightweight files, which can be easily packed and delivered to other users by standard uploading and downloading processes. Therefore, the developed tools can be packed as Docker images and provided on the SMIP. Then, users can download those images and use them on their local machines provided they have enough computational power. In addition to that, users can also modify the tools and programs with respect to specific task requirements and save the updated Docker images on the SMIP for their own version management. Moreover, Docker containers can be mounted to the local machine with the official command lines, which allows efficient communication between the host and the container. This workflow is illustrated in Figure 3.14.

In this work, we build two docker images to enhance the interaction with SMIP. Specifically, we build an interface for the data upload script to simplify the upload process and an optimizer to
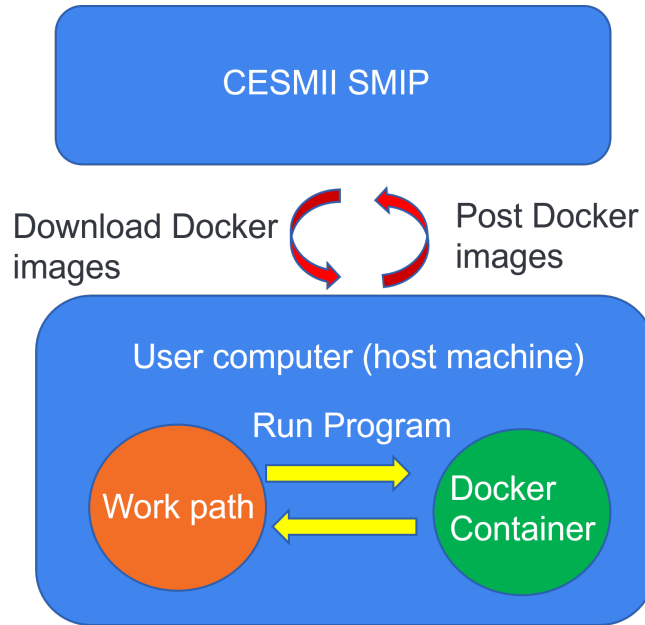
Figure 3.14: Docker-SMIP synchronization.

solve an optimization problem by using a machine learning neural network model that we developed using process experimental data. These tools can be found on the SMIP. The building and sharing of the Docker image can be achieved by running Docker command lines with the command line interpreter (e.g., CMD for windows and terminal for Mac) of the local computer. The development of Docker image usually starts from getting a baseline image that contains only the OS from the official server. This can be done by directly pulling the whole image or running a docker file.

After obtaining the baseline image, by running it, a Docker container with only the OS will be built automatically and then users can develop any functions from scratch in a fashion similar to setting up a new computer. We used the baseline image with Linux OS to build both tools because the Linux system provides the highest level of authorization to the users compared to other commercial OS such as Windows. However, Linux OS does not provide a developed user

interface like Windows OS. It will be an additional task for the developers to build a user interface

that can interact with the host machine. For instance, we use the Jupyter notebook interface for

the optimizer and build an HTML page (shown as Figure 3.11) to drive the upload script via the

Python Django website framework. Furthermore, we make a program for the containers to run the

interface on certain network port when they are turned on. Subsequently, by mounting the pre-

defined network ports to the host machine, these website-based interfaces can be run through any

browser on the host machine. Finally, we connect the built containers to images and upload them

to the SMIP for sharing.

Remark 12 *The optimizer is developed based on the IPOPT, which is an open-source software*

package for nonlinear optimization provided by the COIN-OR Foundation [186, 187, 185]. A

python library, pyipopt, developed by Eric Xu is used to connect python scripts to IPOPT. Instruc-

tion for installing IPOPT and PyIPOPT can be found on the official website of IPOPT and Eric's

Github page (`https://github.com/xuy/pyipopt`), respectively.

Remark 13 *The data upload interface is built using the Python Django website framework. Specif-*

ically, HTML works as the front-end of a website to collect information from the user and return

the collected information for further analysis processes. For this tool, the HTML is requesting the

path to access the data list from the user, and then the path will be passed to the aforementioned

data upload script to upload the data. In this way, users do not need to access and modify the

data upload script directly, which makes the data upload process easier for users without extensive

coding experience.

Remark 14 *In addition to creating tools for the SMIP, Docker can be used for other management*

tasks. For example, code and program dependencies can be packed by a researcher into Docker images, which can then be used to reproduce their working environment effectively. Besides that, Docker images can be easily shared between team members and can be used for version management.

3.6 Electrochemical Reactor Modeling Using SMIP

Dynamic models, whether first principles-based or data-based, are central to process control applications. First principle models are useful in providing process insights and describing process behavior [216]. However, there are many cases where the first principle models are not available or require high computational power. If enough data is available, data-driven models appear as a feasible alternative. Data-driven models typically do not generalize as well as first principle models to operational regimes outside of the training data set. If there is available first-principles knowledge of the process, a hybrid model can be built by inserting a data-driven model into a first principle model equation. One early example of this approach can be found in [37]. In this work, the product size distribution of a polymer process was modeled. However, even though there are available first-principles size distribution models, inherent complexities of particle processes might render the use of first-principles insufficient. Thus, a data-based partial least squares model was used to model the unknown process components and this model was used along with a first-principles model for real-time control. Our research embraces a similar approach for electrochemical reaction rate modeling. The electrochemical reduction of CO_2 on a flat copper catalyst is subject to production rate variations due to the unpredictable nature of the inherent initial catalyst activity

and the deactivation of the catalyst. Also, an exact first-principles model of these reactions is not available. Thus, the reaction rates were modeled with a polynomial kernel support vector regression model based on the available GC data. In Chapter 2, this reaction rate model was inserted into a first-principles dynamic gas-phase species concentration model for a better representation of the gas phase product concentrations for real-time control. Since this process captures some catalyst-related uncertainties, steady state modeling of this process utilized most likelihood estimation methods to build an artificial neural network (ANN) for production rate estimation based on the available experimental data and the experimental data standard deviations. Based on this model, the most energy-efficient set-points were calculated with IPOPT for real-time control in Chapter 2 and [105]. The ease of data transmission and handling that the SMIP provides aided significantly our data-based modeling and optimization efforts.

Chapter 4

Machine Learning-Based Ethylene and Carbon Monoxide Estimation, Real-Time Optimization, and Multivariable Feedback Control of an Experimental Electrochemical Reactor

4.1 Introduction

The percentage of renewable energy in the electricity grid has increased as decarbonization efforts have gained momentum against the detrimental effects of global warming. With current advances and the increasing popularity of sustainability, falling cost and increasing availability of renewable electricity generation, electrochemical methods have become an attractive alternative for transforming CO_2 gas into organic chemicals and synthetic fuels [33]. However, the overall

116

reaction mechanisms of this transformation have not been fully understood and limited efforts have been made to build dynamic models to understand and control this electrochemical process. Moreover, the applications of CO_2 reduction (CO_2R) have not gone beyond the bench scale (e.g., [73]). The major bottleneck of the industrial implementation of electrochemical CO_2 reduction technology is the deconvolution of intrinsic kinetics from mass, heat, and charge transport effects, which has prevented the development of accurate reaction mechanisms [75]. To further explore the fundamentals of electrochemical CO_2 reduction, a gastight rotating cylinder electrode (RCE) cell was recently developed, which can decouple the effects of mass transfer from surface reaction kinetics [73]. This novel electrochemical reactor shown in Figure 4.1 has demonstrated that mass transport phenomena and intrinsic reaction kinetics can independently affect the productivity and selectivity of electrochemical CO_2R, which implies the potential to control the product distribution of the reaction by manipulating multiple inputs. To understand the hydrodynamics effects on the very thin (μm scale) catalyst surface boundary layer, which is ultimately crucial, [147] simulated the gastight RCE reactor using computational fluid dynamics (CFD) software. Even though there are such endeavors to capture the mechanism of CO_2R in RCE cells in detail, there is no dynamic model available yet.

In the absence of steady state or dynamic process models such as first principles-based models that rely on known physical relations, it is possible to build models using data-driven approaches, such as gray box or black box models that give an output for a corresponding input without exposing correlations. The involvement of machine learning (ML) models in electrolyzers has attracted attention over the last decade due to their capability to approximate nonlinearities with no prior physical information of the system [25]. The use of ML in electrochemistry has recently received

117
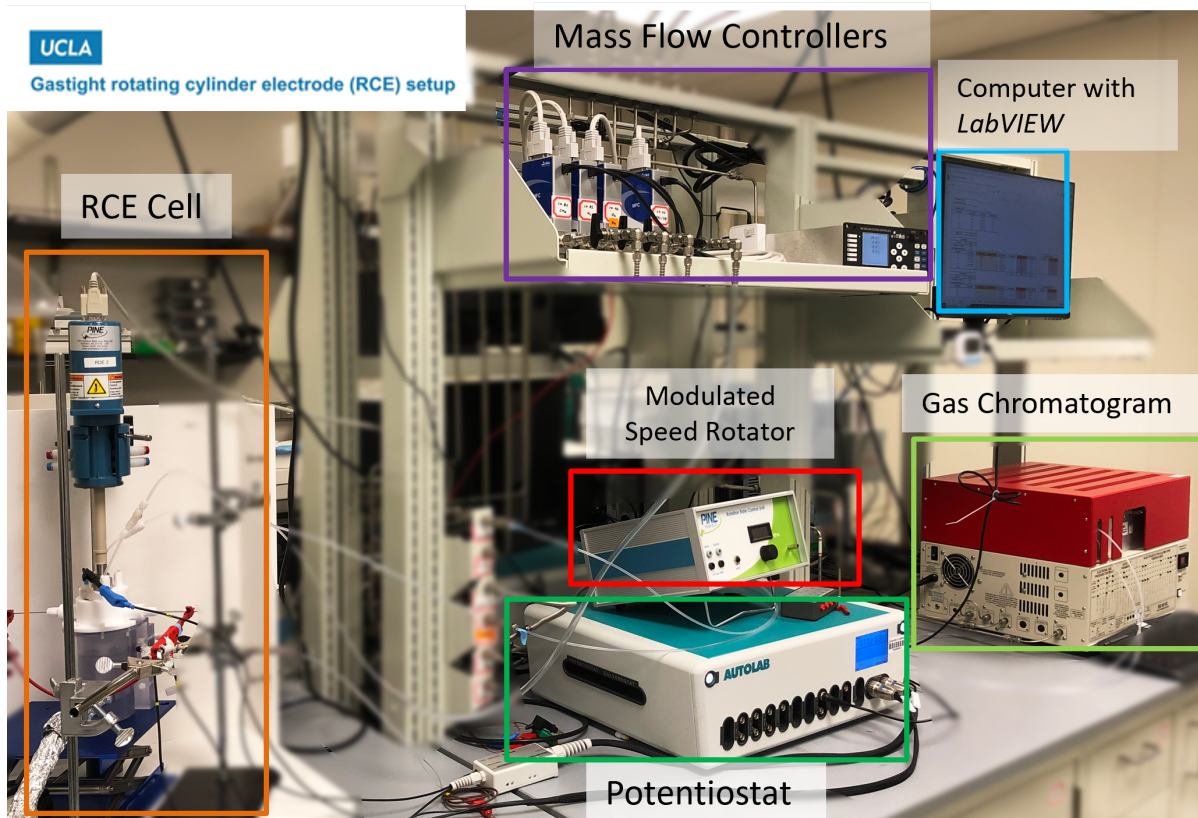
Figure 4.1: UCLA gastight RCE reactor setup.

attention and has been used in predicting the next generation of catalysts without earmarking major

budgets and time for experiments with different material combinations. For instance, [176] used

a neural network trained with X-ray absorption fine structure spectroscopy data to reproduce the

rate and time based structural changes of the catalyst under CO_2 reduction. In another work, [21]

built a catalyst database using density functional theory (DTF) simulations and used these data to

build an extreme gradient boosting (XGBoost) regression model to predict the change in Gibbs

free energies in CO adsorption to find the most feasible CO_2 reduction electrocatalyst among more

than 1000 combinations of metals and nonmetals. However, there have been limited efforts to dy-

namically model the electrochemical reactions and advanced ML methods such as recurrent neural

networks (RNN) should be used in the electrolyzer context to capture time-dependent process relations.

The use of RNNs has been becoming widely common as they are very promising for leveraging the process data for various applications [70]. RNNs have been used successfully in modeling various processes, and they can be very efficient in modeling the dynamic behavior of electrochemical reactions. The ability of RNNs to learn time dependencies makes this approach an alternative to first-principle models, as they can capture the trends in data emerging from the behavior of the process with respect to variation in inputs. This enables RNNs to be used in process control tasks; especially highly nonlinear aspects can be learned due to the nonlinear activation functions in the hidden layers of the neural network structure [206]. This would also be extremely valuable to control a process that does not have a dynamic first-principle model. [22] is one of the early examples of using neural networks before RNN architectures such as long short-term memory (LSTM) became widely popular to model dynamic processes with long and variable dead times. The study successfully models a pH neutralization process and uses internal recurrent neural networks (IRN) with variable dead times, which feed back the calculation from the hidden node as an input, behaving like a one time step delay. [175] modeled the dynamic behavior of a batch methyl methacrylate (MMA) polymerization reactor with a hybrid stack of RNN models. Using this model, an effective feedback control scheme was implemented to regulate the temperature, and in turn, key process variables like monomer conversion. In general, ML models that can represent transient behavior can be used for process control system design and implementation. In this direction, [143] summarized how to incorporate various ML models into a model predictive controller (MPC). For example, [207] introduced Monte-Carlo dropout method to the LSTM training case to improve

119

the modeling performance. In addition to that, the co-teaching method was employed to include ideal first-principles model data in the training for a better performance. Then, these RNN models were incorporated into an MPC to simulate the control performance of a CSTR. [82] used RNNs to create a multi-input multi-output (MIMO) control scheme for a test bed furnace temperatures in which the weights of the RNN model are adopted as the operation proceeds. This neuro-controller was shown to be successful in setpoint tracking and against disturbances. With the advancements in sensor technologies and tools to digitalize experimental systems, feedback control with ML models can be implemented in a smarter manner.

Our previous work on the CO_2 reduction process incorporates smart manufacturing techniques into the experimental field to fully automate and digitalize the setup to leverage the potential for voluminous data production from multiple sensors to accelerate the experimental procedures and contribute to the scale-up efforts. Specifically, Chapter 3 summarized the efforts to connect UCLA's experimental RCE reactor to the Clean Energy Smart Manufacturing Institute's (CESMII) Smart Manufacturing Innovation Platform (SMIP) to securely store, organize and contextualize data generated during the experiments as well as meta information of the setup components. SMIP can also be used as a deployment environment for data-driven models and control, to monitor real-time data, and to extract correlations between experimental parameters. In addition to that, Chapter 3 elaborates on the RCE's sensors and how these are upgraded to smart sensors using the available data. One example is the automated gas chromatogram (GC) code, which fully automates a manual procedure by imitating the steps followed by the experiment supervisor. Consequently, the experimental setup becomes more compact and efficient.

There are previous data modeling and control efforts for the RCE setup with a different cat-

alyst electrode, which are used as a base for this work. [105] explains in detail how statistical feedforward neural network (FNN) architectures can be used to model the steady state operation points given the initial experimental input parameters. This model is trained with a database of open-loop steady state experiments conducted over the past years, accounting for the experimental uncertainties. This work also presents a method to reciprocally use empirical first-principal models with the developed ML models to improve the experimentally extracted correlations. Finally, in Chapter 2, we used support vector regression (SVR) to model real-time reaction rates for ethylene and inserted the SVR model into dynamic mass balance equations to implement a feedback control scheme at economically optimized setpoints. The SVR model constructed in this work also accounts for a fast catalyst decay and uses delayed feedback concentrations from the GC to update the model. Thus, this work has been the first successful single-input single-output (SISO) control instance of the complex electrochemical CO_2 reduction process using ML methods.

Motivated by the above considerations, this work proposes an ML-based scheme to implement real-time optimization (RTO) and multivariable feedback control in an experimental electrochemical reactor for CO_2 reduction. Specifically, two RNN models are developed on the basis of existing experimental data to estimate the dynamic response of the reactor operation. Subsequently, the information from the sensors and RNN models are integrated and used by two Proportional-Integral (PI) controllers that manipulate two inputs, applied voltage and electrode rotation speed, to the electrochemical cell, constructing a MIMO control scheme. In addition, an operational steady state model and an RTO are developed to calculate the economically optimum setpoints for the ethylene and carbon monoxide production rates by integrating market information. The proposed control and optimization scheme is demonstrated by a series of experiments that control the production of

121

ethylene and carbon monoxide from the RCE cell.

We believe the demonstration of a MIMO feedback control and RTO using this bench-scale reactor will be critical for the realization and operation of an industrial-scale CO_2 electrolyzer for the following reasons. First, moving forward from SISO to MIMO control and generalizing the approach to account for different combinations of inputs and outputs is essential for reactions involving multiple products such as the CO_2R. In addition, due to the non-selective nature of the reaction, there is a limit to solving this problem solely through the development of catalysts or the design of single-unit reactors, and efforts should be made to integrate multiple processes. Such process integration approaches include not only the connection of upstream/downstream separation or conversion processes, but also the design of a multi-stage electrolyzer sequence [139]. Real-time feedback control of a single-unit reactor demonstrated in this work acts as a building block for constructing a process control network where its communication with neighboring units is extremely important. Finally, the implementation of RTO which optimizes the overall process but can be applied to the actuation of individual control units is critical, considering how tightly the control of a single-unit reactor is tied to other processes and the electric grid infrastructure.

The rest of this chapter is organized as follows. In the section entitled "Preliminaries", the experimental reactor setup and database generation for ML model training are described. In the next section, entitled "Modeling and Optimization of the Experimental Electrochemical Process", the construction of a steady model using FNNs and the real-time optimization of this model are discussed. In the following section entitled "Dynamic Modeling of the Experimental Electrochemical Process", the methodology for enhancing our dataset to train a dynamic RNN model is elucidated. The section entitled "MIMO Control Architecture and Controller Tuning" elaborates on the imple-

mentation of the two PI controllers, their integration with the RNN models and the tuning of the controllers. Finally, the performances of the models and the controllers are evaluated.

4.2 Preliminaries

This section presents the details of the experimental setup used in this work. All of the experimental devices apart from nuclear magnetic resonance (NMR) are digitized via Laboratory Virtual Instrument Engineering Workbench (LabVIEW) software. The experimental process employs a code for processing GC signals, controllers and actuators, and it is fully automated with Python scripts that are integrated into a LabVIEW interface, in addition to the ML models that predict the concentrations at each second. Figure 4.1 shows the experimental RCE reactor at UCLA. The specific feedback controller parameters are determined in advance. The control system implementations include closed-loop experiments with arbitrary starting concentrations, which are driven to the setpoints. The process data flow is connected to the database of *SMIP* provided by *CESMII*.

4.2.1 Process Overview

Electrochemical CO_2 reduction on copper 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 involving multiple reaction intermediates. Mass transport characteristics of an electrochemical system affect the transfer of reactant to the catalyst surface as well as the removal of intermediates and products away from the surface. The relative time scales of different processes in the overall

reaction can be realized and controlled systematically in our RCE reactor [73]. 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 GC. The relevant reactions for these products are shown below:

$$2CO_2 + 8H_2O + 12e^- \rightarrow C_2H_4 + 12OH^- \tag{4.1a}$$

$$CO_2 + H_2O + 2e^- \rightarrow CO + 2OH^- \tag{4.1b}$$

$$CO_2 + 6H_2O + 8e^- \rightarrow CH_4 + 8OH^- \tag{4.1c}$$

$$2H_2O + 2e^- \rightarrow H_2 + 2OH^- \tag{4.1d}$$
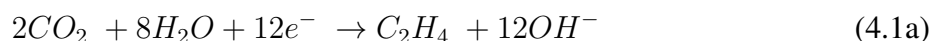
In Chapter 2 and [105], we used electropolished atomically-flat polycrystalline copper cylinder electrodes as CO_2R catalyst. The smooth nature of the exposed electrode surface caused a fast catalyst deactivation throughout the experiments, and various modeling and control approaches were applied to handle this challenge. In this study, we use the same experimental setup with nanoporous copper cylinder electrodes. Nanoporous structure directly synthesized on Cu cylinders increases the roughness of the surface with higher electrochemically active surface area (ECSA) inside the pores [149] and is more resistive to catalyst deactivation. Inside the pores, internal pore diffusion is the dominant mode of mass transport where all species have a long residence time and the electrochemical environment (e.g., concentration, pH, and electrical potential) becomes highly localized. The high ECSA as well as the longer residence time of intermediates due to internal pore

diffusion shift the selectivity towards producing more multicarbon (C_{2+}) products. To account for this change in selectivity, new modeling and control approaches are applied in this work.

4.2.2   Experimental Setup

The RCE system consists of two electrode chambers divided by an anion-exchange membrane, a mass flow controller (MFC), a potentiostat, a temperature control block, and a modulated speed rotator (MSR). During the experiment, pure CO_2 gas is fed at a fixed mass flowrate at 20 $mL \cdot min^{-1}$ 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 a 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 hermetic so that gas phase products can be quantified by a gas chromatogram (GC) in real time. An automated GC code is written for triggering injections, peak detection, baselining, and calculation of the areas under the peaks to quantify the gas phase concentrations in ppm using available calibration data, as explained in Chapter 3. One GC injection takes 14.3 minutes to complete, and is followed by 6 minutes of cool down before the following GC injection. Thus, when a GC measurement is obtained, it is delayed and is related to the reactor overhead gas concentrations from 14.3 minutes ago. Liquid phase products accumulate in the electrolyte solution and are measured

by NMR at the end of the experiment.

In this work, 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.

$$r_{C_2H_4 \mid CO} = \frac{C_{C_2H_4 \mid CO}^{ppm}}{10^6} \times \frac{\dot{V}_{CO_2}}{60 \times 10^6} \times \frac{P}{RT} \qquad (4.2)$$

where $C_{C_2H_4 \mid CO}^{ppm}$ 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 in $mL \cdot min^{-1}$ at standard temperature and pressure (STP), P is the standard pressure at 1 atm, R is the universal gas constant and T is the standard temperature at $0\,^{\circ}C$. 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. 4.2 are constants, the production rates are proportional to the concentration in ppm.

This experimental setup is automated and digitalized as explained in detail in Chapter 3. Before the digitalization and automation efforts started, there were already accumulated open-loop steady state experimental data obtained under different input parameters, which were also sent to the SMIP. 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, the gas phase concentrations are measured via GC at 15th, 35th, 55th, and 75th minute, and the resulting current (A) and a calculated variable surface potential (V vs SHE) is sensed 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 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:

$$E_{\text{surface}} = E_{\text{applied}} - i \times R + E_0 \tag{4.3}$$

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) [73]. E_0 is the standard reduction potential of the reference electrode used (Ag/AgCl/1 M KCl), so Eq. 4.3 removes the potential drop across the solution due to the resistance to ion transport in these systems.

Remark 15 *Experimental conditions may cause uncertainty, therefore experimental data modeling brings some unique challenges compared to well-structured data, such as data generated by simulations or obtained from industrial facilities. In the RCE setup, despite the electrolyte resistance being kept as steady as possible (6.2 $\pm$ 0.2 Ω), the resistance values are measured a priori, and may vary for each experiment based on stock solution preparation, environmental temperature, connectivity of inner electrical circuit, etc. Also, the differences in roughness factors that emerge from doing cyclic voltammetry during the preparation of the porous electrode may affect the catalyst activity, especially at lower overpotentials. Consequently, it is possible to observe different product concentrations at the same input conditions. Therefore, the averages and standard deviations from the steady state open-loop experimental data are shown in Table 4.1 and are used*

in the ML model construction. Also, a plot is shown in Figure 4.3 for three experiments conducted at 100 RPM and in the close vicinity of -1.317 V vs SHE that have nearly overlapping current flows but produce C₂H₄ concentrations within a standard deviation of 43.4 ppm.

The existing experimental data were collected under various applied potentials at the electrode rotation speed of 100 and 800 RPMs as shown in Table 4.1. Also, an empirical correlation for estimating the current value at a specific surface potential and electrode rotation speed is developed using the steady state experiments and is as follows:

$$i = i_0 \Omega^{0.203} e^{kE_{surface}} \tag{4.4}$$

where i_0 and k are constants and Ω is the rotation speed. In order to find i_0 and k, Eq. 4.4 is linearized into the following form.

$$\ln \frac{i}{\Omega^{0.203}} = kE_{surface} + ln(i_0) \tag{4.5}$$

Average values of the experimentally measured current and surface potential and rotation speeds are linearly fitted into Eq. 4.5 to find the constants. The distribution of the fitted data is shown in Figure 4.2. The following open- and closed-loop experiments, and data shifts can be explained, and this experimental system can be simulated using this empirical correlation.

Table 4.1: Averages and variations in surface potential, current, and C_2H_4 and CO concentrations in repeated open-loop experiments under different operating conditions.

Rotation Speed	Potential (V vs SHE)	Current (A)	C_2H_4 (ppm)	CO (ppm)
100 RPM	-1.31	-0.0296	376	157
	-1.28	-0.0201	217	159
	-1.23	-0.0131	67	211
	-1.20	-0.0086	30	255
800 RPM	-1.30	-0.0359	363	505
	-1.27	-0.0267	216	500
	-1.24	-0.0173	86	488
	-1.20	-0.0129	25	483

(a) Averages of experimental inputs and outputs in various ranges.

Rotation Speed	Potential (V vs SHE)	Current (A)	C_2H_4 (ppm)	CO (ppm)
100 RPM	0.0029	0.0029	35	14
	0.0047	0.0032	96	21
	0.0041	0.0002	43	42
	0.0015	0.0002	11	8
800 RPM	0.0128	0.0030	76	65
	0.0019	0.0010	49	56
	0.0060	0.0010	31	22
	0.00199	0.0010	9	16

(b) Standard deviations of experimental inputs and outputs in various ranges.
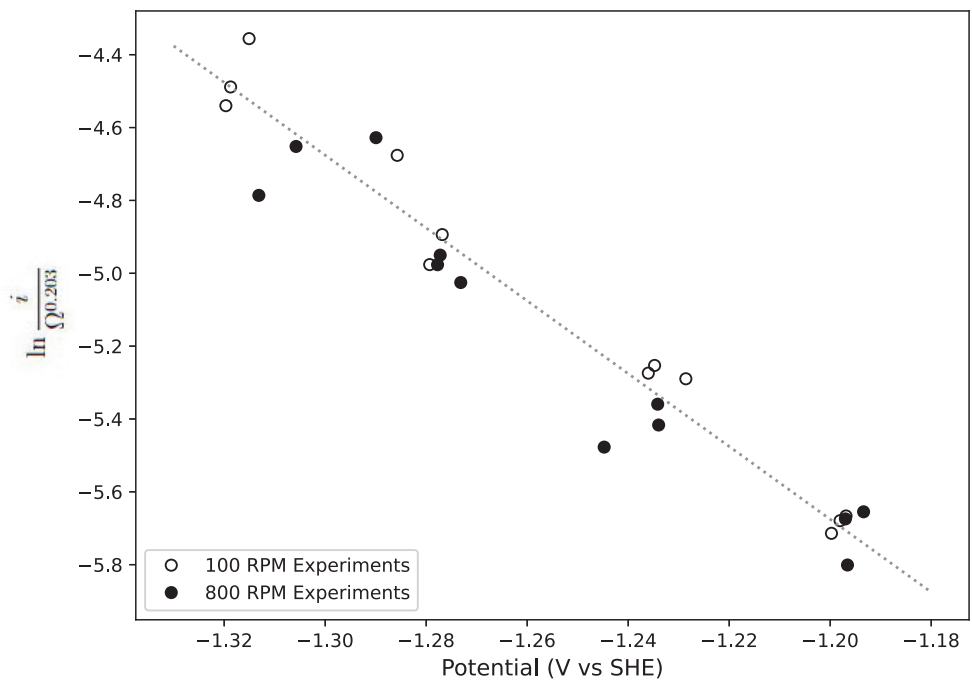
Figure 4.2: Empirical correlation of normalized current versus surface potential at two different electrode rotation speeds (100 RPM and 800 RPM). The dashed line represents the reduced logarithmic correlation with respect to surface potential.
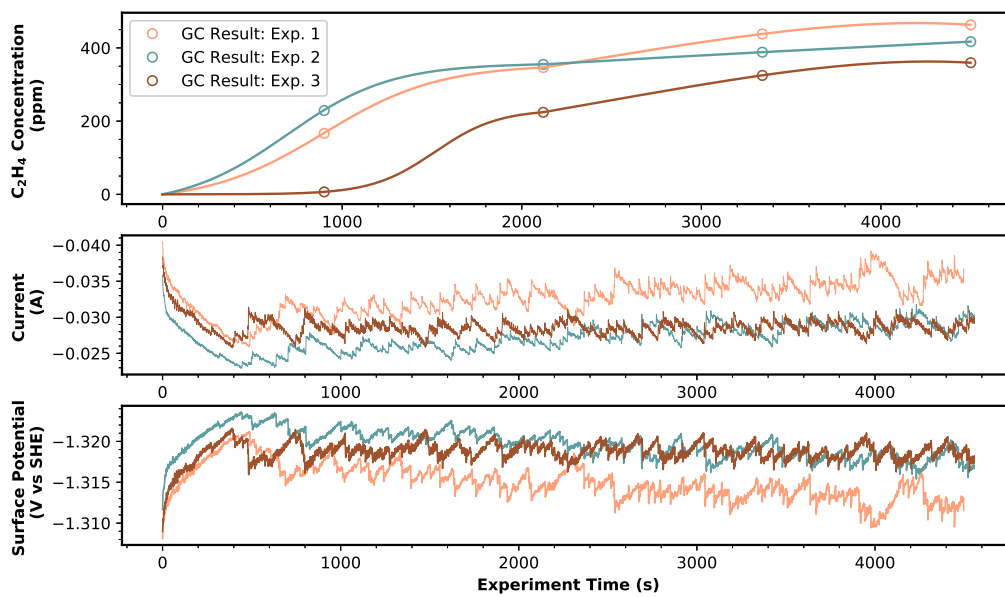
Figure 4.3: Comparison of 3 experimental C_2H_4 concentrations from open-loop steady state experiments conducted at nearly identical surface potentials (bottom plot) and 100 RPM. The top plot shows experimental GC results with fitted curves calculated by polynomial regression.

4.2.3 Open-Loop Step Change Experiments

The modeling objectives of this study are C_2H_4 and CO concentrations; therefore, it is important to mention the trend with respect to the changes in one of the parameters while keeping the other one constant, where the parameters are the applied potential and the rotation speed of the catalyst. C_2H_4 has a very strong correlation with the surface potential and as the applied/or surface potential increases, C_2H_4 increases strongly. On the other hand, CO concentrations have a very strong correlation with the rotation speed, so that increasing the rotation speed results in increasing CO concentrations. These effects can be seen from our steady state machine learning model discussed in the following section and the cross effects will be discussed in detail.

In order to control the experimental setup, dynamic data must be generated in addition to the steady state data. Following the steady state experiments, various applied potential and rotation speed step change experiments were conducted separately. The timing of the GC injections is the same as the previous open-loop steady state experiments, and it started at the 15th minute with a sampling period of 20.3 minutes. Based on the steady state experiments, it takes a maximum of 5 GC injections (around 80 minutes) to reach the new steady state and stay there. However, to see the shorter-term effects, there are a few experiments in which the step change is applied at 3 injections time. One example of a step change experiment is starting the experiment under a fixed applied potential and an initial rotation (e.g., at 100 RPM) and changing the rotation speed to 200 RPM after 5 GC injections and then changing it to 800 RPM until the new steady state is reached. Then, the reverse procedure is applied such that the rotation speed is first reduced to 200 RPM from 800 RPM and then to 100 RPM. A similar procedure is applied for the surface potential, in which the

applied potential is manipulated to adjust the surface potential to the desired value. The surface

potential values are increased/decreased, and the changes in gas phase concentrations are recorded.

One instance of rotation speed step change experiments under a constant applied potential for CO

concentration is shown in Figure 4.4. It is important to note that changing the rotation also affects

the current passed between electrodes and thus affects the surface potential. Consequently, when

the rotation speed is changed in an open-loop step change experiment, the surface potential does

not remain constant and is subject to small changes. However, this effect is relatively small.
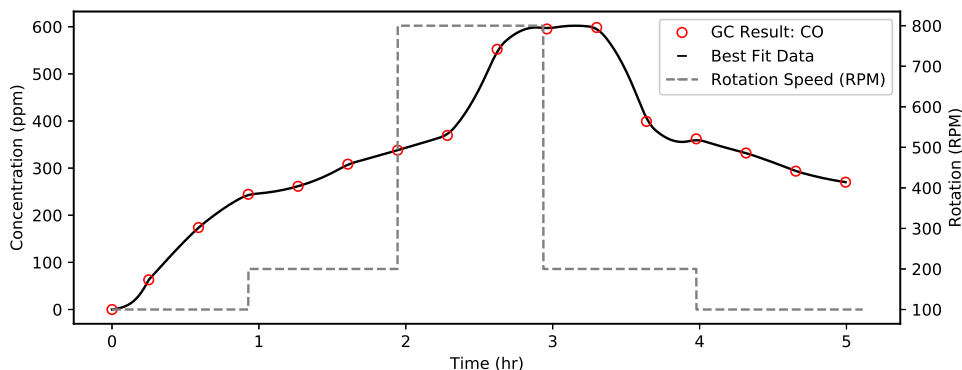
Figure 4.4: Probable experimental trajectory of CO concentration using polynomial fit curves for open-loop rotation speed step change experiments while keeping the applied potential constant.

4.3   Modeling and Optimization of the Experimental Electro-chemical Process

4.3.1   FNN Modeling for Steady State Setpoints

After selecting the controlled outputs and manipulated inputs, a process model needs to be

developed to capture the input-output relationship and used for the design and tuning of the control

system. Ideally, a mathematical model with an explicit form (e.g., first-principles model) is the best modeling option because of its explainability and reliability. However, in this work, it is challenging to develop such a model due to the complexity of the reaction mechanisms and the lack of full understanding of the electrochemical reactions. As summarized in [120], there are several articles proposing respective unique explanations of the reaction mechanisms for this reaction. Therefore, there is not a single solid conclusion of the reaction chemistry that can be used to develop a first-principles model.

To address this challenge, a data-driven model is developed. Specifically, a feed-forward neural network (FNN) is trained based on the experimental data collected from the steady state experiments discussed in Section 4.2.3. There are seventeen products coming out from the electrochemical reactor. The FNN model uses two inputs (i.e., surface potential and catalyst rotation speed) to predict the production rate of sixteen product species. Hydroxyacetone production rates are not included in the modeling phase, since its production rates are either 0 or very low (under 2 ppm). The inputs are normalized with a standard scaling factor. The FNN model has a hidden layer with 64 neurons activated by a ReLu function. The Softplus function, $f(x) = ln(e^x + 1)$, is selected as the activation function of the output layer to ensure non-negative prediction since the reactor did not consume any of the product such that the output of our FNN model can not be negative. Additionally, the Softplus function predicts the output with a smoother curve which aligns the physical expectation better than other candidate activation functions, such as ReLu and Sigmoid. The mean squared error (MSE) function is utilized as a cost function to train the FNN model as explained in [105].

Although an FNN model has the potential to provide a universal approximation to any nonlin-

ear relation [155] and has demonstrated reliable performance in addressing chemical engineering and process control tasks [92, 115], it nominally treats each training data point equally, and therefore, it can lead to relationships that are affected by data points that have significant experimental variability. To account for this issue, we calculated the coefficient of variance for each data point, based on their respective mean and variance, and used it as the weight of the specific data point in the training process to account for data uncertainty [105]. Specifically, the loss function used to train the weighted-FNN model can be expressed as follows:

$$Loss = \frac{1}{d}\frac{1}{m} \sum_{i=1}^{d} \sum_{j=1}^{m} \frac{1}{v_{i,j}^2} |y_{i,j} - \hat{y}_{i,j}|^2 \tag{4.6}$$

where d is the number of training data points, m is the number of output states, $y_{i,j}$ is the ith reference data point for the jth product, and $\hat{y}_{i,j}$ is the predicted production rate for the jth product under the ith input combination. $v_{i,j}$ is the coefficient of variance of the ith data point for the jth product, which can normalize the variability of each data point and provide unbiased weight for products having production rates in different magnitudes. With the weighted loss function, the FNN model is granted more tolerance for prediction error to prevent the model from overfitting the data uncertainty when a training data point has a higher variance. The visualization of the FNN model can be seen in Figure 4.5.

Remark 16 *Although the production rate of the liquid phase product cannot be measured in real-time during the experiment, it can be calculated based on the result of the NMR after the experiment is complete. Therefore the steady state neural network model can predict the production rate of the liquid-phase product but cannot be implemented in real-time with the dynamic model.*
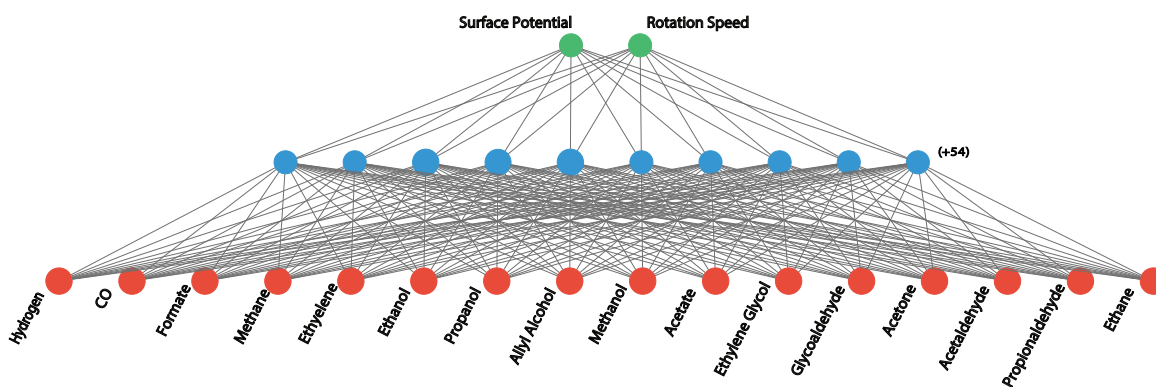
Figure 4.5: FNN architecture based on weighted data mapping two inputs (i.e., surface potential and rotation speed) represented in green circles to the production rates of sixteen products (outputs) represented in red circles through a densely connected hidden layer represented in blue circles. Only 10 of the 64 nodes are shown in the figure as blue circles. The model includes 54 more hidden nodes, in the same hidden layer.

Twelve averaged experimental data points representing surface potentials, rotation speeds and corresponding 16 product output concentrations from 36 steady state experiments were used for training and testing of the model. Nine of the experiments are earmarked for training and 2 were used for testing. There are 4 experimental points for each 100 RPM and 800 RPM experiments, 2 experimental points for the 400 RPM experiments, and one experimental point for 200 and 600 RPM experiments, each. The model is trained for 3000 epochs based on a mean squared error (mse) loss function. The mse values are 0.0055 and 0.0063 for training and testing. The results for gas phase products are shown in Figure 4.6. Ethane is not included in the plots since its concentration does not go higher than 3 ppm under our operation range and thus it is very low. The hydrogen concentrations in Figure 4.6 (a) show a linearly increasing trend with increasing surface potential. The rotation speed has a small increasing effect on the concentrations. Methane concentrations shown in Figure 4.6 (b) are exponentially increasing with increasing surface potentials at higher
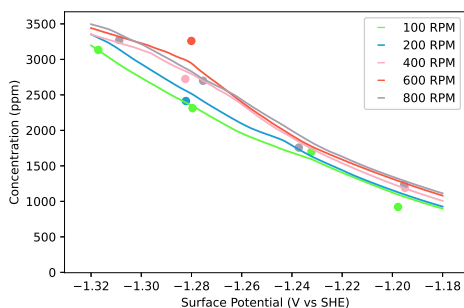
(a) H$_2$ concentration versus surface potential and rotation speeds.
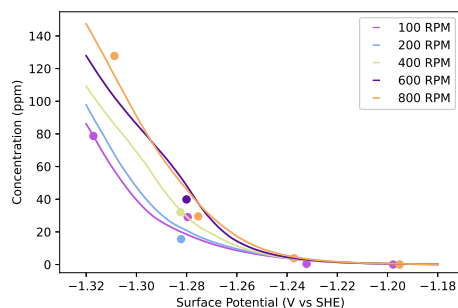
(b) CH$_4$ concentration versus surface potential and rotation speed.
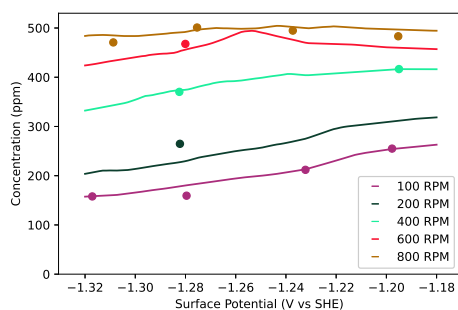
(c) CO concentration versus surface potential and rotation speed.
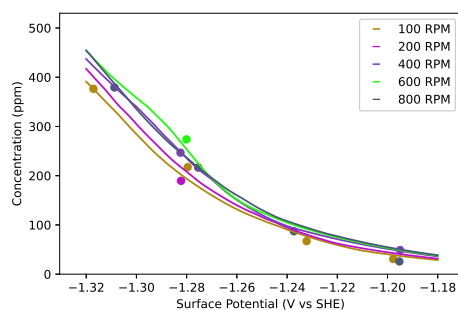
(d) C$_2$H$_4$ concentration versus surface potential and rotation speed.

Figure 4.6: FNN predictions for gas-phase products under various input conditions, where (a), (b), and (d) demonstrate that the production rates of H$_2$, CH$_4$, and C$_2$H$_4$ are weakly correlated to the rotation speed, and (c) demonstrates that the production rate of CO has stronger correlation with the rotation speed. The solid dots represent the experimental data averaged over three repetition experiments at corresponding input conditions and support the predicted curves calculated from the FNN model.

overpotentials, and the effect of rotation speed is more significant at higher overpotentials. CO concentrations shown in Figure 4.6 (c) show a very strong proportional correlation with rotation speed. However, at lower rotation speeds, an increase in surface potential results in a decrease in CO concentrations. At higher rotation speeds, the CO concentrations are affected very weakly by the surface potentials. Finally, the ethylene concentrations shown in Figure 4.6 (d) exhibit exponentially increasing trends with increasing surface potential. The effect of rotation speed on

the ethylene concentrations is small, and the direction depends on the specific rotation speeds.

4.3.2 Real-Time Optimization

The developed FNN model based on weighted data can predict the reactor performance by mapping combinations of control actions with the production rates of each species produced by the reactor. The next step is to apply this information to our multi-variable control scheme. Specifically, the prediction from the FNN model is used to solve an optimization problem computing the optimum setpoint for the multivariable control system. The optimization problem is designed to maximize the economic benefit of operating the reactor. To simplify the optimization problem for this study, electricity consumption is assumed to be the only operational cost for our reactor and the revenue of the operation is the total value of the generated product calculated based on the prediction of the FNN model. Thus, the optimum setpoint is where the reactor profit is maximized. The mathematical expression of this optimization problem is given below and a third-party software IPOPT [188] is utilized to solve this optimization problem. Specifically, the optimization problem has the form:

$$\mathcal{J} = \underset{\hat{x} \in \mathbf{D}}{\arg\max} \ R(\hat{x}, E_{surface}) - C(I, E_{surface}) \tag{4.7a}$$

$$s.t. \quad F_{nn}(E_{surface}, \Omega) = \hat{x} \tag{4.7b}$$

$$C(E_{surface}, I) = c_e \times I \times E_{surface} \tag{4.7c}$$

$$R(\hat{x}, E_{surface}) = \sum_{j=1}^{m} c_j \times \hat{x}_j \tag{4.7d}$$

$$100RPM \leq \Omega \leq 800RPM \tag{4.7e}$$

$$-1.3VvsSHE \leq E_{surface} \leq -1.19VvsSHE \tag{4.7f}$$

In the above equations, functions $C(E_{surface}, \Omega)$ and $R(E_{surface}, \Omega)$ calculate the cost and revenue of operating the reactor, where $E_{surface}$, Ω, I, c_i, and c_e stand for the surface potential, the rotation speed of the catalyst in the unit of rotations per minute (RPM), current (A), the market price of the jth product, and the electricity price respectively. $\hat{x}$ is the prediction from the FNN model containing production rates for $m = 16$ product species. When solving for the optimum setpoint, the initial guess for control actions (e.g., $E_{surface}$ and Ω) is first made by the users and provided to the optimization problem among the product and electricity prices. Subsequently, the IPOPT will alter the control actions, which leads to a change in the energy consumption and the production rates predicted by the FNN model, to maximize the profit of the reactor. Once the optimum control actions are found, the corresponding production rate of C_2H_4 and CO, given by the FNN model, will be sent as the setpoint for the multivariable control system. As an example, we assumed that the electricity price is the only varying price parameter, and by picking a value

of 0.066 $/kWh$, the most economically feasible setpoints are found to be 112 ppm for C_2H_4 and 490 ppm for CO. For the second case scenario, a 40% price decrease for H_2 and C_1 products (e.g., CO) and a 60% price increase in C_2 products (e.g., C_2H_4 or C_2H_5OH) are assumed. The optimization problems are solved and setpoints are calculated to be 283 and 350 ppm for C_2H_4 and CO respectively. Finally, one pair of setpoints is selected to show that selectivity can be adjusted at any desired value such as 1:1. Thus, setpoints corresponding to 1:1 ratio are calculated to be at 200 ppm and this value is selected to be the final setpoint. This optimization design assumes that all the products are sold at market values. However, the extra costs like the separation of gas and liquid products must be included in a real case scenario.

4.4 Dynamic Modeling of the Experimental Electrochemical Process

In order to implement multivariable control of this experimental process, there are two dynamic models needed: one for C_2H_4 and another one for CO. The concentration of the gas products in the headspace of the RCE cell can be approximated as a mixing volume where the gases produced on the catalyst mix and are carried over by the CO_2 gas flow, as shown in Chapter 2. This approach can be improved by modeling only the dynamic reactor data, which will involve all the inherent reactor dynamics without being derived by mass balances built for other similar reactor types (e.g., CSTR). Due to the absence of first principle dynamic equations of the RCE reactor for electrochemical CO_2 reduction, data-driven models can be built to model process dynamics.

4.4.1   Automatic Data Fitting to GC Measurements

The experimental concentration data from the GC are few and discrete; therefore, it is challenging to apply the deep-learning method (e.g., neural network modeling) to the raw experimental data. To tackle this, the GC results are combined using polynomial best fits to approximate a probable experimental trajectory, which significantly increases the number of data points, and enables building a recurrent neural network model.

In order to connect the respective GC points, 3rd order polynomials are used for 3 data points. From the experimental side, three GC measurements represent 40 minutes of the experimental time span. For each 3 respective GC measurements, 2 polynomial curves are fitted. For example, the first polynomial fit is between 2nd, 3rd, and 4th GC measurements. The second polynomial curve fitting is between 3rd, 4th, and 5th GC measurements. However, with this method, there are two polynomial trajectories for the interval between each respective GC point. The most convenient polynomial fit between these two trajectories can be chosen intuitively. To automate this procedure, an algorithm is used to select the lower trajectory for an increasing concentration of more than 5% and the upper trajectory for a decreasing concentration of more than 5%. This assumes that the aforementioned increase or decrease is in an exponential trend. If the concentrations that decrease or increase are less than 5%, then opposite trajectories are selected, implying that the experimental trajectory evolved into a logarithmic change phase. Due to the combination of different polynomial fits with this method, the trajectory obtained might not be smooth. To resolve this issue, a Savitsky-Golan filter with a time window of 500 seconds is applied, and the obtained trajectory is adjusted to the correct timing to compensate for the time dislocation due to the filter window length. Figure 4.7

demonstrated an example of using the proposed method to enhance the experimental data from one experiment. In this example, sixteen data points, represented in red circles, are collected using the GC during over five hours of experiment. This raw data provides very limited information to train a neural network model. After the data fitting, the available concentration estimations increased to 18000, for this one experiment, which is shown as the black curve in Figure 4.4.

Remark 17 *This data fitting is not claimed to be the actual experimental trajectory or the best data interpolation method. Changing the order of polynomials or the number of fitting data points in the intermediate trajectory may result in different interpolated trajectories. Other functions (e.g., linear interpolation, sigmoid function, etc.) can be considered as additional candidates for this fitting method.*
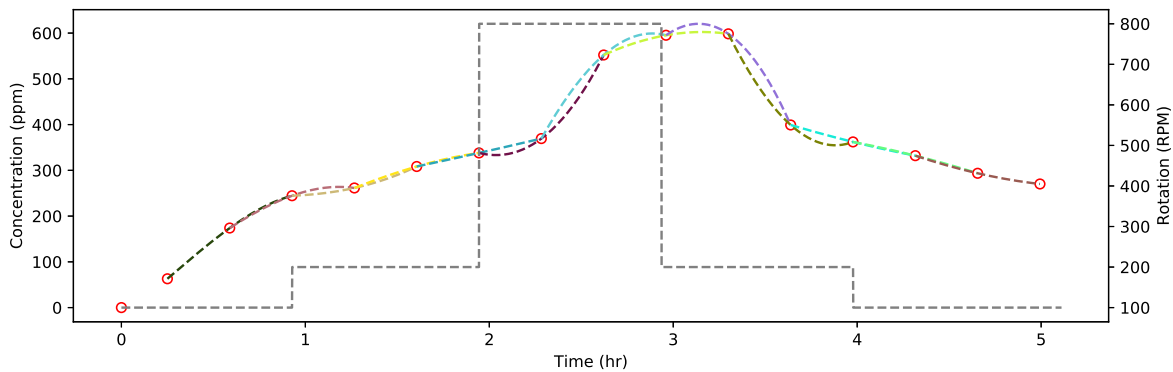
Figure 4.7: Visualization of auto data fitting algorithm for change in CO concentration in an open-loop step rotation speed change experiment. The best-fitted trajectory is demonstrated as the black solid curve in Figure 4.4 generated by the second step of the algorithm that automatically picked the best fit from the candidate trajectories between every two points and smoothed the overall trajectory with Savitzky–Golay filter.

4.4.2   Recurrent Neural Networks and LSTMs

RNN models are proven to be effective in capturing trends from time series data, and thus, are used in this work to model the dynamic behaviors of the electrochemical reactor for feedback control purposes. The increased amount of probable trajectory time series data is used to train this RNN model. Specifically, the RNN model can learn the time series from a defined time window and the correlations between the respective data points [208]. The RNN architecture is depicted in Figure 4.8 (a). Input parameter vectors in time series are fed into the RNN and each time series vector is subject to recurrent calculations in the hidden layers. RNNs learn the time dependencies and provide results to the output layer, which is usually a fully connected dense layer to better map the hidden states into meaningful time series outputs.

One of the most powerful RNN architectures for time series is the Long-Short-Term-Memory (LSTM) model. LSTMs are different from simple RNNs due to their resistance to exploding/vanishing gradients exhibited in other neural network models thanks to the forget, input and output gates in each recurrent unit [208]. Each gate and the LSTM recurrent unit are shown in Figure 4.8 (b). Here, cell state is used to transfer useful information and past relations from the previous recurrent units to the next recurrent units, and thus, it can be considered as the memory of this sequence [218]. Cell state is able to keep all the information from the initial time step recurrent unit until the final time step recurrent unit. The more relevant relationships are kept in long-term memory, whereas the less relevant information is removed at each time step. On the other side, there is the hidden state used to keep the output from each recurrent unit and transfer it to the following LSTM layer. These hidden-state outputs are formulated in a way that ultimately needs to be fed into a

dense layer outside the LSTM, as mentioned in the previous paragraph. At each recurrent LSTM unit, the previous hidden state vector is combined with the new time step input vector and fed into the gates.

Additionally, there are three gates in an LSTM layer: forget, input, and output gates. Each gate contains trainable sigmoid activated neural networks, as shown in 4.8 (b). The combined vector of previous hidden state and new input vector is fed to the forget gate. Due to the sigmoid activation function, the output is between 0-1. If the forget gate yields 0, then the previous cell state will be forgotten. Conversely, if the output is close to 1, the previous information from the cell state is retained between the forget and input gates. Next, there is the input gate, which also has a sigmoid activated trainable neural network that decides to what degree the new input vector should be remembered. The output of the input gate is pointwise multiplied with the Tanh activated neural network outputs which is trained to learn the effect of the new input vector for the current time step output. Tanh activation function yields results between [-1,1], a negative value signifies that the new input vector might have a decreasing effect. The input gate yield is added pointwise to the cell state value. Finally, the output gate, again, filters the previous hidden state and input vector with a sigmoid activated neural network and this output is pointwise multiplied by the Tanh activated neural network output of the hidden state. This gives the time step output in the hidden

144

state [218]. The LSTM structure can be represented mathematically as follows:

$$i(k) = \sigma(\omega_i^x x(k) + \omega_i^h h(k-1) + b_i) \tag{4.8a}$$

$$f(k) = \sigma(\omega_f^x x(k) + \omega_f^h h(k-1) + b_f) \tag{4.8b}$$

$$c(k) = f(k)c(k-1) + i(k)tanh(\omega_c^x x(k) + \omega_c^h h(k-1) + b_c) \tag{4.8c}$$

$$o(k) = \sigma(\omega_o^x x(k) + \omega_o^h h(k-1) + b_o) \tag{4.8d}$$

$$h(k) = o(k)tanh(c(k)) \tag{4.8e}$$

$$\hat{x}(k) = \omega_y h(k) + b_y \tag{4.8f}$$

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, $w^{h,x}$ is the weight matrix to the hidden state vector h and input vector x, b_i, b_f, b_c, b_o, b_y represents biases and the subscript y indicates relationship to the output [206].

Meaningful Time Series Predictions

Feedforward Neural Network

Output Layer y_t y_0 y_1 y_n

Recurrent Layer $=$

Input Layer x_t x_0 x_1 x_n

$$x_t = \left[E_{surf\_t}, \Omega_t, i_t \right]$$
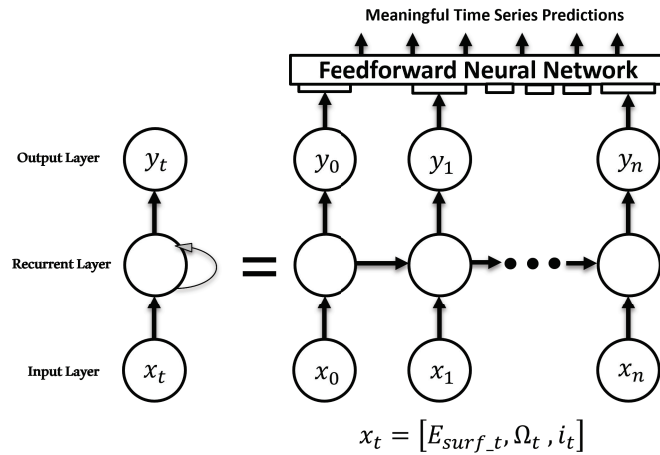
(a) Rolled and unrolled RNN architecture visualization.

c_{t-1} Forget gate Input gate c_t
Cell state

Output gate tanh

σ tanh σ σ

Hidden state
h_{t-1} h_t

x_t
Input vector
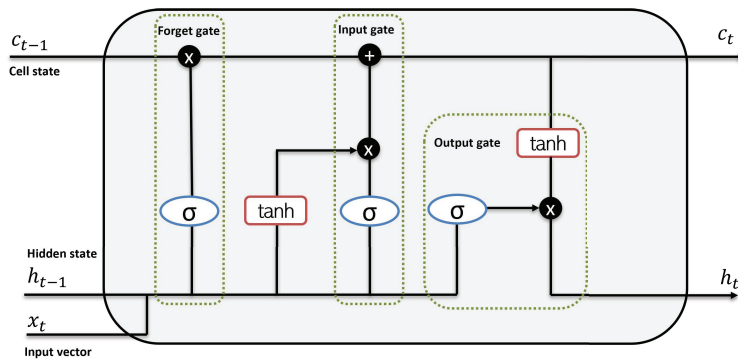
(b) LSTM layer diagram.

Figure 4.8: Overall structure of recurrent neural network (RNN) and long short-term memory (LSTM) network unit. The LSTM unit fits into the empty circles of the recurrent layer in the top plot.

4.4.3   LSTM based RNN Model Architecture

Two LSTM models are trained using the data calculated through the polynomial best fits between GC measurements. The inputs for the first LSTM model are surface potential, rotation speed, and current, and the output is C_2H_4 production rate converted to ppm using Eq. 4.2. For the second LSTM, the inputs are surface potential and rotation speed, and the output is CO production rate in ppm. Based on the experimental observations, this process has dead times of 600 to 1500 seconds for step changes on different input parameters. It can be seen in Figure 4.7 that it takes around 1 injection for the rotation speed change to show its effect on concentration. Also, it was seen from the experimental data that it might take more than 1500 seconds after the dead time for the process output to reach the new steady state. Consequently, the pastime window is very important for the dynamic behavior of the process and this time window should be around one hour. From this point, the models are trained with various time windows. The best model performances are obtained for 3600 seconds of time window for C_2H_4 and 3800 seconds for CO.

If one time window consists of 3600×3 (3 is the number of inputs parameters for the C_2H_4 estimator) data points, the model would be too big, computationally expensive, and perform poorly due to high number of training parameters. In addition to that, the predictions would be very noisy. Thus, the time window can be discretized in a way that still represents the last one hour without violating the correlations between consecutive data points. For example, the last one hour can be represented with data points at each 100 seconds. As a result, the (3600×3) time window is reduced to the (36×3) time window without losing relevant information. The same approach is applicable for reducing the model size for the CO estimator, which would have (3800×2) input

parameters for a per second base estimation and has (38×2) input parameters after 100 seconds of discretization. The model will make a prediction every 100 seconds. Also, the model is trained to predict the next 800 and 600 seconds of the production rates for C_2H_4 and CO respectively, with 100 seconds of discretization, having the output shape (8×1) and (6×1) in each LSTM. This way, LSTM is seen to learn the delay behavior better compared to an output shape of (1×1). Finally, both models are built with 200 hidden nodes in one hidden recurrent layer. It was seen that the prediction accuracy decreases if we decrease the number of hidden nodes. Also, the computation time increases and erroneous nonlinear trends are predicted by the model if we increase the number of hidden nodes.

Furthermore, regularization methods, such as recurrent dropout and L2 kernel regularizer, are performed to increase the generalization performance of the models to unseen data. Specifically, the following regularization methods are tuned to the best performance at 30% for recurrent dropout and 0.08 L2 kernel regularization. Increasing or decreasing those tuned parameters may lead to divergent predictions. The recurrent layer is connected to a dense layer of 8 nodes for C_2H_4 and 6 nodes for CO, which correlates the information from the recurrent node with the production rate values. A sigmoid function is used in the dense (FNN) output node to limit the predictions by the highest and lowest values in our training set. Multiple experiments from constant applied potential and rotation and two long step change experiments were selected for the test set. The remaining experiments were used to train the models. The predictions of the LSTM models are evaluated with an unseen testing set, and one set of the testing examples is demonstrated in Figure 4.9.
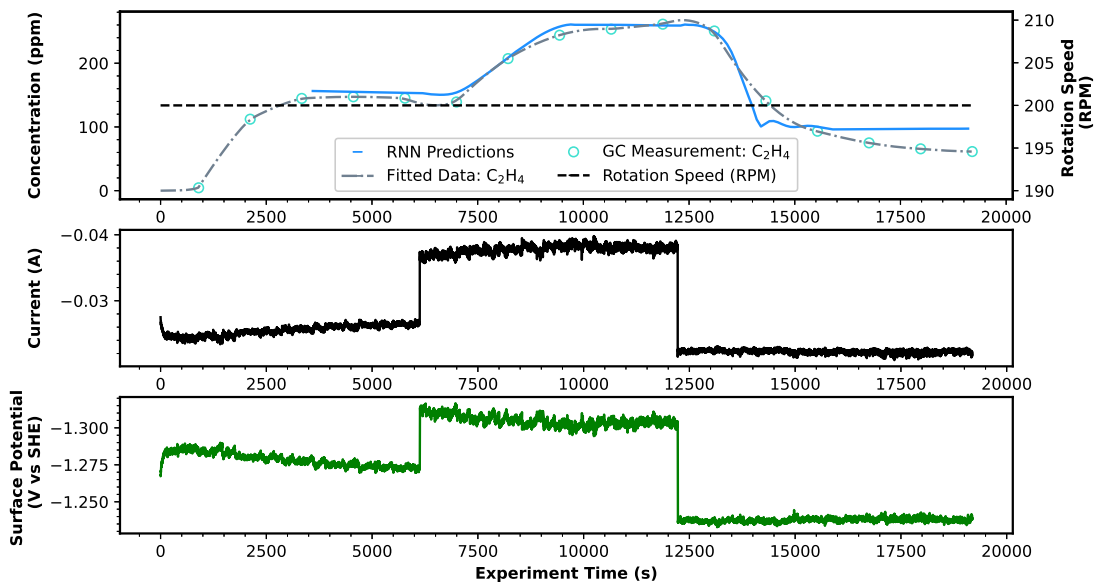
(a) C_2H_4 concentration prediction model with surface potential (V vs SHE), rotation speed, and current as inputs for a time window of 3600 seconds.
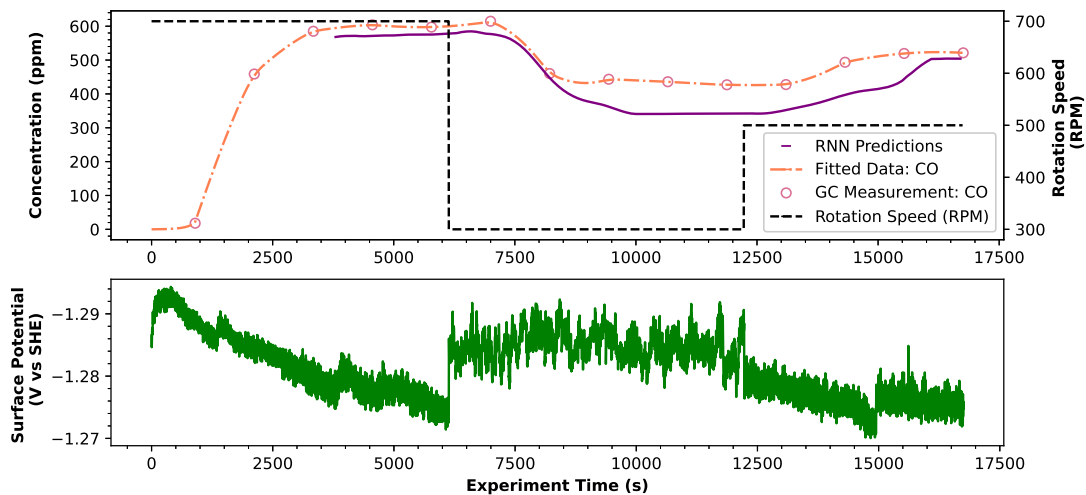
(b) CO concentration prediction model with surface potential (V vs SHE) and rotation speed as inputs for a time window of 3800 seconds. This experiment was conducted under constant applied potential and the change in surface potential is due to the change in current caused by the electrode rotation speed variation.

Figure 4.9: C_2H_4 and CO RNN predictions for open-loop experiments from the testing set.

Remark 18 *Electrochemical CO_2 reduction in RCE cell is seen to have long process delays of more than 1000 seconds. The models take around 1 hour time window to model those delays. However, 3600 seconds of time window is a very long sequence. Thus, training those models only with open-loop experiments might cause poorer performance in a closed-loop context. These models perform very successfully when only one of the manipulated input parameters varies sharply while the other one is constant. On the other hand, when two manipulated input variables vary at the same time gradually, the large time window of the LSTMs might not catch the dynamic trends as necessary. In order to enhance the LSTMs trained with the open-loop experiments, two preliminary controlled experiment were conducted, using the initial RNN models trained only with open-loop experiments. The dynamic data obtained from this experiment are used to train both estimators, which also leveraged the models to improve themselves to adapt their weights to the dynamic changes of the feedback control context.*

The models need 3600 and 3800 input data points to make predictions for C_2H_4 and CO, respectively. Thus, the first prediction is obtained at the 3600th second for the C_2H_4 estimator and at the 3800th second for the CO model. The RNN architecture could have been built in such a way that the concentration output from the previous time step is fed to the LSTM model to guess the next time step concentration. This approach could have been useful to model the experiment starting from the beginning of the experiment rather than 3600th second. However, before each experiment, pre-experiment measurements are performed including EIS to measure the solution resistance and cyclic voltammetry (CV) to check catalytic activity which already generates some products while cycling through a potential window. Thus, depending on the time span after this CV

150

procedure, the development of gas products in the early stage may vary. This can be seen from the brown experimental curve (Experiment 3) in Figure 4.3, in which there was a longer waiting time between pre-experiment measurements and the actual open-loop experiment compared to the other two experiments. This caused the delayed development of gas concentration trajectory at the early stage with a slower accumulation of gas products in the reactor headspace. Thus, feeding back the previous time step concentration estimation to the LSTM model causes confusion for the training set. As a result, the LSTM estimators wait about an hour until the system reaches equilibrium and are not affected by previous experimental development.

Remark 19 *Initially, both of the estimators were trained with three inputs: Surface potential (V vs SHE), electrode rotation speed, and current. The current is a significant indicator for the C_2H_4 concentration and it is very important to include the current for training. However, the CO concentration is more dependent on the rotation speed and the surface potential. When the models were trained with open-loop step-change experiments, it was seen that the LSTM model for CO has a stronger correlation with current rather than surface potential. On the other hand, increasing current under a fixed applied potential decreases the surface potential, as can be seen from Eq. 4.3. Conversely, if the applied potential is not constant, the surface potential will increase with increasing applied potential. Thus, increasing the rotation speed will have an increasing effect on the CO concentration, whereas it will also increase the current and decrease the surface potential. Consequently, the LSTM model trained with open-loop experiments will learn that increasing current means increasing CO. This provides a great training accuracy in the testing set performance of open-loop experiments. However, the model should also learn the competition between*

the surface potential and the rotation speed for dynamic experiments. When this model was tested, it was found that the model yields incorrect trends when the ethylene concentration is increased by adding more surface potential (thus increased current) and the rotation is decreased. From the known correlations and GC results, the CO concentrations should have decreased, whereas the estimator predicted increasing concentrations due to increasing current. After seeing this deficiency with the dynamic model prediction based on model involving three inputs, the CO estimator was retrained using two inputs, surface potential and rotation speed, and the aforementioned problem was resolved. A diagram that explains how the model training and verification were carried out using both open-loop and closed-loop experiments is shown in Figure 4.10.
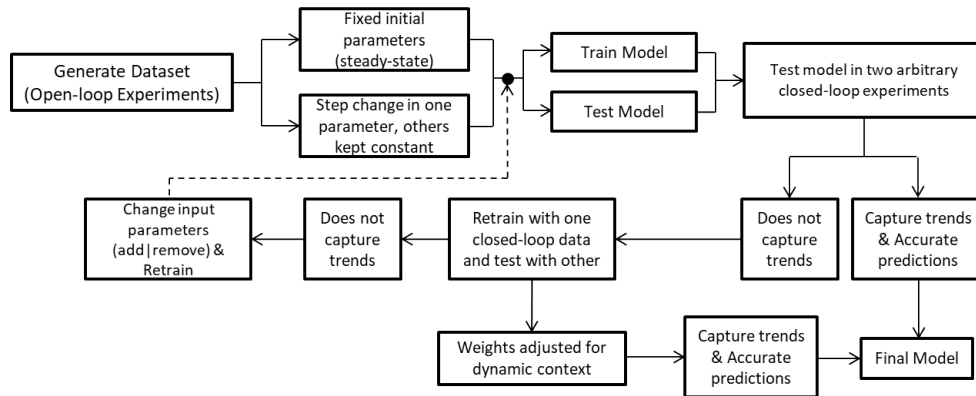
Figure 4.10: Procedure to optimize the LSTM model using open- and closed-loop experiments.

The C_2H_4 and CO estimators are trained with 83393 and 112789 data points obtained through probable experimental trajectories. The difference between the training test sizes is due to using different experiments for training both models. Also, the CO model is trained with 2 preliminary dynamic controlled experiments whereas the C_2H_4 model is trained with only one. The training and testing mean absolute error performances are 0.62 ppm and 3.15 ppm for C_2H_4 model and 0.63 ppm and 11.1 ppm for CO. This indicates that the models fit to the training data quite well, but it generalizes to the new data satisfactorily well within the experimental standard deviations.

4.4.4   Regularization Effects on Experimental Data Modeling

In 2014, [169] proposed the dropout method that randomly eliminates a percentage of hidden units during the training to eliminate/alleviate the overfitting effects in neural networks. When a unit is randomly excluded for an epoch, it prevents the rest of the neurons from excessively co-adapting. The dropout method is also used in our neural network architecture between the LSTM output and dense output layers. The addition of dropout in between these two layers has shown considerable overfitting mitigation. Specifically, the recurrent dropout in our model trained

with Tensorflow/Keras uses the approach proposed by [160] and handles the connection between pointwise multiplication and pointwise summation in the input gate shown in Figure 4.8 (b).

Although dropout is a very strong regularizer, adding other regularization methods to our model training can further improve the generalization performance of the model. L1 and L2 regularizers can be used as a part of the model training loss function. L1 regularizer is used to keep the specified model parameter close to 0 whereas L2 regularizer is used to prevent the model parameter from having too high values [26]. The regularizers can be applied on model weights, biases or to the output. In our model training, an L2 kernel regularizer is used on the weights. The L2 kernel regularizer in the loss function is shown in Eq. 4.9.

$$\hat{J}(w; X, y) = J(w; X, y) + \frac{1}{2}\lambda w^T w \tag{4.9}$$

where $\hat{J}$ is the modified loss function, J is the loss function (e.g., mean squared error), w is the weight matrix that will be optimized, λ is the user-defined L2 regularization parameter that determines the intensity of the regularization.

Figure 4.11 illustrates the enhancement in the model predictions with the addition of L2 regularization and dropout. The experiment shown here is used to test run the LabVIEW script with the experimental setup under a closed-loop trial. The slateblue curve is the fitted data with the method mentioned in Section 4.4.1 and the model predictions are supposed to converge to this curve. Orange prediction is the model trained without any regularization method and it can be seen that the predictions have large deviations from the fitted curve. Then, a slight L2 regularization with the λ=0.04 is shown with the dashed blue curve. Between the 7500th and 10000th seconds, there

is a slight improvement; however, the latter stages of the experiment still have significant error margins on the order of 300 ppm. Then, the L2 regularization value is increased to λ=0.08 (dash-dotted blue curve), and even though the noise in the predictions decreases and the trends are better represented, the model overshoots the probable experimental trajectory. Then, the regularization value was increased to a relatively high value at λ=0.15 (dotted blue curve), and this resulted in a drift and increased noise in predictions at the later stages of the experiment. This demonstrates that the low and high values of L2 regularization parameter λ do not improve the model, whereas a suitable fine-tuned λ value boosts the model performance. Finally, the best model with λ=0.08 is further improved with an appropriate percentage of recurrent dropout (red curve) and the dynamic trends are captured as well as the error predictions are greatly reduced. Thus, it is shown that the application of proper regularization parameters significantly augments the model generalization to unseen data. The regularization values presented in this section also generalize well to the other experimental operations (open- and closed-loop experiments).

Remark 20 *The test case in Figure 4.11 is a closed-loop experiment in which both surface potential and rotation are manipulated. Thus, this case is harder to generalize than open-loop experiments for the dynamic behavior.*
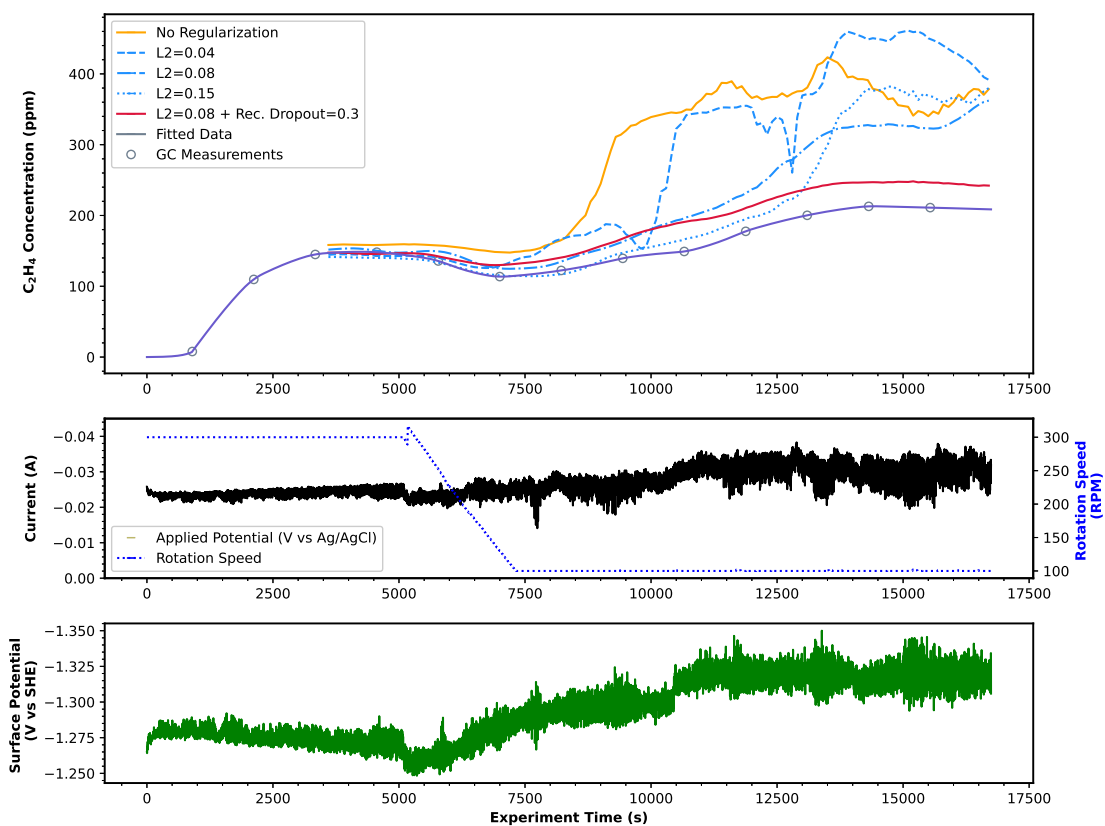
Figure 4.11: The effect of regularization in improving the model predictions. Multiple parameters for L2 regularization are compared to the addition of recurrent dropout and no regularization case.

4.4.5 Implementation of the LSTM Model in Real-Time Operation

The estimators are built to give a prediction each 100 seconds. However, for efficient control, we are expecting a per-second model feedback. Also, if the models were able to yield a per-second prediction, which would require a much larger model than a time window of 36/or 38 data points every 100 seconds, it would be computationally expensive for the LabVIEW program and might cause a time shift for 1 second intervals since there are many calculations being made at the same time and more data being kept in the memory. Therefore, to tackle this problem, a linear regression extrapolation approach is used to estimate 99 time steps between two consecutive LSTM predictions. In this approach, the recent LSTM prediction and the previous one are kept in memory and fitted to a line via linear regression for the previous 100 points. For the next 99 seconds, this linear regression formula is used to extrapolate between 101th to 199th following predictions. With this method, there is a very small loss of accuracy, but a gain of interconnecting points estimation and computational efficiency. Since the 100-second interval is quite small in the development time of the experimental trajectory, the linear nature of this guess does not disturb the modeling of the setup. With this approach, long time window processes can be modeled more accurately using smaller models for many time steps without losing important information.

4.5 MIMO Control Architecture and Controller Tuning

This study aims for the multivariable feedback control using two PI controllers for C_2H_4 and CO concentration by manipulating applied potential and rotation speed.

4.5.1 Using Data for Process Parameter Extraction

The first-order plus time delay (FOPTD) model, as shown in Eq. 4.10 is used to extract the process parameters for the transfer function. Various open-loop experiments were conducted by applying separate step changes to applied potential and rotation speed. After applying the step change, the changes in C_2H_4 and CO concentrations are fitted to a sigmoid function and normalized between 0-1 for easy comparison between various step changes. From those plots, a tangent line from the inflection point is drawn to intersect the X axis. The intersection value on the X axis is defined as the dead time, indicated by θ, and the time required to reach 63% of the final steady state is defined as τ. These parameters are used to extract the FOPTD model shown in Eq. 4.10, where K is the steady state gain and s in the Laplace domain variable of the form:

$$G(s) = \frac{-Ke^{-\theta s}}{\tau s + 1} \tag{4.10}$$

However, electrochemical CO_2 reduction is a highly nonlinear process; therefore, one FOPTD model will not be sufficient to model the entire operating range. Thus, various FOPTD models should be extracted for different surface potential and rotation speed ranges. Our open-loop experiment range might not include an instance of a change between the limits of the designated ranges; thus, we should check general correlations for the dead time and τ for different step changes. If there is a general correlation for the time parameters, it is possible to extract steady state gains from the FNN model. The graphs presented in Figure 4.12 are used to display the dynamic responses of C_2H_4 ((a),(b)) and CO ((c),(d)) under various rotation speeds and rotation speed changes.

Figure 4.12 (a) clearly shows that the concentration change exhibits the same dynamic behav-

158

ior under applied potential step regulations and constant rotation, which means that we can assume the same dead time and τ for the tuning process. After analyzing the second plot, it was seen that similar dead times and τ values are obtained for rotation speeds less than 200 RPM. These values are used in the FOPTD model for C_2H_4. A similar procedure is followed for the extraction of the CO FOPTD model parameters.
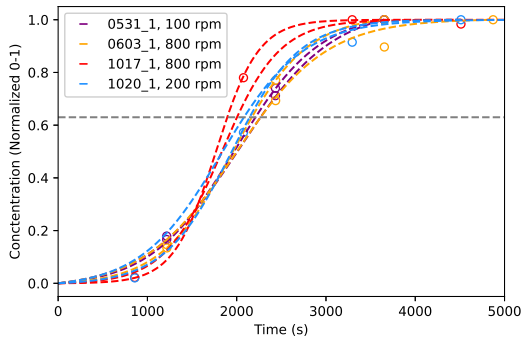
(a) C_2H_4 concentration evolution for potential change under constant rotation speed.
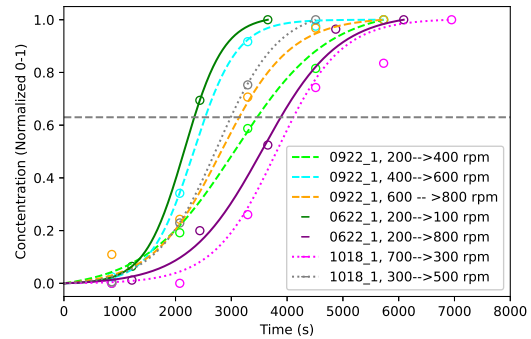
(b) C_2H_4 concentration evolution for rotation speed change under constant applied potential.
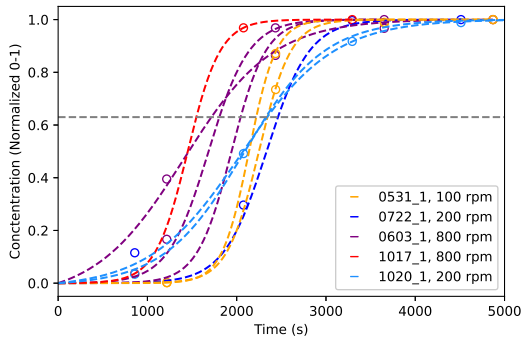
(c) CO concentration evolution for applied potential change under constant rotation speed.
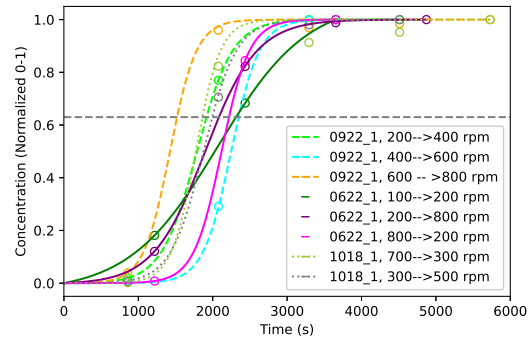
(d) CO concentration evolution for rotation speed change under constant applied potential.

Figure 4.12: The evolution of C_2H_4 and CO concentrations for step changes in the input parameters, fitted to a sigmoid function and normalized between 0-1.

Figure 4.12 shows the dynamic concentration evolutions for C_2H_4 and CO under a step change in one input parameter (applied potential or rotation speed) while keeping the other input parameter
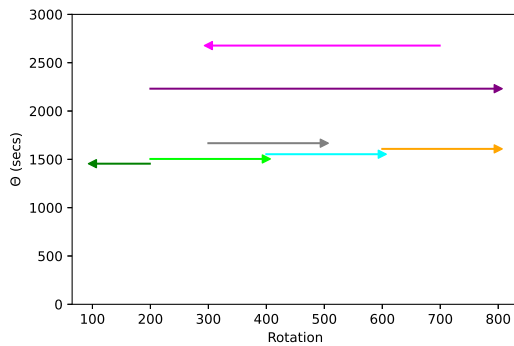
(a) Extracted dead time values for various rotation speed changes in open-loop step change experiments for fitted C_2H_4 dynamic data depicted in Figure 4.12 (b).
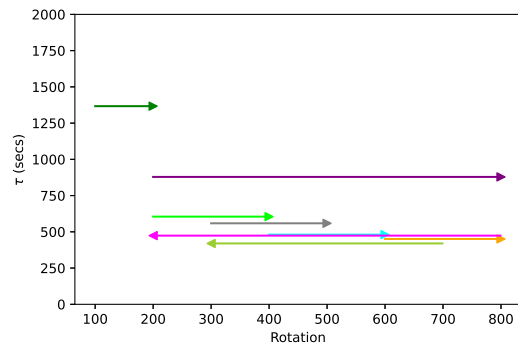
(b) Extracted τ values for various rotation speed changes in open-loop step change experiments for fitted CO dynamic data depicted in Figure 4.12 (d).

Figure 4.13: Process dead time (θ) and time constant (τ) distribution extracted from experimental data for dynamic changes.

constant. For example, in Figure 4.12 (a), the normalized dynamic data fittings for applied potential change under a constant rotation speed overlap with each other for each experiment, and they have the same dead time. In Figure 4.12 (b) and (d), the dynamic data for the change in C_2H_4 and CO concentrations with respect to a variation in electrode rotation speed under a constant applied potential is demonstrated. For C_2H_4, it was seen that for the rotation speed changes under 200 RPM, the dead times are around the same value, and this is shown in Figure 4.13. For CO, the variation in τ values for different rotation speed changes are illustrated in Figure 4.13 (b). For each step change experiment, θ and τ values are extracted from the data and they are averaged for a more generalized results. These values will be used to extract FOPTD models for operational regions. The extracted θ and τ values can be seen in the multivariable control array shown below in Eq. 4.11.

As the time constants and delays for the FOPTD model are obtained, there is now a need to

160

obtain steady state gains. Electrochemical CO_2 reduction process is inherently highly nonlinear, as can be seen in Figure 4.6. This can also be seen from the exponential increase in operational steady states in C_2H_4 concentrations with increasing surface potentials shown in Figure 4.6 (d) and different behaviors of CO concentrations at different rotation speeds in Figure 4.6 (c). Thus, we divide the operation range into 9 regions as the following:

- Potential ranges = (-1.19, -1.26V), (-1.26, -1.30 V), (-1.30, -1.32 V);

- Rotation ranges = (100-200 RPM), (200-400 RPM), (400-800 RPM).

The multivariable control array and relative gain array shown below have the following structure.

$$
\begin{array}{cccc}
& 100:200\ RPM & 200:400\ RPM & 400:800\ RPM \\
-1.19:-1.26\ V\ vs\ SHE & \ddots & \ddots & \ddots \\
-1.26:-1.30\ V\ vs\ SHE & \ddots & \ddots & \ddots \\
-1.30:-1.32\ V\ vs\ SHE & \ddots & \ddots & \ddots
\end{array}
$$

These ranges are expected to satisfactorily linearize the process. The steady state gains are calculated from the FNN model. Using the FOPTD models and different operating windows, we can obtain the multivariable control arrays such as shown below in Eq. 4.11 for the [-1.19, -1.26 V] and [100, 200 RPM] region.

$$
\begin{bmatrix} C_{C_2H_4}(s) \\ C_{CO}(s) \end{bmatrix} = \begin{bmatrix} \frac{-1427e^{-1113s}}{952s+1} & \frac{0.030e^{-1557s}}{1395+1} \\ \frac{919e^{-1409s}}{952s+1} & \frac{0.64e^{-1234s}}{540+1} \end{bmatrix} \begin{bmatrix} E_{surf}(s) \\ \Omega(s) \end{bmatrix} \tag{4.11}
$$

This array is represented in the Laplace domain, and four distinct FOPTD models are used to represent the dynamic process behavior. The first element of the FOPTD matrix in the first row

represents the influence of $E_{surface}$ and the second element in the same row represents the dynamic effects of rotation change on the C$_2$H$_4$ concentration. The second row represents the same impact on CO concentration for the respective inputs. The transfer function matrices in different regions are given below:

$$
\begin{bmatrix}
\begin{bmatrix} \dfrac{-1427e^{-1113s}}{952s+1} & \dfrac{0.030e^{-1557s}}{1395+1} \\[2ex] \dfrac{919e^{-671s}}{1409+1} & \dfrac{0.64e^{-1234s}}{540+1} \end{bmatrix}
&
\begin{bmatrix} \dfrac{-1576e^{-1113s}}{952s+1} & \dfrac{0.14e^{-1557s}}{1395+1} \\[2ex] \dfrac{708e^{-671s}}{1409+1} & \dfrac{0.59e^{-1234s}}{540+1} \end{bmatrix}
&
\begin{bmatrix} \dfrac{-1532e^{-1113s}}{952s+1} & \dfrac{0.21e^{-1557s}}{1395+1} \\[2ex] \dfrac{-407e^{-671s}}{1409+1} & \dfrac{0.23e^{-1234s}}{540+1} \end{bmatrix}
\\[6ex]
\begin{bmatrix} \dfrac{-3985e^{-1113s}}{952s+1} & \dfrac{0.039e^{-1557s}}{1395+1} \\[2ex] \dfrac{808e^{-671s}}{1409+1} & \dfrac{0.49e^{-1234s}}{540+1} \end{bmatrix}
&
\begin{bmatrix} \dfrac{-4413e^{-1113s}}{952s+1} & \dfrac{0.13e^{-1557s}}{1395+1} \\[2ex] \dfrac{945e^{-671s}}{1409+1} & \dfrac{0.72e^{-1234s}}{540+1} \end{bmatrix}
&
\begin{bmatrix} \dfrac{-5158e^{-1113s}}{952s+1} & \dfrac{0.13e^{-1557s}}{1395+1} \\[2ex] \dfrac{1171e^{-671s}}{1409+1} & \dfrac{0.29e^{-1234s}}{540+1} \end{bmatrix}
\\[6ex]
\begin{bmatrix} \dfrac{-5445e^{-1113s}}{952s+1} & \dfrac{0.016e^{-1557s}}{1395+1} \\[2ex] \dfrac{403e^{-671s}}{1409+1} & \dfrac{0.50e^{-1234s}}{540+1} \end{bmatrix}
&
\begin{bmatrix} \dfrac{-5088e^{-1113s}}{952s+1} & \dfrac{0.0029e^{-1557s}}{1395+1} \\[2ex] \dfrac{857e^{-671s}}{1409+1} & \dfrac{0.65e^{-1234s}}{540+1} \end{bmatrix}
&
\begin{bmatrix} \dfrac{-4780e^{-1113s}}{952s+1} & \dfrac{0.019e^{-1557s}}{1395+1} \\[2ex] \dfrac{832e^{-671s}}{1409+1} & \dfrac{0.35e^{-1234s}}{540+1} \end{bmatrix}
\end{bmatrix}
$$

4.5.2 Relative Gain Array

There are two general approaches for the MIMO control system design of the RCE reactor based on the manipulated inputs and controlled output relationships: employing two proportional-integral (PI) controllers or a model predictive controller (MPC). If there are specific input-output couplings that are strong (i.e., multivariable interactions are relatively weak), then it is appropriate to use two PI controllers. In the opposite case, where multivariable interactions are strong, an MPC should be used to regulate the process accounting explicitly for these interactions. To evaluate the strength of the multivariable interactions, a relative gain array (RGA) approach is used. In this work, from the experimental results, we have observed that the electrode rotation speed has a strong effect on the CO concentration, while the surface potential has a very strong effect on the

162

C_2H_4 concentration. RGA is used to quantify these impacts and is calculated as follows [10]:

$$\Lambda = \begin{bmatrix} \frac{k_{11}k_{22}}{k_{11}k_{22}-k_{12}k_{21}} & \frac{-k_{12}k_{21}}{k_{11}k_{22}-k_{12}k_{21}} \\ \frac{-k_{21}k_{12}}{k_{11}k_{22}-k_{12}k_{21}} & \frac{k_{11}k_{22}}{k_{11}k_{22}-k_{12}k_{21}} \end{bmatrix} = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix}$$

where k_{ij}, $i = 1, 2$ and $j = 1, 2$, are the steady state gains in the matrix transfer function models; for example, referring to Eq. 4.11, $k_{11} = -1427$, $k_{12} = -0.030$, $k_{21} = -919$ and $k_{22} = -0.64$. The RGA only needs the steady state gains. If the diagonals of RGA are close to 1, then it means that the surface potential can be used to control the C_2H_4 production and the rotation speed can be used to control the production of CO using two single-loop proportional-integral controllers. The RGAs of the different regions are presented below:

$$\begin{bmatrix} \begin{bmatrix} 0.9704 & 0.0296 \\ 0.0296 & 0.9704 \end{bmatrix} & \begin{bmatrix} 0.9706 & 0.0294 \\ 0.0294 & 0.9706 \end{bmatrix} & \begin{bmatrix} 1.0181 & -0.018 \\ -0.018 & 1.018 \end{bmatrix} \\ \begin{bmatrix} 0.9444 & 0.0556 \\ 0.0556 & 0.9444 \end{bmatrix} & \begin{bmatrix} 0.9610 & 0.0390 \\ 0.0390 & 0.9610 \end{bmatrix} & \begin{bmatrix} 0.9977 & 0.0023 \\ 0.0023 & 0.9977 \end{bmatrix} \\ \begin{bmatrix} 0.9693 & 0.0307 \\ 0.0307 & 0.9693 \end{bmatrix} & \begin{bmatrix} 0.9669 & 0.0331 \\ 0.0331 & 0.9669 \end{bmatrix} & \begin{bmatrix} 0.9908 & 0.0092 \\ 0.0092 & 0.9908 \end{bmatrix} \end{bmatrix}$$

Since the RGA diagonal elements for each region have values very close to 1, we can infer that two PI controllers are sufficient to control this process, where one of the PI controllers will manipulate the applied potential to control C_2H_4 while the other PI controller will manipulate the

rotation speed to control CO production rate. These RGA conclusions hold true for a broad set of steady state values of the inputs in the operating region of interest; this analysis was carried out in [19] for a similar experimental reactor setup and will not be repeated here. Overall, it is noteworthy that the RGA results based on our lab-scale experimental reactor provide a quantitative basis for the strength of these interactions in a scaled-up reactor based on the design employed in our current process.

4.5.3   Controller Tuning

Two PI controllers are used for the feedback control of the experimental RCE reactor. Since the FOPTD transfer function array is obtained, the controller tuning parameters can be calculated for both PI controllers using the Cohen-Coon technique. Specifically, the FOPTD models are initially used to extract 9 pairs of controller parameters with the Cohen-Coon tuning method. The details of Cohen-Coon tuning method can be found in [58]. However, employing 9 pairs of controller parameters might be complicated, thus, proportional gain values that are in the vicinity of each other are averaged to reduce the number of controller parameters. After having the preliminary proportional gains and integral time constants, MATLAB is used to simulate a setpoint change to fine-tune the proportional gains taking the Cohen-Coon estimations as starting points. The fine-tuned control parameters are presented in Table 4.2 and the simulation results for different regions are presented in Figure 4.14. The integral time constants (τ_i) from Cohen-Coon are 1250 and 1000 seconds, respectively.

However, there is a unit matrix decoupler defined in the MATLAB tuning procedure, meaning that the simulator does not take the cross-coupling effects into account (e.g., rotation effects
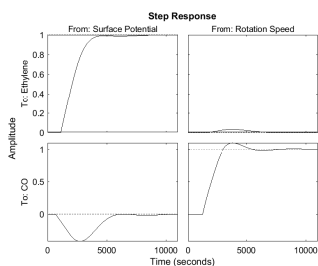
(a) -1.19 to -1.26 V vs SHE and 100-200 RPM.
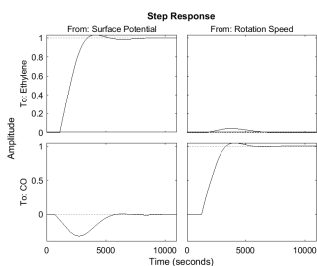
(b) -1.19 to -1.26 V vs SHE and 200-400 RPM.
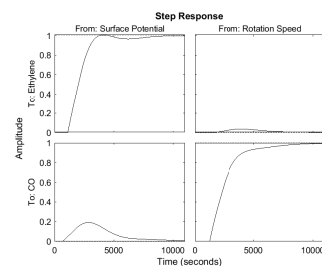
(c) -1.19 to -1.26 V vs SHE and 400-800 RPM.
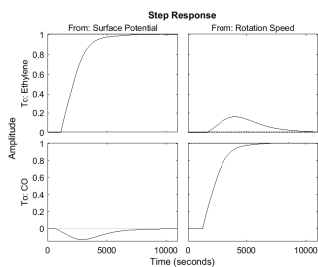
(d) -1.26 to -1.30 V vs SHE and 100-200 RPM.
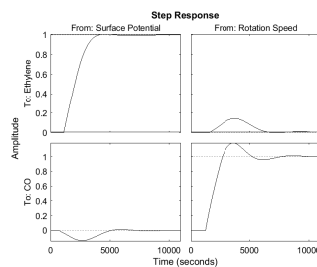
(e) -1.26 to -1.30 V vs SHE and 200-400 RPM.
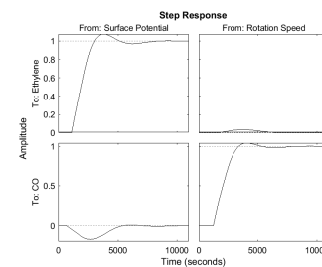
(f) -1.26 to -1.30 V vs SHE and 400-800 RPM.
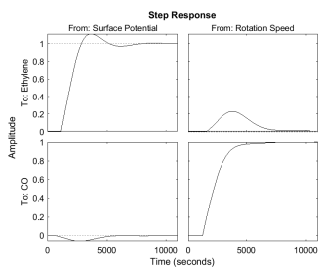
(g) -1.30 to -1.32 V vs SHE and 100-200 RPM.
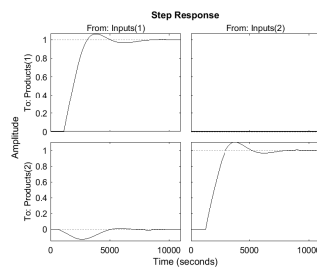
(h) -1.30 to -1.32 V vs SHE and 200-400 RPM.
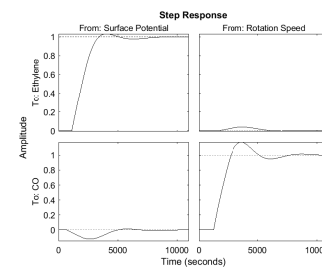
(i) -1.30 to -1.32 V vs SHE and 400-800 RPM.

Figure 4.14: MATLAB-based tuning of the PI controllers using a decoupler.

on C_2H_4 concentration). At some specific operation ranges, the cross-coupling effects might affect the performance of the closed-loop system. For example, as discussed in Section 4.3.1, the effect of surface potential change affects the CO concentration at lower rotation speeds. When the MATLAB-tuned parameters are tested in the experimental setup, it was seen that both of the parameters have huge fluctuations around the steady state setpoints. This means the controller proportional gains are not tuned to handle the cross-parameter effects, and they should be further

165

improved.

It is very time-consuming and costly to fine-tune the controllers on the experimental setup by trial and error, especially when multiple pairs of control parameters are involved. To address this problem, the dynamic RNN models trained for C_2H_4 and CO dynamic evolution modeling are used as the digital model of the experimental reactor to investigate the controller performance under numerous pairs of controller parameters in different regions. To enable this simulation, the instance current is required as the input for the estimation of C_2H_4. Thus, the correlation presented in Eq. 4.4 is used to approximate the current at a specific surface potential. However, this case is idealized compared to the real system because the current magnitude is largely affected by the resistance of the electrolyte solution and experimental conditions might also affect the concentration outputs.

Two PI controllers are added to the simulation. The first controller manipulates the surface potential to control C_2H_4 production rates while the second controller manipulates the rotation speed to regulate the CO concentration in the reactor overhead. It was seen that if both controllers are tuned with values in the vicinity of MATLAB-tuned proportional gains, both of the setpoints suffer from big fluctuations due to cross-coupling effects, and it would take a very long time for both fluctuations to subside. Thus, the case that worked best and fastest to reach the steady state is having one of the controllers with a high proportional gain and the other controller with a low proportional gain. Specifically, the C_2H_4 controller is selected to be the more aggressive one while the CO controller is adjusted to drive the process to the setpoint slowly. Therefore, the MIMO control scheme aims to drive the C_2H_4 concentration to the setpoint quickly; and after it is reached, the second PI controller drives the CO to the setpoint slowly. While C_2H_4 is going to the setpoint, CO could even go in the opposite direction with respect to its setpoint because big changes in the

166

Table 4.2: Controller gains (K_p) resulting from MATLAB-based tuning.

		RPM		
		100 - 200	200 - 400	400 - 800
V vs SHE	-1.19 : -1.26	-0.00038		
	-1.26 : -1.30	-0.000126		
	-1.30 : -1.32			

(a) Applied potential controller.

		RPM		
		100 - 200	200 - 400	400 - 800
V vs SHE	-1.19 : -1.26	0.69		1.4
	-1.26 : -1.30			
	-1.30 : -1.32			

(b) Rotation speed controller.

surface potential could negatively affect CO, especially at low rotation speeds. However, once the surface potential is stabilized at the new setpoint, the rotation speed is adjusted slowly, while the small cross rotation effects on C_2H_4 are compensated by the aggressive potential controller. A control case simulation scenario with the new control parameters is shown in Figure 4.15. The final proportional gains obtained from this simulation are presented in Table 4.3.

Table 4.3: Final controller gains (K_p) resulting from closed-loop system simulation.

		RPM		
		100 - 200	200 - 400	400 - 800
	-1.19 : -1.26	-0.00042		
V vs SHE	-1.26 : -1.30	-0.00025		
	-1.30 : -1.32			

(a) Applied potential controller.

		RPM		
		100 - 200	200 - 400	400 - 800
	-1.19 : -1.26			
V vs SHE	-1.26 : -1.30	0.25		0.35
	-1.30 : -1.32			

(b) Rotation speed controller.

Figure 4.15: Closed-loop simulation using the dynamic RNN models and the empirical surface potential-current correlation used to determine the final PI controller parameters.

4.5.4 Estimator Design Using GC Measurements

In Section 4.2.2, the standard deviation in the experimental results is discussed under nearly-identical surface potential and rotation speed for relatively short duration experiments of less than 2 hours. The experimental conditions mentioned in Remark 15 also affect the variability in the experimental performance observed for individual experiments. Therefore, the LSTM models built using the experimental training data are expected to give a prediction that is compatible with the averaged experimental output and within the standard deviations, however, this prediction still needs to be improved for feedback control purposes. To accomplish this fine-tuning of the control of the reactor, the GC sensor feedback is incorporated into the prediction model and the control scheme.

In Chapter 2, we offered an approach to introduce the GC measurements incurring a 14.3 minutes delay into the concentration estimation and feedback control scheme. However, that approach took advantage of a hybrid model and mass balance equations, calculated the error in machine learning-based reaction rate estimation and corrected the estimator predictions accounting for the possible corrections in pastime instants of the experiment. That approach is not applicable here due to LSTMs being black box models. Instead, taking advantage of the ability of RNNs to learn time series and trends in complex data series, we can scale the RNN predictions with respect to the recent GC measurement and adopt our RNN model to the real experimental trajectory without violating the trends. This is a suitable solution built on the understanding that the catalyst activity may vary from experiment to experiment, while the fundamental mechanism for the transformation of CO_2 on the copper electrodes is unchanged. The RNN model predicts the concentrations

within a standard deviation, and captures the dynamic timing (e.g., dead time and τ) satisfactorily well. Thus, scaling the model does not affect the dynamic trends. To scale the RNN outputs, the prediction from 14.3 minutes ago is kept in the control loop, and it is compared to the GC measurement when the injection is completed. As soon as the GC results are obtained, the GC result is proportioned to the RNN prediction from 14.3 minutes ago and a scaling factor α is obtained. Then, the current RNN predictions are multiplied by the scaling factor until a new scaling function is obtained. This is described as follows:

$$\alpha^i = \frac{GC^i}{F^0_{RNN}(t - 14.3\,min)} \tag{4.12a}$$

$$F_{RNN}(t) = \alpha^i \times F^0_{RNN}(t) \tag{4.12b}$$

where i is the number of GC injections, GC^i is the ith GC injection, F^0_{RNN} is the prediction of the unscaled RNN model and F_{RNN} is the prediction of the scaled RNN model.

This GC feedback correction is not applied after each injection. Applying a scaling factor during a big setpoint change might interfere with the model dynamics, thus we scale the RNN predictions when the consecutive GC variation is small (e.g., at steady state). When the difference between the GC measurement and RNN prediction is substantial, the scaling factor is applied when the concentrations are near a steady state, the model predictions and GC measurements are scaled to be coherent, and the system is driven to the setpoint after the scaling. This correction is applied if the consecutive GC measurements are within 8% or 10 ppm vicinity of each other. The application of the scaling factor is automated on the LabVIEW interface. The final control diagram of the process is shown in Figure 4.16.
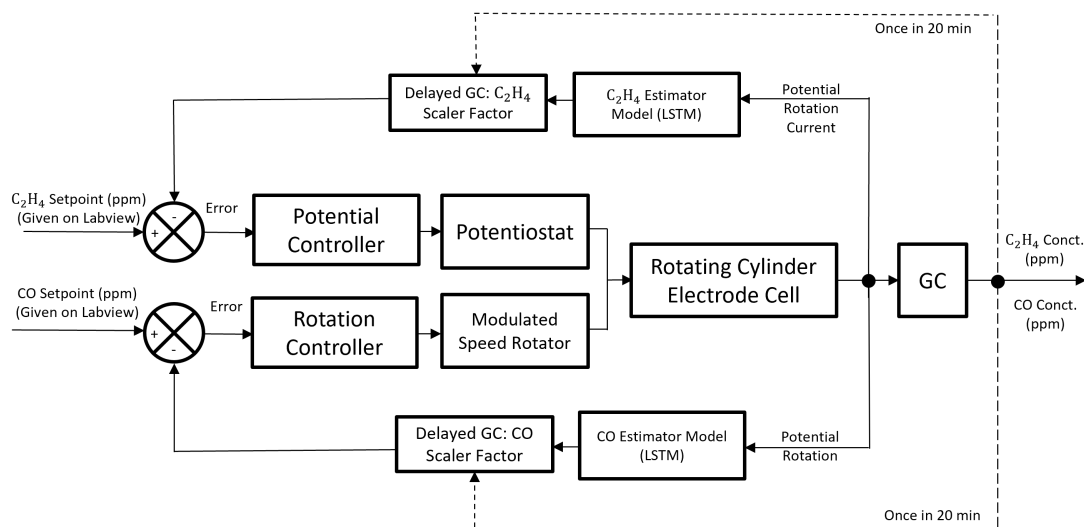
Figure 4.16: Closed-loop system structure using multi-input multi-output control system with ML estimators.

4.6   Closed-loop Experimental Results and Discussion

Three scenarios are considered for the demonstration of MIMO control in this study. The first scenario aims to drive the process to the most profitable setpoint under the base case, which has a selectivity of 1:4 for the C_2H_4:CO concentration ratio. The second scenario drives the selectivity ratio to 1:1 to show the flexibility of the control system by increasing the C_2H_4:CO ratio by 4-fold. This versatility could be advantageous considering a potential integration of a downstream process such as the copolymerization of ethylene and carbon monoxide, where the ratio of the two monomers impacts the structural and thermophysical characteristics of the synthesized polyketone [16, 167]. The third scenario simulates a case based on sensitivity analysis in which the prices of C_2 and C_{2+} products increase by 60% while other chemical prices decrease by 40%. The third scenario drives the C_2H_4:CO concentration ratio to 4:5 which is very close to the 1:1 ratio and thus serves to challenge the accuracy of the control system for close but different concentration ratios

172

over a broad range of concentrations. The three scenarios demonstrate the ability of controllers to implement large increases and decreases in selectivity ratios, as well as precise control of product concentrations. In each experiment, the closed-loop controller is activated at 4500th second and the same control parameters are used for all scenarios. Each case starts from the same initial applied potential and electrode rotation speed of -1.67 V vs Ag/AgCl and 300 RPM, usually yielding a surface potential of around -1.27 V vs SHE. This initial point is chosen to have both the rotation and potential values close to the region where the controller parameters change to observe the effects of changes in controller gains.

4.6.1   Experimental Results

Setpoints for C_2H_4 and CO under the first scenario are calculated from the real-time optimizer to be 112 and 490 ppm. The experimental results for the first case scenario are shown in Figure 4.17. It is noteworthy to compare these experimental results to those obtained in the simulated closed-loop experiments shown in Figure 4.15 that utilized identical setpoints. Due to the control strategy mentioned in Section 4.5.3, the controller first drives C_2H_4 to the setpoint while the electrode rotation speed increases gradually, which slowly drives CO to its setpoint. As shown previously in Figure 4.15, this strategy can effectively compensate for the rotation-driven deviation in C_2H_4 concentrations. In this case, to decrease the ethylene concentration, the applied potential is decreased. This also has the effect of increasing slightly the CO concentration at low rotation speeds (see Figure 4.6 (c)). Thus, cross-coupling effects on the input-output relationship in this specific setpoint are helpful in achieving the control objective. The initial increase in CO is mostly due to the decrease in surface potential. In this region, until the rotation reaches 400 RPM, the ef-

173

fect of rotation speed is kept minimal with the CO controller using a small proportional gain. Once the ethylene concentration reaches its setpoint, the rotation speed is increased more aggressively to eventually drive the CO concentration to its setpoint.

The experimental trajectory in Figure 4.17 is very similar to the simulated result in Figure 4.15. The main two differences are that in the simulation, the setpoints are reached earlier than in the actual experiment and the estimation of a setpoint overshoot for the CO concentration, which is not observed in the experiment. These differences are naturally due to the fact that experimental observations are different from model prediction and feedback corrections, and highlight the importance of coupling simulations with experiments. The simulation also assumes that the current is well-described by Eq. 4.4. However, experimentally, the current might deviate from this correlation extracted from open-loop experiments for catalysts operated over relatively short durations, causing slightly different product concentrations. Likewise, when a scaling factor is applied as a part of the feedback correction, the error between the model prediction and the setpoint change must be recalculated and results in some delays which do not exist in the simulation and could have an effect on the experimental observations. The pure model predictions might be very close to the setpoint, which would reduce the actuation. However, when the scaling factor is applied, the model receives feedback when the predictions were off (e.g., at $10 \ ppm$) and the controller drives the corresponding input at higher values to reach the setpoint. Thus, this causes a delay in the setpoint tracking, which can also explain the timing difference between the experiment and the simulation. It is noted that these timing differences are still minor, and the system is still driven to the desired multiple setpoints in a short period of time of just a few hours.

The objective of the second scenario is to adjust the selectivity ratio to 1:1 for a concentration

174

of 200 ppm for each product. The experimental results for this scenario are shown in Figure 4.18. Here, the initial ethylene concentration is closer to 200 ppm, and it requires only a small increase in applied potential to reach the setpoint. On the other hand, CO starts from a relatively high concentration compared to the setpoint at 200 ppm. The initial surface potential activates the higher K_p and the controller starts increasing the surface potential further. At the same time, the error between CO concentration and its setpoint is high, thus the electrode rotation speed continuously decreases resulting in a monotonic decrease in the CO concentration. As the rotation speed further decreases below 200 RPM, the potential effects become more important for CO. When C_2H_4 reaches its setpoint and the controller starts to decrease the surface potential, the decrease in CO concentration loses momentum as the potential effects kick in, increasing the production of CO. To account for this, the MIMO controller reduces the rotation speed more strongly after around 13000 seconds to overcome the effects of fluctuating surface potential until both setpoints are eventually reached after around 20000 seconds of the start of the experiment.

In the third scenario, the system is driven to a selectivity ratio of around 4:5 as shown in Figure 4.19, where the setpoints for C_2H_4 and CO concentrations are 283 and 350 ppm, respectively. In this experiment, the initial CO concentration is already close to its setpoint, whereas the C_2H_4 concentration is less than half that of its setpoint. Here, the C_2H_4 controller steadily increases the surface potential to decrease the error between the setpoint and the estimator model predictions. Although the CO controller slowly increases the rotation speed to raise the CO concentration, the strong increase in surface potential decreases the CO concentration. Because this surface potential increase occurs while the rotation speed is still at relatively slow rates, the surface potential effect is influential on the CO concentration. The fast increase of the surface potential causes a small

175

overshoot in the C_2H_4 concentration but this is then corrected by slowly decreasing the applied potential and driving C_2H_4 slowly to its setpoint. After this task is achieved, the rotation is increased to drive CO to its desired setpoint.

In all 3 experiments, different setpoints are reached for both C_2H_4 and CO, thus experimentally demonstrating that the developed control scheme is efficient. It must not be lost in the technical description of this chapter the fact that this is the first time that this level of control is achieved for multiple products in a CO_2 electrochemical RCE reactor. The MIMO controller demonstrated here delivers on the promise of RNN-based modeling frameworks for the control of advanced reactors driven purely by electricity. This should motivate further integration of RNN-based modeling in the research and development of electrified chemical and fuels manufacturing technologies.
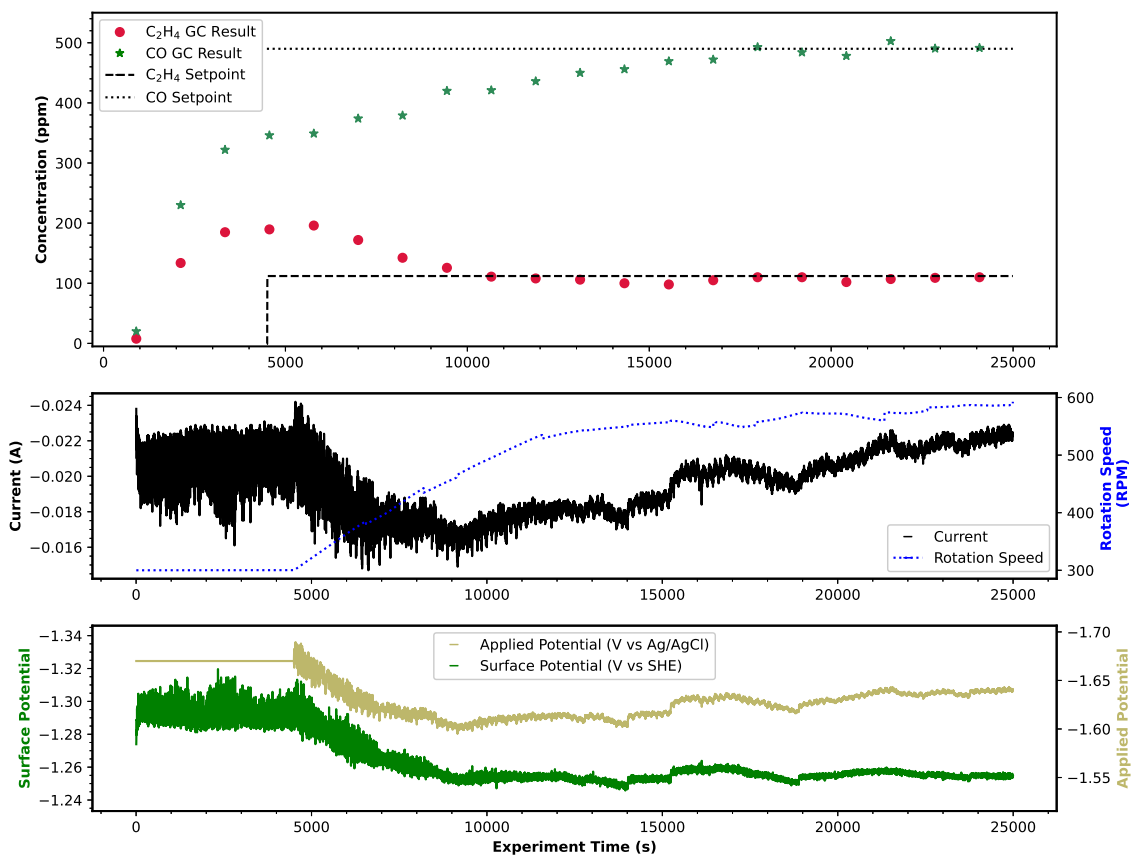
Figure 4.17: Closed-loop experimental results for the economically optimal setpoints for which the C_2H_4:CO selectivity ratio is 1:4.
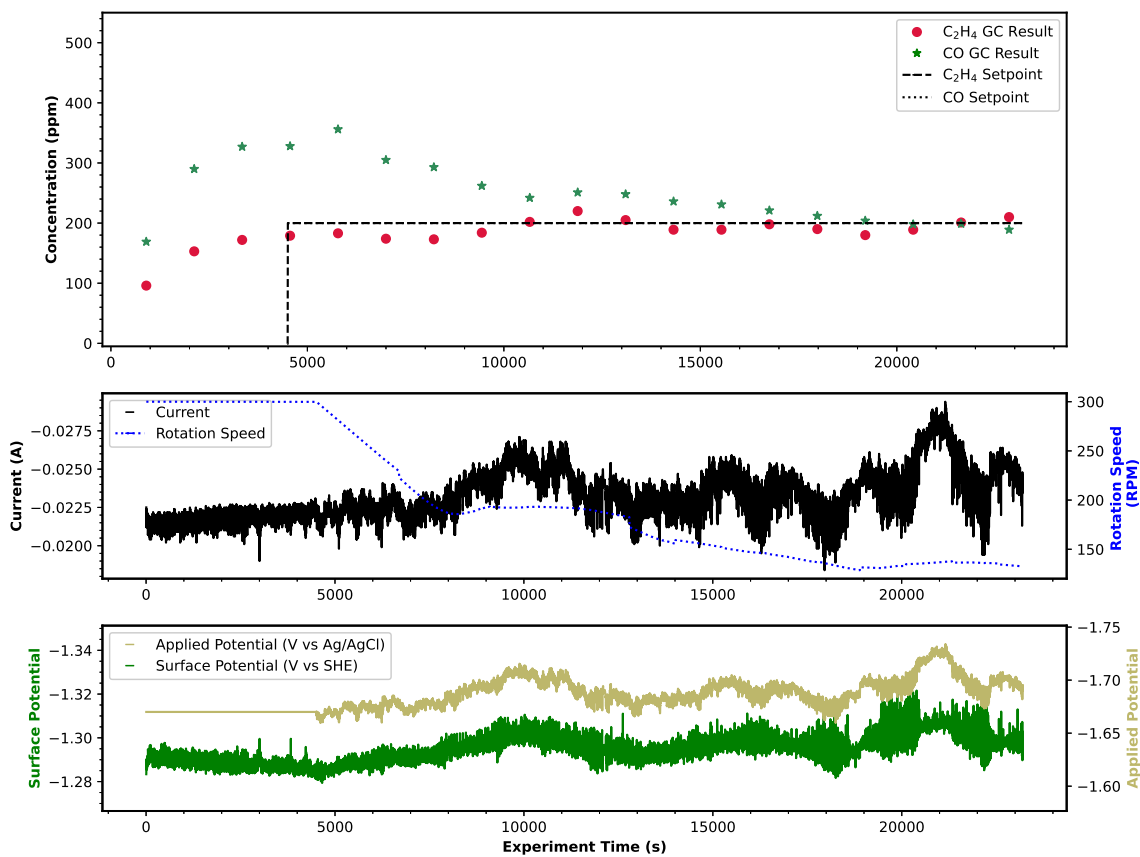
Figure 4.18: Closed-loop experimental results for the C_2H_4:CO selectivity ratio of 1:1 which results in a higher selectivity towards ethylene compared to the economically optimal case.
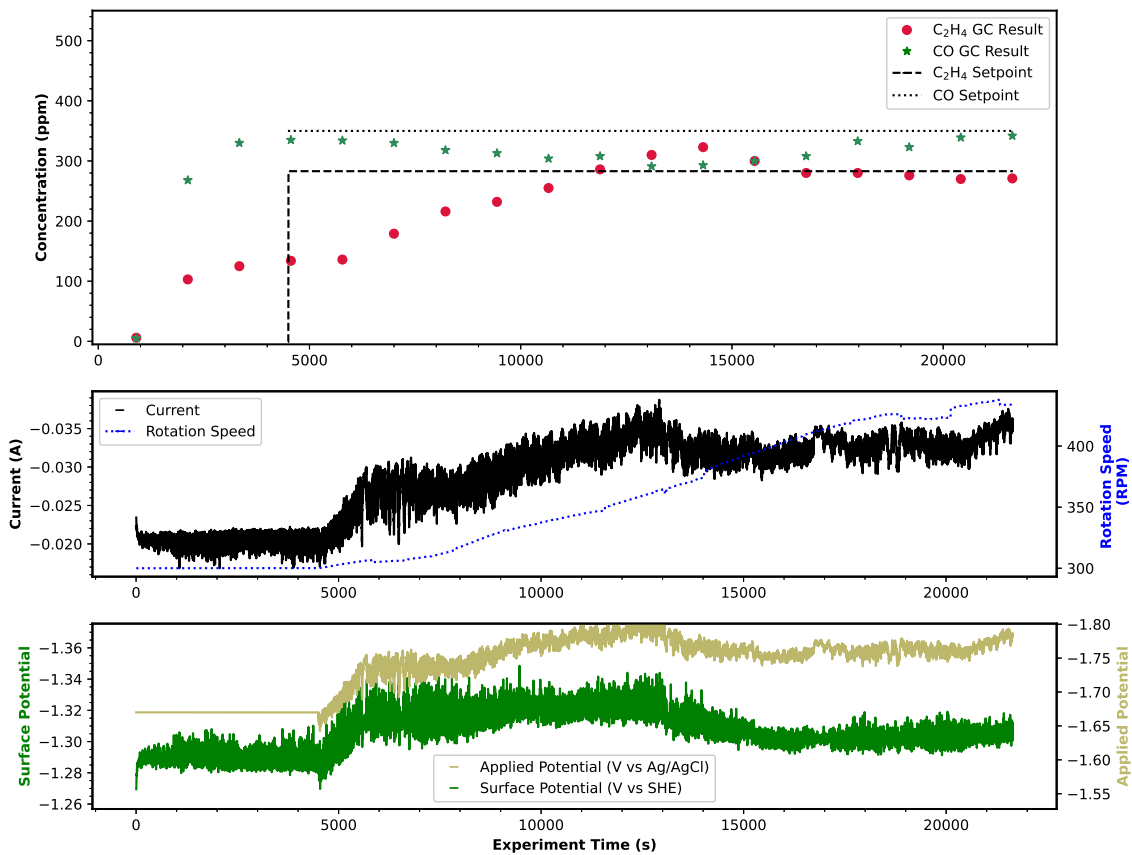
Figure 4.19: Closed-loop experimental results corresponding to setpoints with C_2H_4:CO selectivity ratio of 4:5.

4.6.2 Experimental Errors in Closed-loop Experiments and Outlook

Each closed-loop experiment started from the same initial conditions of applied potential and rotation and yielded 189, 172 and 132 ppm for C_2H_4 and 346, 328 and 335 ppm for CO after reaching the first steady state at around 5000 seconds from the start of the experiment. This highlights the experimental difference in each of the catalysts freshly prepared, and demonstrates the need for a feedback corrector to adjust the model predictions to the GC points for each individual experiment.

Furthermore, there are shifts in current. For example, if the system generates less current in a new experiment at the same surface potential compared to the base case steady state experiments, it is very probable to obtain less ethylene. These dynamics are captured by the RNN model as many case scenarios are fed to the model during the training. One of the reasons for this current shift might be the lack of Ohmic drop compensation feature on the LabVIEW interface. As mentioned in Remark 15, the electrolyte solution is measured in the beginning of the experiment. However, the resistance of the electrolyte solution might increase or decrease as the experiment proceeds, and the surface potential calculations are affected by these resistance shifts. It would be necessary to program into the potentiostat controller an algorithm that, without changing the applied potential, performs electrochemical impedance spectroscopy or a current interrupt analysis during a few seconds to measure the evolving solution resistance. This can be the focus of future works as we expand the MIMO controller to account for more operation variables and products.

Chapter 5

Machine Learning-Based Predictive Control Using On-line Model Linearization: Application to an Experimental Electrochemical Reactor

5.1 Introduction

In today's chemical manufacturing industry, fossil fuels serve as the primary energy source for the chemical industry, leading to significant energy consumption and greenhouse gas emissions [14]. Alternatively, there has been increasing interest in electrochemical reactions, such as converting carbon dioxide (CO_2) into carbon-based fuels and chemicals with electricity, as a means to mitigate CO_2 emissions. This approach holds the potential for leveraging electricity generated from renewable resources as an energy source for large-scale chemical manufacturing, furthermore

contributing to global-scale renewable energy storage and closing the anthropogenic carbon cycle [33]. Although electrochemical conversion of waste CO_2 is very promising, several challenges hinder the widespread adoption of electrochemical reactors on an industrial scale. Perhaps most importantly, the conversion of CO_2 through electrochemistry requires significant energy consumption [171]. Researchers have focused on improving energy efficiency in electrochemistry through the development of more efficient and selective catalysts through nanostructuring, doping of transition metals, utilization of single-atom catalysts, etc. [120, 99, 83] as well as the design of devices to reduce the overall cell potential and address parasitic carbonation problems [213, 222, 140]. On the other hand, discussions on process scale-up have been limited so far [146]. We have identified another critical challenge in scaling up electrolyzers to be the absence of advanced process control schemes for electrochemical reactors due to the complex and nonlinear nature of electrochemical processes. Since the realization of an economically viable electrochemical process will require optimization in process integration and cascade reactor train [138, 125, 42], the development of a control scheme to regulate individual electrochemical reactor units is necessary.

To address this issue, Chapter 2 proposed a feedback control scheme using proportional-integral (PI) controllers utilizing a support vector regression-based (SVR) hybrid model as a state estimator. This approach enabled real-time state estimation for a PI controller and, subsequently, implementation of single-input single-output (SISO) control in a gastight rotating cylinder electrode (RCE) cell. Building on this work, Chapter 4 introduced a recurrent neural network (RNN) model as an improved state estimator, surpassing the performance of the SVR model. This RNN model captured relationships between process variables and gas product concentration and allowed for the implementation of multi-input-multi output (MIMO) control using PI control techniques for

the same RCE reactor. Alongside the classical control strategies, model predictive control (MPC) methods have emerged as vital components in industrial process control design [135, 97]. MPC offers the advantage of computing optimal control actions by anticipating future output states, making it a powerful tool for multivariable control while considering process constraints and nonlinearities, for example, [66].

Although the specific application of MPC in electrochemical reactors is limited, MPC has been widely used in various research areas, including chemical reactors, battery management, and self-driving cars. For instance, [145] provided a comprehensive discussion on implementing MPC for a crude oil distillation unit in the petroleum industry. Furthermore, [20] explored MPC design for a multivariable distillation column, demonstrating superior performance compared to PI-based control through MATLAB simulations using the Wood and Berry Model. MPC has also been applied to develop a battery management system, which is similar to the application of an electrochemical reactor in the sense that both tasks involve manipulating electrochemical reactions, even though the battery management system focuses on storing and releasing electricity instead of generating products using electrical potential. For example, [132] proposed a nonlinear MPC design based on the electrochemical models capturing the internal phenomena of the battery to solve the charge unbalancing problem in lithium-ion cells connected in series. These applications have demonstrated the ability of MPC to control systems with electrochemical reactions. Considering the advantages of MPC over classic control strategies (e.g., PI control) with respect to explicitly handling actuator and state constraints, multivariable interactions and nonlinearities, it is potentially practical and valuable to leverage the application of MPC to control electrochemical reactors.

Implementing MPC in electrochemical reactions poses two major challenges that need to be overcome: model accuracy and computational expense. The accuracy of the model prediction is crucial for the performance of MPC. Ideally, a first-principles-based model that accurately captures the underlying phenomena in the electrocatalytic system would be optimal. However, such models are often unavailable for practical cases. To this end, in our research, we focus on a data-driven approach to model the process system. Data-driven modeling offers a systematic approach that can be applied to any process system if sufficient data quantity and quality are ensured. One of the significant examples of data-driven modeling is machine learning (ML), which is a class of techniques that can be generally applied to various systems without the need for formulating specific physical patterns discovered in experiments [36]. Classical ML methods, including SVR, linear regression, Gaussian process regression, and decision trees, have been widely utilized for modeling tasks [209, 166, 11, 64, 221]. Additionally, deep learning methods, employing neural network (NN) structures, have demonstrated superior performance in capturing nonlinear and complex systems compared to classical ML methods. As a result, NN modeling has drawn significant attention and has been applied in recent research works [123, 161]. Considering the nonlinearity and complexity of the electrochemical reactor and to facilitate the modeling process, a NN method is utilized to model the system, which has been demonstrated to be an effective technique for this specific reactor in Chapter 4.

While the use of nonlinear data-driven models in MPC has shown promising performance in various research studies, implementing MPC with a nonlinear model generally involves solving a non-convex optimization problem. This complexity often results in high computational costs and unstable gradient concerns. [214] demonstrated in their work that using RNN models in

MPC can be highly accurate yet intractable to solve in real-time, motivating the exploration of linearization approaches to improve the computational efficiency of MPC. Several techniques have been proposed for linearizing nonlinear models, such as Taylor series, piecewise linearization, etc. [156, 102, 96]. Specifically, the Koopman operator method is developed to be a data-driven approach that can be applied to any nonlinear model [90, 134, 4]. [214] showed the Koopman operator method is a type of linearization approach that can have better performance than the classical Taylor series method, particularly when linearizing over a larger domain. Furthermore, the application of MPC using a linearized model has been studied in-depth in the chemical engineering domain [e.g., 112, 76]. These results have highlighted the potential of employing MPC with on-line linearized models in practical control applications. By leveraging efficient linearization techniques, NN-based MPC can potentially be applied to control the electrochemical reactor effectively.

Motivated by the above considerations, this study aims to develop an advanced process control scheme using MPC with suitable process models for an electrochemical CO_2 reduction reactor. Specifically, a neural network model is initially constructed using reactor data to capture the nonlinear complex input-output relation of the reactor, followed by on-line linearization of the NN model using the Koopman operator method to reduce the computational cost of MPC. The control design is applied experimentally to the electrochemical reactor. This chapter is organized into the following sections: Section 5.2 introduces the background information of this study, including the mathematical notation used in this chapter, the overall design of the process, and the equipment setup. Section 5.3 elaborates on the technical details for the design and development of a NN model. Section 5.4 discusses the Koopman operator method and the procedure of using it to lin-

earize the NN model in real-time. Finally, Section 5.5 reports the results of this study including simulation, open- and closed-loop experiments.

5.2 Preliminaries

5.2.1 Notation

For a matrix M, the notation M^{-1} is used to represent the inverse of the matrix M and $M^{\dagger}$ denotes the pseudoinverse of matrix M. x, $\hat{x}$, and u are the controlled outputs of the process control system (i.e., the productivity of the reactor for the targeting species), the prediction of the process output given by the process model (i.e., the NN model), and the inputs (control actions) calculated by the process control system (i.e., applied potential, rotation speed, and current), respectively.

5.2.2 Process Overview

The overall objective of our process is to electrochemically reduce CO_2 into valuable chemical products and fuels. A copper electrode is used in this process because it is the only known single-element catalyst that can reduce CO_2 into C_{2+} hydrocarbons and alcohol products, which are energy-dense and valuable, with a considerable production rate [131]. However, the process of electrochemical CO_2 reduction on copper is intricate, which results in the production of 17 different chemicals through a series of complex reaction pathways [120]. Among multiple factors contributing to the complex reaction mechanisms, mass transport and reaction kinetics play critical roles. Specifically, the transport phenomena in the diffusion boundary layer are directly related to the residence time of the reactant CO_2 and intermediates near the catalyst surface as well as the

adsorption on and desorption of the catalyst, which determine the selectivity of final products. On the other hand, the reaction kinetics on the catalyst surface is related to the number of electrons transferred to the surface, which can be manipulated by the applied potential. Therefore, we aim to control the selectivity of electrochemical CO_2 reduction by controlling the aforementioned two input factors, potential and electrode rotation speed. Applying real-time control to any process requires on-line measurements of the process outputs. In this work, the productivity of four gas-phase products (i.e., hydrogen (H_2), carbon monoxide (CO), methane (CH_4), and ethylene (C_2H_4)) can be monitored in real-time using a gas chromatograph (GC). The overall reaction formulas producing these four products are summarized as follows:

$$2\,CO_2 + 8\,H_2O + 12\,e^- \longrightarrow C_2H_4 + 12\,OH^- \tag{5.1a}$$

$$CO_2 + H_2O + 2\,e^- \longrightarrow CO + 2\,OH^- \tag{5.1b}$$

$$CO_2 + 6\,H_2O + 8\,e^- \longrightarrow CH_4 + 8\,OH^- \tag{5.1c}$$

$$2\,H_2O + 2\,e^- \longrightarrow H_2 + 2\,OH^- \tag{5.1d}$$
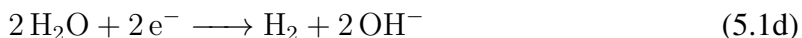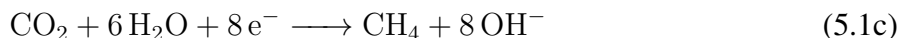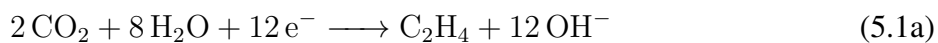
Finally, the production rates of CO and C_2H_4 are chosen to be the control outputs to be regulated by the process control system. These two outputs are influenced differently by the input variables; specifically, the production rate of CO is highly correlated to the rotation speed, and the production rate of C_2H_4 is strongly influenced by the applied potential, as discussed in Chapter 4.

5.2.3 Electrochemical Reactor Setup

The gastight RCE cell was designed to examine how mass transport and reactions kinetics affect the electrochemical reduction of CO_2 while ensuring a gastight environment for the real-time detection of gas products [73]. As shown in Figure 5.1, the experimental reactor consists of two reaction chambers separated by an anion-exchange membrane preventing the crossover of products. The cathode is the working electrode in the cylindrical geometry carrying out the CO_2 reduction reaction, while the Pt foil anode works as the counter electrode. Before each experiment, polycrystalline Cu RCE was mechanically and electrochemically polished following the procedure described in [73] followed by roughening of the surface via electrochemical redox cycling in the presence of chloride ions [149]. The preparation for this catalyst is the same as in Chapter 4. Both the working and the counter electrodes are immersed in 0.2 M potassium bicarbonate electrolyte solutions. During the experiment, the CO_2 gas is directly bubbled into the electrolyte in both chambers with a fixed volumetric flow rate of 20 mL/min. Subsequently, the dissolved CO_2 molecules are transported to the reacting surface on the cathode to be reduced to various products. The potentiostat manipulates the potential applied to the working electrode against a reference Ag/AgCl electrode and records the electrical current passed between the working and the counter electrode. The control of the mass transport properties in the reactor is made possible by magnetically coupling the shaft where the RCE is mounted to another magnet connected to the modulated speed rotator (MSR) outside the reactor.
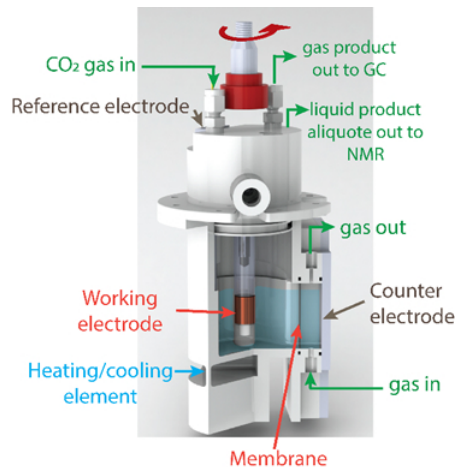
Figure 5.1: The experimental setup of the gastight rotating cylinder electrode (RCE) cell.

5.2.4 Model Predictive Control

MPC is an advanced control strategy used in various industrial processes. It involves utilizing a dynamic mathematical model of the system to predict its future state or output behavior and optimize control actions by iteratively solving an optimization problem over a defined time horizon. Specifically, MPC determines the optimal control actions to minimize a specified cost function while satisfying system constraints. The design of the MPC in this work can be mathematically

defined as the following optimization problem:

$$\mathcal{J} = \min_{u \in S(\Delta)} \int_{t_k}^{t_{k+N_h}} L(\hat{x}(t), u(t)) \, \mathrm{d}t \tag{5.2a}$$

$$\text{s.t.} \quad \hat{x}(t) = F_{nn}(x(t), u(t)) \tag{5.2b}$$

$$L(\hat{x}(t), u(t)) = (\hat{x}(t) - x_r)^\top Q(\hat{x}(t) - x_r) + (u(t) - u_r)^\top R(u(t) - u_r) \tag{5.2c}$$

$$u(t) \in U, \ \forall\, t \in [t_k, t_{k+N_h}) \tag{5.2d}$$

$$\hat{x}(t_k) = x(t_k) \tag{5.2e}$$

$$|u(t_k) - u(t_{k-1})| \le u_c \tag{5.2f}$$

where x and u in R^m are the output states and control actions (calculated by the model predictive control system), respectively. The set U represents the control action space that defines the upper and lower bounds of the m control actions applied to the reactor. The absolute difference between the control actions to be applied in the next control period from the instance time $u(t_k)$ and control action applied in the current control period $u(t_{k-1})$ is bounded by the vector u_c containing absolute boundaries for m control actions (in this particular case, $m = 2$ as we have two manipulated inputs and dimension of $u(t_k)$ is 2). Furthermore, 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 quadratic 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_{nn} is the NN model, N_h is the prediction horizon, and the set $S(\Delta)$

comprises of piecewise constant functions having a period of Δ.

In this work, the outputs of the process to be regulated by the model predictive control system, x, are the production rates of CO, C_2H_4, and H_2. Specifically, we are aiming to get the productivity of CO and C_2H_4 to a certain setpoint while minimizing the productivity of the side product hydrogen from the competing hydrogen evolution reaction.

5.3   Neural Network Modeling

To account for the complexity of the electrochemical reaction mechanism and fill in the lack of a first-principles model, a neural network (NN) model is developed to capture the dynamic response of the output states under various input conditions. Subsequently, the trained NN model is utilized as the process model of the MPC to estimate the output states over a certain time horizon known as the prediction horizon N_h. This section describes the design and development of the NN model for this purpose.

5.3.1   Data Collection

The data set used to develop the NN model is similar to the one reported in Chapter 4, and three types of experiments (i.e., open-loop steady input, step changes, and closed-loop experiments) are performed to collect the data. Specifically, constant inputs (applied potential and catalyst rotation speed) are applied to generate some portion of the training set data, which provides information about the expectation of the steady state output values under certain input conditions in addition to the dynamic trends while reaching respective steady states. In the second type of

experiment, step-change inputs of random amplitudes are applied, but the input actions remain in a predefined range throughout the experiment. Finally, the closed-loop experimental results from Chapter 4 are included in the data set. Although the controller type used in Chapter 4 is different from the one in this work, the underlying physico-chemical phenomena are the same. Thus, including those results can help the model to capture the dynamic behavior of the system more efficiently.

The GC is used to monitor the outlet concentrations of the gas products in real-time during data collection. Specifically, the GC takes a gas sample injection and quantifies the production rates of the four gas-phase products every 1300 seconds during the experiment. Analyzing the injected gas sample takes 15 minutes, and the GC needs to cool down for 400 seconds before taking the next injection. Therefore, only four data points can be collected from a one-hour duration of the experiment. As a result, there are a total of about 200 GC measurements collected at the end of data generation experiments for the training, which is not enough to train a neural network model. To address this problem, a 3rd-order polynomial regression based on three consecutive GC measurements is applied to determine a probable output data trajectory between every GC measurement using the inputs measured every second. More details about this data enhancement process are reported in Chapter 4.

5.3.2 Long Short-term Memory Networks

Among many ML methods that can be used to capture nonlinear processes, the RNN family has been proven to be an effective modeling strategy for time-series forecasting tasks. Recently, RNN models have become popular in the research area of process modeling and control and have

been applied in many academic and industrial works [59, 208]. The long short-term memory network (LSTM) is one of the well-developed NN models that belongs to the RNN family. It shares the major design of architecture with other types of RNN models that have information flowing in two directions to capture the time-dependent relationship within the training data [143]. Furthermore, the LSTM model has its special "gates design" to store the historical information and determine how to use it to predict the output [65].

The architecture of the LSTM model used in this work is shown in Figure 5.2. The model is developed to predict the output state at the next consecutive sampling time using p historical state predictions and control actions. Therefore, there are only three outputs given by the model, which represent production rates of CO, C_2H_4, and H_2 (in ppm) at the $p + 1$ time step. Specifically, the LSTM layer maps the time-sequence input containing the historical state prediction and control actions to 180 hidden states. Subsequently, a dropout layer with a 30% dropout rate of the hidden states is inserted to prevent overfitting, and the remaining hidden states are densely connected to the output nodes.

Remark 21 *The number of hidden states and percentage rate of dropout are included in the hyper-parameters of the LSTM model. Therefore, their value can be found following the general hyper-parameter tuning process. Specifically, in this work, we performed a random search to locate those values. More precise methods to perform the hyperparameter tuning include cross-validation and grid search. More details about hyperparameter tuning can be found in [44].*

Remark 22 *In the area of machine learning, preventing the model from overfitting the data is an important task. Overfitting refers to the situation where the NN model can perform well with*

the training data set but fails to maintain good performance for data outside the training set. Several factors contribute to this problem, and a critical one is when the NN model has too many weight parameters, which can result in allowing the model to memorize the training data instead of extracting underlying trends from it. One method to reduce overfitting is the regularization [49], with the dropout method used in this work being one example of a regularization method [189, 169].
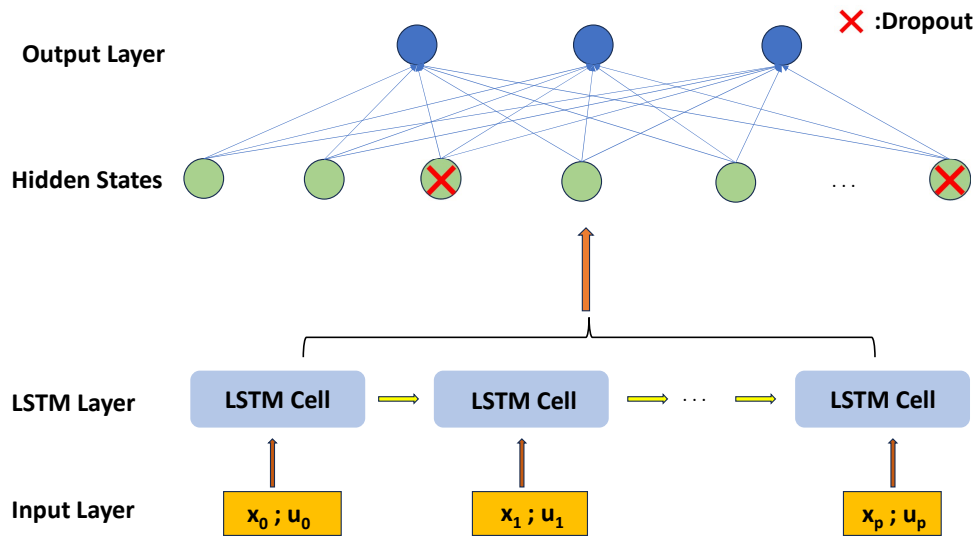
Figure 5.2: The architecture of the LSTM model used in this work that processes the input sequence with a LSTM layer and yields the prediction for the output states at the next time step (i.e., 100 secs later from the instantaneous point in time).

5.3.3   Model Training

Based on prior knowledge of the experimental reactor, the input sequence of the LSTM model is designed to contain one hour (3600 seconds) of historical information. However, if the data is

194

formatted on a per-second basis, each sequence in the data set will contain 3600 elements, which results in unnecessarily high computational costs in both time and space consumption. Therefore, the time step of the data sequence is designed to be on a per-hundred seconds basis to reduce the length of the input sequence to 36 elements. The time space in the input sequence is preserved in the output sequence, and since the output sequence only contains 1 time step, the overall function of the LSTM model is to use the one-hour historical information up to the instant to predict the output states at 100 seconds later in the future (i.e., the output sequence has a shape of (1,3), where 3 is the number of output species).

This sliding window algorithm was employed to create a training dataset from a collection of 35 experiments. Specifically, a window of one hour was used as the input for the training data, and the output of the LSTM was determined to be the production rates of the target species at 100 seconds after the final time step of the input sequence. The window was systematically slid by a stride of 100 seconds, and the first 1000 (seconds of) measurements in each experiment were skipped to enhance the reliability of the training data. Therefore, the input of the LSTM model has a shape of (36, 6), where 6 denotes the input features (i.e., surface potential, rotation speed, current, and previous states of production rates for C_2H_4, CO, H_2) measured at the respective time step. The sliding window algorithm is applied to 18 experimental data sets to generate the data sequence to develop the LSTM model. When developing an NN model for time-series forecasting problems, it is crucial to ensure that the validation data retains a certain level of independence from the training data to avoid potential information leakage. To address this concern, we randomly allocated results from 5 out of the total 18 experiments as the testing set, while the remaining experiments were assigned to the training set. This training set is further divided into two parts

195

before the model training for ratio validation purpose, using the train-test ratio of 70:30. Finally, the Scikit-learn Minmax Scaler was utilized to normalize the data.

In this study, the LSTM model was trained using the TensorFlow API. The model was optimized with the NADAM optimizer. As the data did not provide dense coverage of the overall operating conditions, it was crucial to maintain generalization and prevent overfitting. Therefore, we applied L2 regularization to the LSTM layer with a factor of 0.07 and performed 30% recurrent dropout within the LSTM cells. The mean squared error was selected to be the cost function to evaluate the model performance. The LSTM model underwent training for 45 epochs, with a batch size of 32. Additionally, a callback function was utilized to capture the best-performing weights based on minimization of the validation loss throughout the training process. As a result, the training and validation loss of the trained LSTM model were found to be 0.0028 and 0.00456, respectively.

Remark 23 *The length of the input sequence (i.e., 3600 seconds) was found based on the combination of experimental observations and hyperparameter tuning. Specifically, from the experiment, we found that the dead time of the process can vary up to 2000 secs, which meant, to capture the delay of the reactor, the length of the input sequence should be at least 2000 secs. Starting from there, we tuned the length of the input sequence and found that, with the length of 3600 secs, the LSTM model can capture well the dynamics of the process. Notably, increasing the length of the input sequence will result in higher computational cost, and since the length of the input sequence can be considered as a part of the hyperparameter tuning, the cross-validation method was used in this step.*

5.3.4 Model Performance

The trained model demonstrates significantly low training and validation losses, indicating its successful training. To further assess the model's performance, a comparison is made between the model's predictions, based on input data recorded from a validation experiment, and the corresponding output state measurements. Figure 5.3 is an example of such a comparison, with solid curves representing the predictions made by the LSTM model and dashed curves representing the measured trends during the experiment. The close alignment between the curves depicted in Figure 5.3 highlights the model's adequate prediction capabilities.
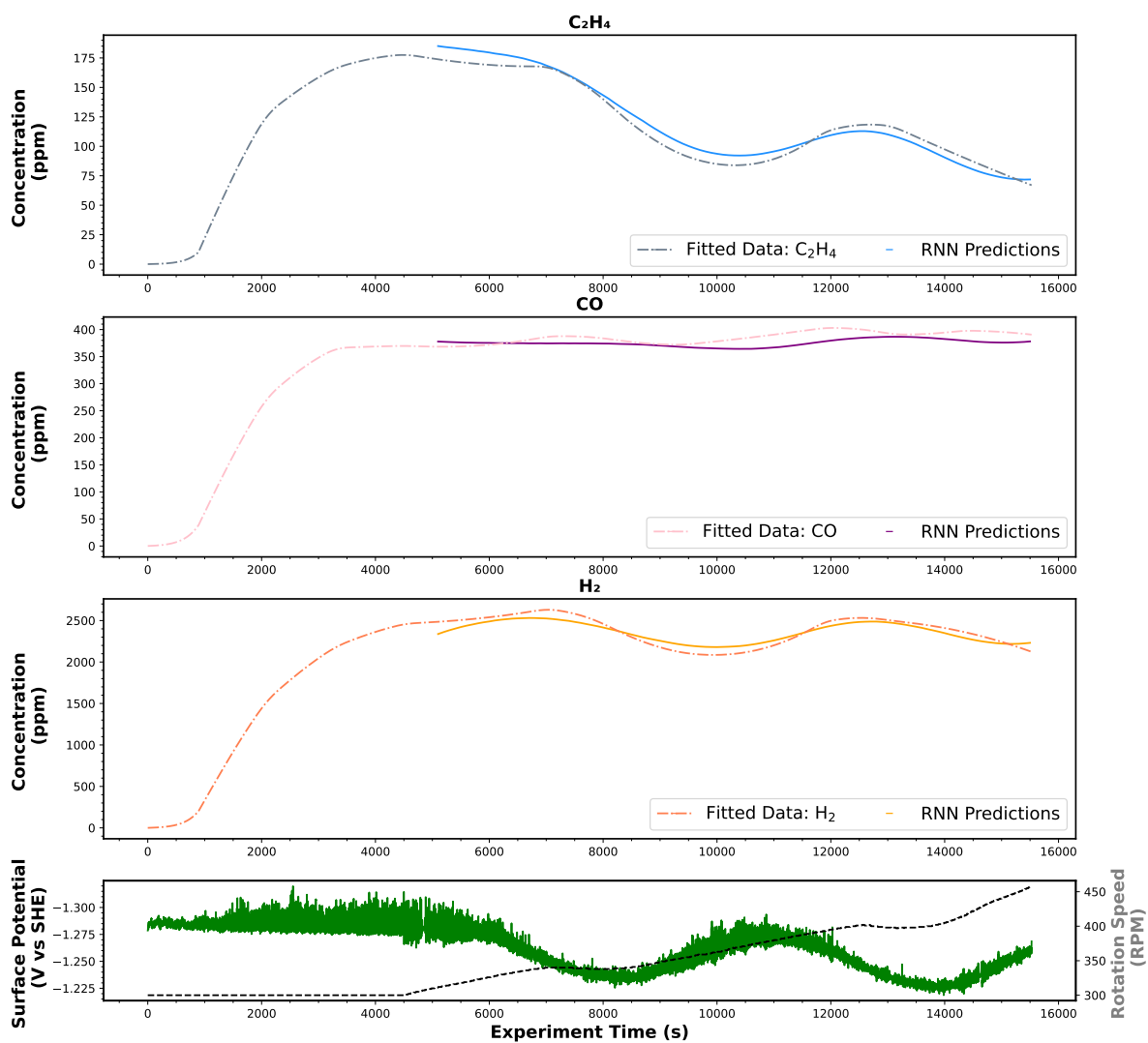
Figure 5.3: LSTM predictions of C_2H_4, CO, and H_2 concentrations compared to the reference data in the testing set. Inputs (surface potential and electrode rotation speed) used for the prediction are shown at the bottom.

Once the model demonstrated its proficiency in predicting the dynamic behavior of the output state, we proceeded to evaluate its ability to accurately capture the reactor's steady state performance. The electrochemical reactor is an inherently stable process in the operating region of interest, meaning that regardless of the initial output state conditions, the application of the same constant control actions throughout a period of time should lead to the convergence of the outlet species concentrations to the same steady state every time. The prediction results of the LSTM model for an open-loop experiment are shown in Figure 5.4. It can be observed that, regardless of the starting point of the trend, the predictions consistently converge to the same steady state for all three output states under a fixed control action. However, due to the stochastic nature of the electrochemical reaction and other experimental uncertainties, there exists a variance in the steady state. Therefore, the steady state given by the LSTM is ideally the average of the steady state values obtained if the experiment is repeated with the same fixed control inputs.
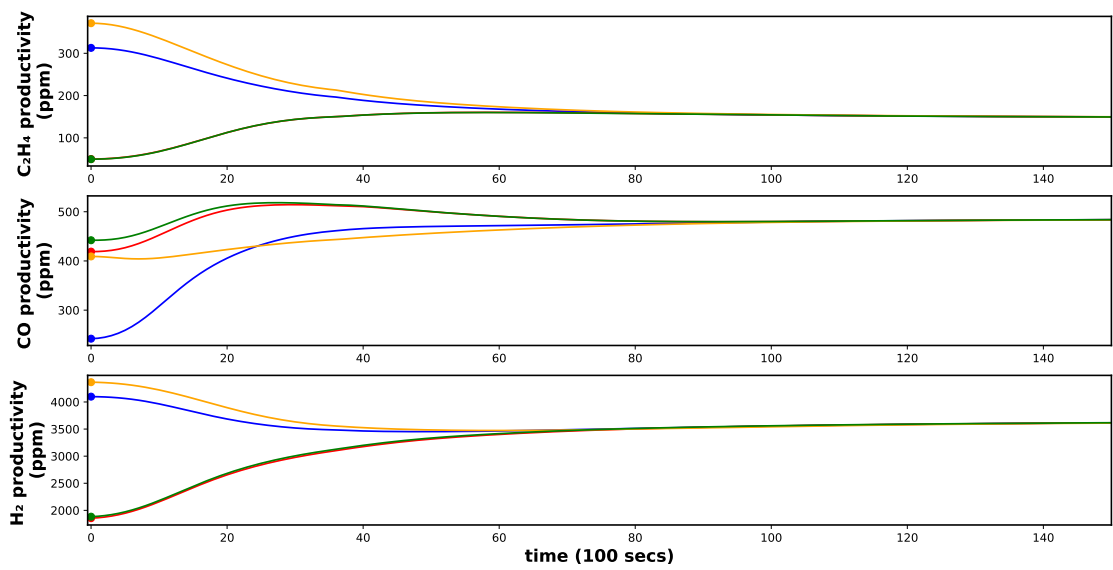
Figure 5.4: Open-loop simulation using the trained LSTM model with consistent fixed inputs from various initial states. The predicted trends for different initial states are represented in different colors.

5.4 Koopman Operator-based Linearization of RNN Model

The motivation behind the exploration of a method to linearize the neural network model for utilization in MPC arises from the fact that NN-based MPC involves solving a constrained optimization problem with a highly nonlinear NN model. Consequently, this optimization problem becomes a challenging nonlinear optimization task, which remains a topic of considerable mathematical exploration without a definitive approach for effective resolution. As a result, solving the nonlinear optimization problem within a reasonable time frame (certainly, within the process sampling time for real-time control purposes) might not be possible, which renders this type of nonlinear MPC application impractical for many industrial processes. On the other hand, the development of an MPC framework with a linearized system is a well-established approach. By approximating the NN model with a linear system on-line and at each sampling time, an MPC can be formulated as a quadratic programming problem, which lends itself to efficient solution techniques. This implies that if we can effectively approximate the NN model with a linear system, the NN model-based MPC can be solved quickly and efficiently, and applied to real-world applications. This section presents the systematic process utilized in this project, drawing inspiration from the work of [214], to linearize the RNN-based process model and integrate it into an MPC.

5.4.1 Koopman Operator Theory

[214] presented a method to linearize an RNN model based on the principles of the Koopman operator theory. The Koopman operator theory, initially proposed by Bernard Koopman in the 19th century [87, 88], plays an important role in analyzing, modeling, and controlling nonlinear

processes. The core concept of the Koopman operator theory involves mapping inputs of a nonlinear function into a higher-dimensional feature space, thereby obtaining a linear approximation of the nonlinear system [87]. In other words, the Koopman operator can linearize an arbitrary finite-dimensional nonlinear system at the cost of expanding its dimensionality up to infinity. Notably, this concept is also similar to the idea of feature engineering, in the machine learning terminology, which serves as a fundamental aspect in various ML models such as support vector machines (SVM) [27].

The Koopman operator can be defined mathematically with the following equations:

$$x_{k+1} = \mathbf{f}(x_k) \tag{5.3a}$$

$$\mathcal{K}g(x_k) \triangleq g(\mathbf{f}(x_k)) = g(x_{k+1}) \tag{5.3b}$$

where Eq. 5.3a is the discrete representation of a nonlinear dynamical system, and the function $\mathbf{f}$ captures the output state evolution of the system from an arbitrary time step k. Eq. 5.3b is the definition of the Koopman operator $\mathcal{K}$, where $g(\cdot)$ are a set of scalar functions named observables. From Eq. 5.3b, it can be easily proven that $\mathcal{K}$ is a linear operator, which allows finding the eigen decomposition of $\mathcal{K}$ and rewriting the evolution of the nonlinear system as a linear combination based on the eigen decomposition of $\mathcal{K}$ as follows:

$$\mathcal{K}\phi_j(x) = \lambda_j \phi_j(x), \quad j = 1, 2, \ldots, \infty \tag{5.4a}$$

$$\mathcal{K}g(x_k) = \sum_{j=1}^{\infty} \lambda_j \phi_j(x_k) \mathbf{v}_j \tag{5.4b}$$

where λ_j, ϕ_j, $\mathbf{v}_j$ are known as the eigenvalues, the eigenfunctions, and the mode of the Koopman operator $\mathcal{K}$.

The discussion about Koopman operator theory so far has been centered around an autonomous system with time-varying inputs. However, to allow using this method in a dynamic control system requires extending the Koopman theory to be able to handle a non-autonomous system including time-varying control inputs. In [134], a generative Koopman with inputs and control (KIC) method was proposed to generalize the application of Koopman operator theory to non-autonomous systems. Specifically, the KIC method defined a new representation of the Koopman operator as follows:

$$x_{k+1} = \mathbf{f}(x_k, u_k) \tag{5.5a}$$

$$\mathcal{K}g(x_k, u_k) \triangleq g(\mathbf{f}(x_k, u_k), u_{k+1}) = g(x_{k+1}, u_{k+1}) \tag{5.5b}$$

where u_k is the input applied at the kth time step, and Eq. 5.5a is the discrete representation of any nonlinear system accepting external inputs. There are other works proposing different formulations for Koopman operator with inputs [e.g., 90], and the core ideas shared around those methods involve augmenting the states x and the inputs u into the same matrix and use it to form the observables instead of just the states, which all allow linearizing the nonlinear system using the method applied to an autonomous system.

Remark 24 *The method of constructing the observables g is an essential research area of Koopman operator theory, and there are significant efforts on this subject, such as using a nonlinear function to augment the state measurements [151, 198, 17, 134]. In this work, we define the observables to be the output states of the system, such that $g(x) = x$.*

Remark 25 *The Koopman operator can also be applied to a dynamic system with continuous representation. However, the focus of our mathematical analysis and investigation in this work is centered on the discrete representation, as the LSTM model can be considered as a discrete approximation of the underlying nonlinear dynamic system. Therefore, applying the Koopman operator to a discrete nonlinear dynamic system fits better to the application of the Koopman operator to the LSTM model.*

5.4.2 Dynamic Mode Decomposition

Although the Koopman operator method suggests linearizing a nonlinear system into an infinite-dimensional linear system, it is practical to work with a finite dimension that is high enough to achieve the desired accuracy. Considering this, the Dynamic Mode Decomposition (DMD) method, first proposed in [157], is an effective method to provide a finite-dimensional approximation of the Koopman operator. Specifically, the DMD method is a data-driven method that requires obtaining measurements to start with. We define $O_k = g(x_k)$ to be the observation of a nonlinear system and $O_k^+ = O_{k+1}$ to be the observation one time-step after O_k. By performing experiments or simulations with the nonlinear system, time-sequence data can be collected for the observations and yield:

$$\mathbf{O} = [O_0, O_1, \ldots, O_{n_s}], \quad \mathbf{O}^+ = [O_0^+, O_1^+, \ldots, O_{n_s}^+] \tag{5.6}$$

where n_s is the total number of samples. Notably, the notation O_1 is not necessarily the next time step of O_0. Subsequently, the DMD of the nonlinear system based on the measurements can be

found as the eigen decomposition of the linear mapping matrix $\mathbf{A}$ that forms the equation,

$$\mathbf{O}^+ = \mathbf{A}\mathbf{O} \tag{5.7}$$

The analytical solution of Eq. 5.7 yields the matrix $\mathbf{A}$ as $\mathbf{A} = \mathbf{O}^+\mathbf{O}^\dagger$. Finally, the eigenvalues and eigenvectors of $\mathbf{A}$ are the approximation of the eigenvalues and the mode of the Koopman operator, respectively.

For a nonlinear system with inputs, the Dynamic Mode Decomposition with control (DMDc) method was proposed in [133], which includes the measurements of the control actions $\mathbf{O}_u = [u_0, u_1, \ldots, u_{n_s}]$ to compute the linear mapping matrix $\mathbf{G} = \mathbf{O}^+ \begin{bmatrix} \mathbf{O} \\ \mathbf{O}_u \end{bmatrix}^\dagger$ defined for the DMDc method. Similarly, the singular value decomposition of $\mathbf{G}$ can provide a finite approximation of KIC. Eventually, an arbitrary non-autonomous nonlinear system defined as Eq. 5.5a can be linearized with the DMDc method into the following system:

$$\mathbf{G} = [\mathbf{A}\ \mathbf{B}] \tag{5.8a}$$

$$x_{k+1} = \mathbf{A}x_k + \mathbf{B}u_k \tag{5.8b}$$

$$y_k = \mathbf{C}x_k + \mathbf{D}u_k \tag{5.8c}$$

Furthermore, the process of computing the matrix $\mathbf{G}$ involves solving a linear least-squares problem, which can be solved more effectively in practice with the regression method rather than finding the analytical solution [90]. Therefore, the extended dynamic mode decomposition (EDMD) method proposed in [198] introduced a regression procedure to approximate the Koopman opera-

tor.

5.4.3   Linearization of LSTM model and Performance Evaluation

The Koopman operator theory and EDMD method are utilized to linearize the LSTM model because they are data-driven and independent of the form of the nonlinear model [5, 214]. The first step of implementing these methods is to collect time sequence trajectories of the LSTM model. Following the procedure of [134, 214], at the kth time step, we first define the vectors $y = [\hat{x}_{k+1}, \hat{x}_{k+2}, \ldots, \hat{x}_{k+N_t}]^\top$, $\hat{x} = [\hat{x}_k, \hat{x}_{k+1}, \ldots, \hat{x}_{k+N_t-1}]^\top$, and $u = [u_k, u_{k+1}, \ldots, u_{k+N_t-1}]^\top$, where N_t is the distance between the farthest time step contained in the linearization samples and the kth time step. Notably, the historical information that is used by the LSTM model to make predictions up to the kth time step is available at the time t_k. Thus, the prediction $\hat{x}_{k+1}$ can be computed using the LSTM model. Furthermore, by adding the new prediction and the next control action u_{k+1} while removing the first element of the LSTM input, the vector y can be obtained by iteratively running the LSTM model.

In this work, we applied a constraint on how much the input actions can be changed from one sampling time to the next, which is mathematically defined by the following equations:

$$u_k = [v_k, r_k, c_k] \tag{5.9a}$$

$$c_k = C(v_k, r_k) \tag{5.9b}$$

$$u_d \triangleq [v_{k+1} - v_k, r_{k+1} - r_k] \tag{5.9c}$$

$$|u_d| \leq u_c = [v_b, r_b] \tag{5.9d}$$

where v_i, r_i, c_i are the surface potential, rotation speed, and the current value given by the reactor at the ith sampling time. The surface potential and rotation speed are the control actions that can be manipulated during the experiment, and the current varies as a consequence of these control actions. v_b and r_b are positive numbers referring to the maximum absolute step changes allowed per time step for the potential and rotation speed and are equal to 0.01 V and 30 RPM, respectively.

The data to linearize the LSTM model is generated with respect to the constraints of Eq. 5.9. Specifically, we first determined the number (N_s) and the length (N_t) of the time-sequence data. Then, we randomly generate N_t control actions starting from the same initial control action u_0 that obey the constraints of Eq. 5.9 and run the LSTM model to generate one time sequence of "measurements" of y. This process is repeated N_s times to obtain N_s sequences. [180] pointed out that, due to the reduction of the problem into a linear regression formulation, the data set used to perform the DMD-based method does not need to retain the sequential order of the data points (i.e., the rows of the data matrices can be shuffled such that, for example, the last row of the target vector y, $\hat{x}_{k+N_t}$ can instead be moved to be the first row, as long as the last rows of $\hat{x}$ and u, $\hat{x}_{k+N_t-1}$ and u_{k+N_t-1} are also moved to be their first rows, respectively). Therefore, the data matrices y, x, and u with a shape of ($N_s \times N_t$) can be reshaped into three vectors containing ($N_s \times N_t$) elements (also note $n_s = N_s \times N_t$), as long as the triplets of x_i, u_i, and y_i remains the same. With this data structure, the linear least-square regression problem to find $\mathbf{G}$ can be easily solved by using the Scikit-learn linear regression function without fitting the intercept. The pseudocode to implement the linearization of LSTM in our work is represented in algorithm 1.

The prediction given by the linearized model was compared with the original LSTM prediction. Specifically, the initial inputs to activate the LSTM model are randomly cropped from existing

Algorithm 1 Procedure of linearizing the LSTM model.

Input: $v_b \geq 0, r_b \geq 0, NN$ $\triangleright NN$: the initial inputs of LSTM with shape (1, 36, 6)
Input: x_{train} , y_{train} $\triangleright$ Define empty arrays to store linearization samples
  1: **for** $i \leftarrow 1$ **to** N_s **do**
  2:      $NN_i \leftarrow NN$ $\triangleright$ Initialize the LSTM input by making a deepcopy of NN
  3:      **for** $t \leftarrow 1$ **to** N_t **do**
  4:          $v_i \leftarrow NN_i[0, -2, 0] + rand(-v_b, v_b)$ $\triangleright rand(l, h)$: randomly pick number between l
     and h
  5:          $r_i \leftarrow NN_i[0, -2, 1] + rand(-r_b, r_b)$
  6:          $c_i \leftarrow C(v_i, r_i)$ $\triangleright C(\cdot)$: eq. 5.9b
  7:          $NN_i[0, -1, : 3] = [v_i, r_i, c_i]$
  8:          $\hat{x}_{i,t} = LSTM(NN_i)$ $\triangleright LSTM(\cdot)$: LSTM prediction with shape (3)
  9:          $x_{\text{train}}.append(NN_i[0, -1, :])$
10:          $y_{\text{train}}.append(\hat{x}_{i,t})$
11:          $NN_i[0, : -1, :] \leftarrow NN_i[0, 1 :, :]$
12:          $NN_i[0, -1, 3 :] \leftarrow \hat{x}_{i,t}$
13:      **end for**
14: **end for**
15: $[A, B] \leftarrow LG(x_{\text{train}}, y_{\text{train}})$ $\triangleright LG(\cdot)$: Sickit-learn linear regression

---

experimental results and provided to algorithm 1 to generate a linearized model. Subsequently, the linearized model was utilized to make predictions over a time span based on a sequence of control actions randomly picked within the step change constraint and compared to the prediction given by the original LSTM model using the same control sequence. The comparison is shown in Figure 5.5, where the prediction given by the original LSTM model over a time span of 800 secs is represented in the blue solid curve, while the prediction given by the linearized model is denoted in the red dashed curve. The predictions given by the two models are close to each other, which supports that the linearized model can approximate the LSTM model adequately.
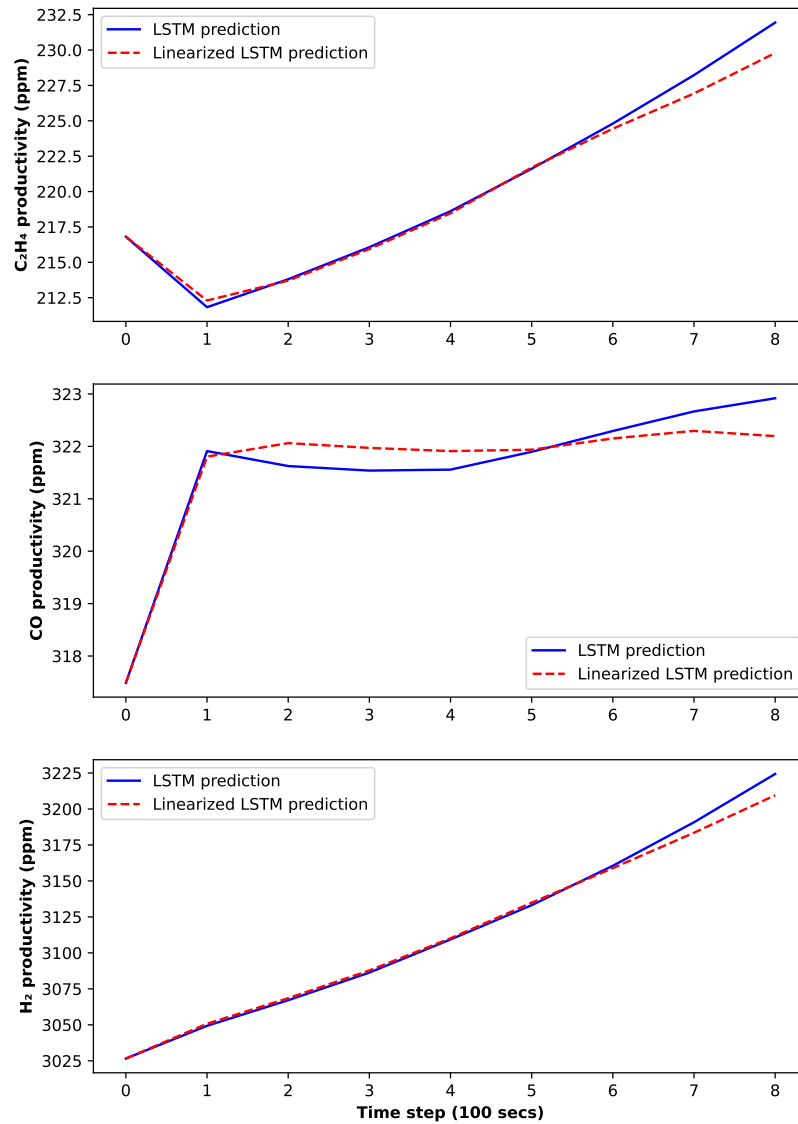
Figure 5.5: Comparison between the linearized model prediction (dashed curve) and the original LSTM model prediction (solid curve) over a sampling period.

Remark 26 *Notably, for this work, the current flowing at a fixed applied potential depends on the electrolyte solution resistance due to the Ohmic loss. The solution resistance between the working and the reference electrode is determined from the electrochemical impedance spectroscopy (EIS) and is around 6.5 ± 0.3 Ω in the RCE cell setup when using 0.2 M potassium bicarbonate electrolyte. Although the value is practically constant, there are slight variances from experiment to experiment and during the experiment, while it is measured only before and after the experiments, as discussed in Chapter 4. Therefore, the measured current value will not be the same with the same control action, and thus, provide additional information for our LSTM model to learn the electrochemical reactor system better. The current value is measured and recorded during the experiment, and those measurements are used to train the LSTM model. However, when collecting samples for the Koopman-based linearization of the LSTM model, the value of the current needs to be approximated with the correlation between c_k and the control actions denoted as transformation C in Eq. 5.9b. In simulations, this value was approximated using the average resistance obtained from various experiments. For the closed-loop experiments, the resistance value was measured right before starting the experiment and used to anticipate the current value in the prediction horizon.*

5.5   Closed-Loop Experiments

The details of implementing the linearized NN-based MPC for the electrochemical reactor are presented in this section. As a quick recap, referring to Eq. 5.2, the main objective of the MPC in this work is to drive the productivity of C_2H_4 and CO to their specific setpoints while

suppressing the productivity of H_2. The setpoints for C_2H_4 and CO are selected to be 147 ppm and 478 ppm, respectively, such that $x_r = [147, 478, 0]$. Furthermore, by replacing the LSTM model used in Eq. 5.2d with the linearized model, the overall optimization problem within the MPC becomes a quadratic programming problem, which is convex and can be solved efficiently. In the closed-loop experiment, the MPC is operated in a sample-and-hold manner, which means it will give the optimum control action over a certain control period (i.e., 100 seconds in this work), and the control action will be held fixed during the control period. The overall workflow of the MPC is demonstrated in Figure 5.6. Specifically, the LSTM model worked as the state estimator throughout the experiment. When entering a new sampling time, algorithm 1 was used to compute the linearization of the LSTM model for the specific time-instant, which was then used to find the MPC control action by solving a QP problem. The Gurobi optimizer was used to solve the optimization problem in this work.
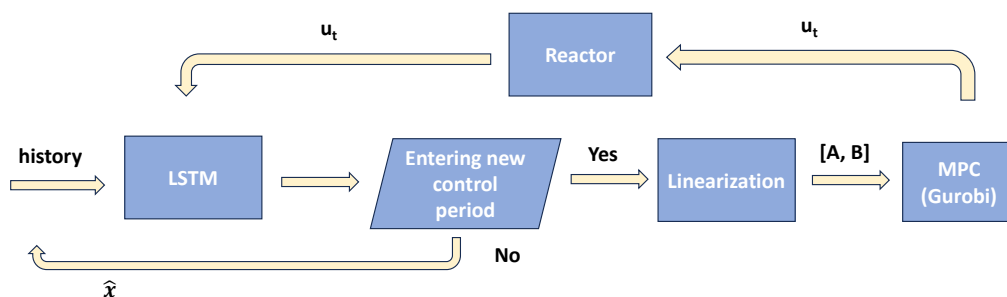
Figure 5.6: The overall workflow of the MPC in this work. The LSTM model is used as a state estimator when the MPC is not activated. Once entering a new sampling time, the MPC is activated and computes the control action for the reactor with the linearization of the LSTM model.

5.5.1 Implementation of the MPC in the Experimental Setup

The LSTM model (without linearization) worked as the state estimator in the closed-loop experiments by predicting the instantaneous reactor productivity. When the processing of a new GC measurement is finished, the LSTM model has to reinitialize its prediction, which means that it uses the state measurements in the input of LSTM instead of the state prediction given by the LSTM in the previous iteration. This reinitialization is expected to prevent the accumulation of prediction errors. Specifically, the output states are estimated through 3rd-order polynomials based on the 3 consecutive GC measurements up to the newest measurement and used as the input of the LSTM model to predict the output state at 100 seconds after the newest GC injection made. Note that the GC measurement has a delay of 15 minutes because it takes 15 minutes to separate and analyze the sample taken from the injection. Therefore, through the reinitialization, the LSTM predicted the output state 15 minutes ago again, and needs to run iteratively using the reinitialized prediction to correct all the predictions for the previous 15 minutes. Furthermore, since this 3rd-order polynomial's approximation can only be activated once every 21 minutes, it can not be used as the process model or state estimator that requires to be able to give prediction every 100 seconds. But once the 3rd-order polynomial approximation is activated, it can estimate the output states for the last 1 hr effectively and accurately.

The LabVIEW software was utilized to digitalize the electrochemical reactor in this work. LabVIEW is a graphical programming language that allows a user to develop a user interface to monitor the system and develop control systems to implement the control actions in the working equipment. Although LabVIEW also allows users to develop simple programs (e.g., PI controller),

it is technically challenging to implement the aforementioned workflow that involves using the NN model, linearization, and optimization in LabVIEW. Therefore, a data pipeline was developed to allow information to flow between a Python script operating the workflow and the LabVIEW controlling the operating equipment.

The options for pipeline design include reading the data from a real-time updated csv file or data transfer through a database. Since opening a real-time updated csv file to read data might disturb the process of writing data, this option is not optimal. On the other hand, sending data from LabVIEW to a database is an easy task that is already combined into our automation scheme using the Clean Energy Smart Manufacturing Innovation Institute's (CESMII) Smart Manufacturing Innovation Platform (SMIP) as discussed in Chapter 3. Specifically, the SMIP can work as a database to store and organize our data at defined endpoints for each piece of equipment, and the use of start and end dates for query and mutation of SMIP's data transfer protocol, GraphQL, makes it a perfect candidate for data transfer application. In short, this data flow is designed to use LabVIEW as the edge device performing process control and monitor tasks, SMIP as a cloud database for data management, and a high-performance computer running Python interpreters as a back-end server. The data transfer protocol is shown in Figure 5.7.

When performing an experiment, constant physical properties, such as solution resistance, open circuit potential, etc., are measured before the electrolysis and sent to the SMIP at the beginning of the experiment. Process data collected through LabVIEW, such as applied potential, surface potential, rotation speed, and current, are mutated to the SMIP every 2 seconds. The reactor was put to run in open-loop for the first 7000 seconds of the experiment because, at the beginning state of the experiment, the rector does not reach the equilibrium giving higher variance

213
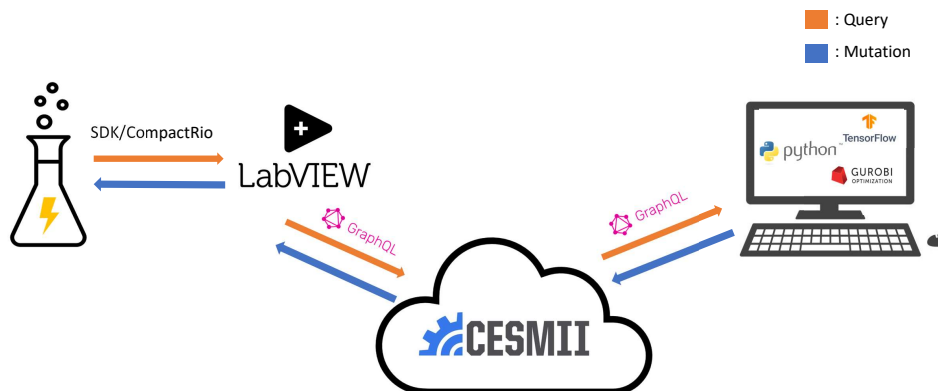
Figure 5.7: Data flow between the experimental setup and local Python script through SMIP for MPC calculations.

in its productivity. Thus, data collected at this stage is expected to have a different distribution from the rest of the experiment and is excluded from the LSTM training. Control actions applied to the reactor at this stage are fixed to be $-1.22\ V$ for potential and 300 RPM for rotation speed.

After letting the reactor run in open-loop for the first 6298 seconds, the MPC will be activated, and the Python script queries the last one hour of process data every 100 seconds to form the initial input for the LSTM model. Subsequently, the LSTM was linearized to compute the first control action, while the original LSTM model estimations along with input values are mutated to SMIP. LabVIEW script also queries those values from the SMIP to feed the new input values to the potentiostat and modulated speed rotator to implement the new applied potential and rotation speed. From then on, the experiment was run in closed-loop.

5.5.2 Closed-loop Experiments

The result of the closed-loop experiment is summarized in Figure 5.8, where the dots are the GC measurements collected in the experiment and the dashed curves are the approximated output states with probable experimental trajectory method employing 3rd-order polynomials. The productivity evolution of C_2H_4 and CO_2 is shown in the top figure, and their setpoints, which are 147 and 478 ppm, respectively, are denoted with the dotted straight lines. A reference control action $u_r = [-1.28\ V,\ 600\ \text{RPM}]$, stated in Eq. 5.2c, was used in the MPC objective function to achieve better control performance. The reference control action u_r was found using the trained LSTM model. As discussed above, the LSTM model is stable, such that it can be used to find the theoretical control actions that can give the targeted steady state. The weight matrix was chosen to be $Q = \text{diag}(0.01, 0.01, 1 \times 10^{-6})$ and $R = \text{diag}(1 \times 10^4, 1.0/1200)$. The weight parameters were tuned based on the simulation and experiment results.

The design of the weight matrices considered scaling their importance on the cost function. For example, when designing the matrix Q, the first two parameters are the weight of C_2H_4 and CO_2, respectively, which are equally important in our control scheme. The value 0.01 in the Q matrix was used to prevent the cost value from becoming too big. On the other hand, the cost for the H_2 is much less than the other two weights because driving the outputs to the setpoint is the first priority for the control system. Reducing the productivity of the side product H_2 can maximize the energy efficiency of our reactor. However, it is physically impossible to eliminate the H_2 production, which means if the weight parameter for the H_2 is too high, the MPC will allow the two target states to be away from the setpoint as the trade-off to reach the optimum defined by the

MPC objective function. Drawn from the understanding of the reactor, we considered the reactor operated energy efficiently if the productivity of H_2 was kept below 4000 ppm. Since the states are squared in the objective function, using 1×10^{-6} will give a cost equal to 16 if the productivity of H_2 is at 4000 ppm, which will dominate the MPC with the outputs approaching the setpoints, and thus, the productivity of H_2 was not included in the objective function. The weight matrix R was calculated to balance the speed of the convergence of the states to the steady state and the magnitude of the control actions.

The prediction horizon of the MPC (N_h) is 8 times steps (i.e., 800 seconds in the future), and the length of the time sequence, N_t, collected for linearization is designed to be equal to N_h+2. The number of linearization samples, N_s, is taken to be 30. Theoretically, linear regression can be more accurate with increasing amounts of data. On the other hand, increasing N_t and N_s also requires more time to collect the sample, which makes the linearization more computationally expensive. Notably, the sampling step can be processed in parallel, which means the computational time is independent of the size of N_s if there are sufficient amounts of parallel processors. However, since the processing of the time sequence is iterative, the computation time for linearization is bounded by the size of N_t. In this work, the maximum allowable step changes in u reduces the required number of sequences in our linearization sample. In our implementation, 30 sequences collected with the Monte Carlo method turned out to be sufficient for the linearization task. With this choice, the MPC successfully drives the outputs to the setpoint while maintaining the H_2 production rate below 4000 ppm.
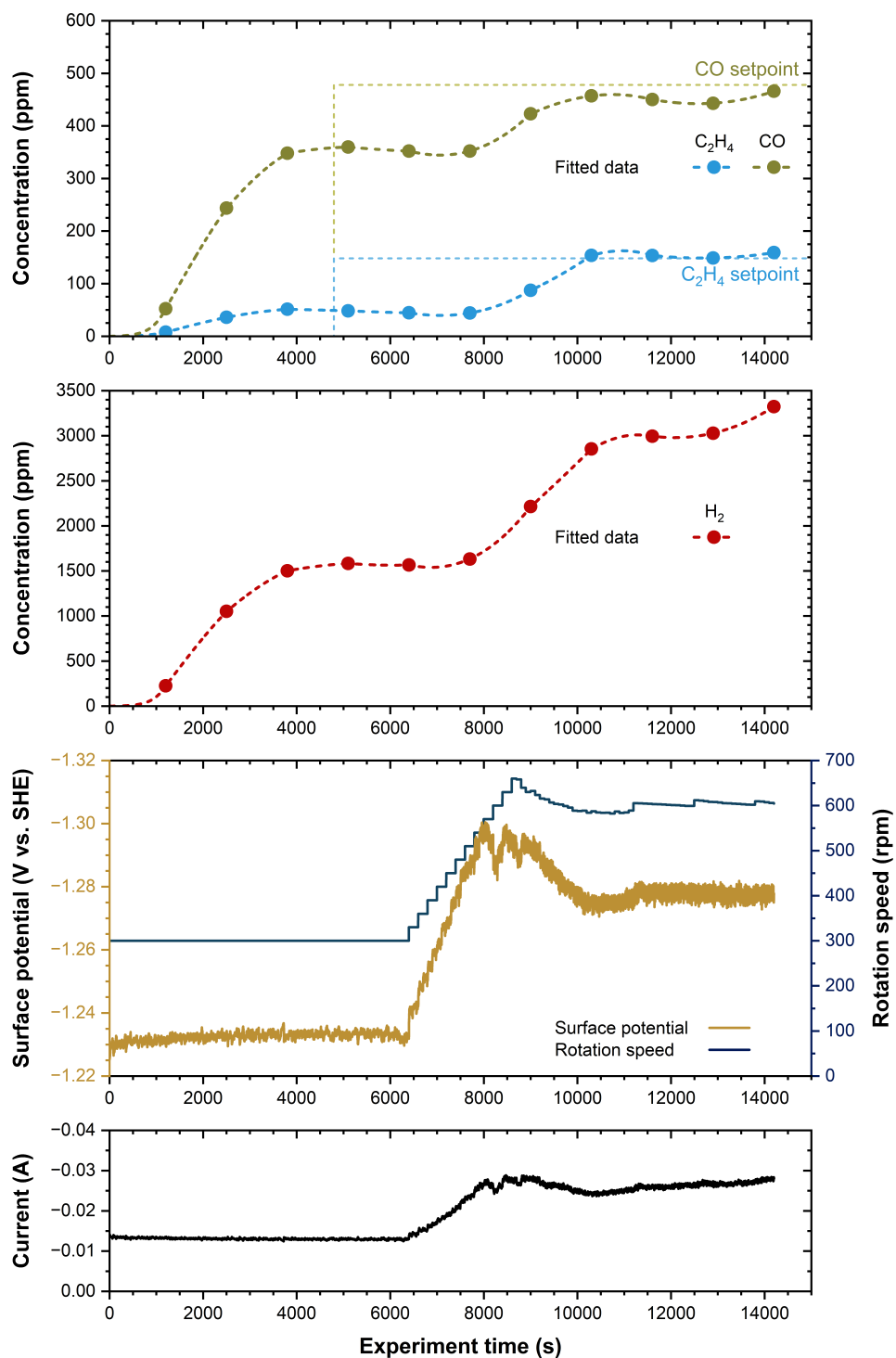
Figure 5.8: Output responses and control actions in the closed-loop experiment controlled by the MPC using the linearization of the LSTM model.

5.5.3 Model Retrain

The MPC design in this work created a feedback loop by using real-time measurements to re-initiate the LSTM model with measurement feedback and improve closed-loop system robustness. However, a correction algorithm that uses the feedback information to improve the LSTM model was not implemented, which means the control scheme demonstrated in this work is based on the assumption that the LSTM model can capture the real process accurately and that the process behavior does not change significantly. However, in real-world applications, the system is very likely to perform differently from the model prediction due to the variance of the application. The reported control scheme in this work should be able to handle this slight variance, as long as the variance is not significant enough to have the steady state shifting on a very different condition. However, sometimes the process may have very different behavior than the data collected to develop the neural network model. This problem is usually called the data (process) shift problem (for example, due to catalyst activity variation as a new catalyst is introduced every certain number of experiments), and more actions need to be taken to account for the data shift problem in model update.

To this end, we introduced a model retrain procedure based on the transfer learning concept to update the process model efficiently when the data shift problem is detected. To imitate this data shift problem in our reactor, we changed the polycrystalline Cu RCE to a new one and followed the same procedure to synthesize nanopores, but the resulting performance was different. Specifically, the new catalyst was more active and had a selectivity toward the C_2H_4 production. Various experiments were conducted with the new catalyst, which gave the result that with the control action at

-1.28 V and 600 RPM, the output for C_2H_4 increased to about 200 ppm but remained unchanged for CO.

Subsequently, the LSTM model was retrained based on the data collected from the new experiments. Of course, the new data set is smaller in size compared to the original data set. Therefore, if we just train a new model after including the new data into the original data set, the newly trained LSTM is very likely to count heavily on the old data set and represent less of the performance of the new catalyst. To account for this, we used the idea of transfer learning, which is a scheme to fine-tune a pre-trained model to make it fit better to a new data set. Since the underlying physico-chemical phenomena do not change with the catalyst change, the available LSTM is a good pre-trained model that captures the critical dynamic relations from the previous training.

Specifically, the training of all layers in the pre-trained LSTM model except the output layer was frozen with the assumption that the ground truth physical relationship is captured in the LSTM layer. Subsequently, the model is trained with only the new experimental results. The model is trained with 10 epochs because it is common in transfer learning to train the model with a small number of epochs to prevent it from overfitting the new data, especially when the size of the new data set is small. Eventually, the retrained model preserved a stable behavior and predicted the new u_r to be -1.26 V and 650 RPM. This result matches our expectation for the new catalyst since the C_2H_4 productivity is more correlated to the applied potential while CO productivity is more correlated to the rotation speed. Based on the experiment observation, the new u_r should decrease the potential to reduce the productivity of C_2H_4 to better approach the setpoint. However, reducing the potential will also reduce the productivity of CO even if it is more correlated to rotation speed. The rotation speed then needs to be increased slightly to compensate for the loss in CO productivity.

Thus, we concluded that the transfer learning-based retrain process calibrates the LSTM model in the correct direction and moved on to using it to perform closed-loop MPC experiments of controlling the reactor with the new catalyst. The result of the closed-loop experiment is shown in Figure 5.9 which demonstrates that the MPC with the retrained model can stabilize the reactor outputs to the desired setpoints.
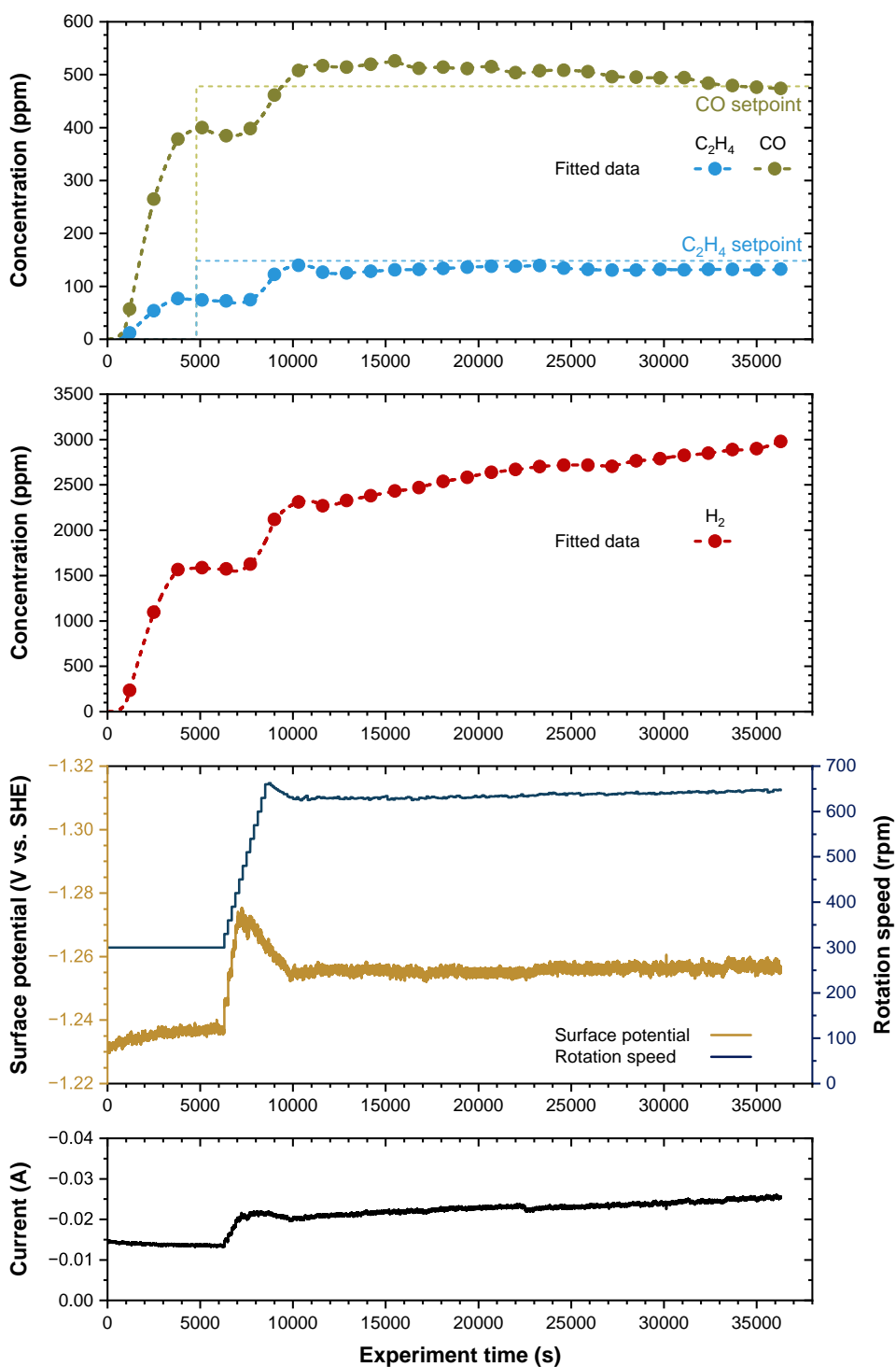
Figure 5.9: Output responses and control actions with new catalyst controlled by the MPC using the retrained model.

Remark 27 *In addition to the retraining method that corrects the model off-line, on-line correction may be implemented to improve the MPC performance using real-time measurements. For example, the extended Kalman filter method can be a good candidate to be considered for this task. To implement this method, the Kalman correction factor should be added to the LSTM model for real-time estimation. Furthermore, since the Koopman method can be applied to any nonlinear model, the overall workflow of the model linearization and MPC implementation does not need to be changed to include the Kalman filter correction. Developing this correction step for an MPC of an electrochemical reactor is one of our future objectives.*

Remark 28 *The retraining correction requires collecting new data from the process, which may introduce a certain delay to update the MPC. Consider the case where the data shift problem is just detected, and the collected data is not enough to retrain the process model, the MPC should be deactivated and switched to a backup controller (e.g., classical proportional integral controller) to ensure that the process operates safely.*

Chapter 6

Model Predictive Control of an Electrically-Heated Steam Methane Reformer

6.1 Introduction

Hydrogen (H_2) plays a fundamental role in the decarbonization and electrification of various industrial applications as a versatile and clean energy carrier [137, 174]. It serves as a building block for the synthesis of ammonia (NH_3), methanol (CH_3OH), fertilizers and petrochemicals [52]. Industries are increasingly using H_2 as a fuel source for power generation, reducing greenhouse gas emissions, and promoting environmentally friendly practices. Hydrogen-powered vehicles, such as fuel cell electric vehicles, are emerging as an alternative to the battery-based electric vehicles in the transportation sector [173]. Industries like steel manufacturing utilize hydrogen in the reduction of iron ore, a process that offers a cleaner alternative to conventional methods, decreasing carbon

emissions. This makes hydrogen an essential element in the pursuit of more sustainable metal production [103]. Furthermore, the refining industry employs hydrogen extensively for desulfurization and hydrocracking processes [23]. These applications enhance the quality of fossil fuels and ensure compliance with environmental standards, which showcases the significance of H_2 in refining operations.

The industrial sector employs various methods to produce hydrogen, each tailored to specific needs and environmental considerations. Water electrolysis, is a clean and increasingly popular but energy intensive method that uses electricity to split water into hydrogen and oxygen. The rise of green hydrogen emphasizes the value of electrolysis powered by renewable energy sources, which also minimizes carbon emissions [183]. Thermochemical water splitting is also an emerging alternative that leverages high temperatures and chemical processes to release hydrogen from water or hydrogen-rich compounds [152]. The most widespread method for the production of hydrogen, however, is steam reforming, where a hydrocarbon or coal undergoes a chemical reaction with steam to generate hydrogen and carbon dioxide as schematically shown in Figure 6.1(a) [119]. Natural gas steam reforming gives the highest yield of H_2 among hydrocarbons and coal. Methane is the molecule with the highest H:C molar ratio and the highest theoretical H_2 yield. The yields achieved in industrial hydrogen production are quite different from the maximum possible theoretical yields (Figure 6.1(b)). This, however, is not due to a low degree of conversion but it is due to the fact that the large amount of energy needed to produce hydrogen is usually supplied by the raw material itself. Burning of natural gas to provide heat to the strongly endothermic steam reforming reaction, generates excess CO_2.

Among the different methods for industrial H_2 production, the most common is steam methane

reforming (SMR), accounting for 48% of the total current hydrogen production [181]. Projections indicate that SMR will still be the most widespread H_2 production technique in 2050 [181], particularly due to natural gas availability and the challenges with scaling up electrolyzer manufacturing to global scales [71]. As industries prioritize sustainability, advancements in these production methods, coupled with innovations, are central in shaping a greener and more efficient future for hydrogen production within the industrial sector.

A sustainable enhancement to steam methane reforming involves substituting fossil fuel combustion with electricity to supply the necessary heat for the chemical reactions [200]. The use of renewable, carbon-free electricity to generate heat in an electric resistance-heated reformer promises to i) displace natural gas combustion as a source of heat, ii) deliver more compact, economic and efficient units for competitive H_2 manufacture, and iii) increase H_2 yields per ton of natural gas while reducing its CO_2 emission intensity (Figure 6.1(b)) [128].

In an electrically heated SMR process, two electrodes are connected to the top and bottom of the tubular reactor with a washcoated catalyst, and an electric current is applied, resulting in a flow of electrons to generate heat. This flow of electrons results in a more uniform distribution of temperature in the radial direction, as shown in Figure 6.1(a). In addition to electrically heated SMR, a novel method is introduced in [109] that leverages electric current for membrane separation of H_2. This process occurs within a proton-conducting membrane reactor, wherein the generated H_2 is extracted through a flow of electrons across a proton-permeable membrane. The real-time removal of H_2 changes the reaction thermodynamics to shift the equilibrium reactions towards more production of H_2 [109]. This increases the hydrocarbon conversion to 99%, which is around 74-85% using conventional methods [119]. At UCLA, we have constructed an experimental setup

for both electrically heated SMR and proton-conducting membrane reforming processes to further examine the reaction dynamics, and to develop modeling and control strategies while comparing these two technologies directly.
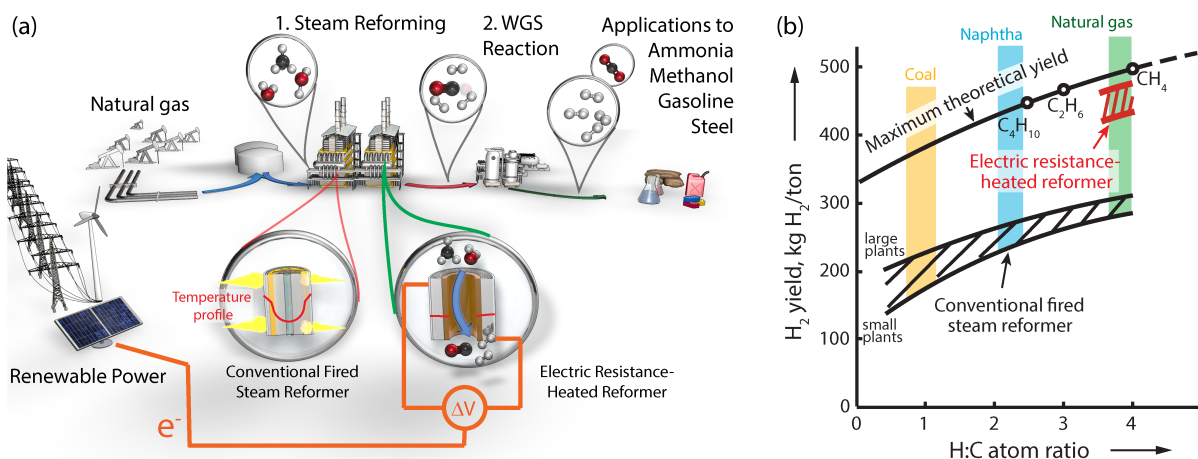
Figure 6.1: (a) Schematic comparison of conventional fired steam methane reforming and electrically heated steam methane reforming. (b) Comparison of hydrogen production from hydrocarbons and coal for conventional fired and electric resistance-heated steam reformers.

Model predictive control (MPC) plays a significant role in enhancing the efficiency and operability of industrial chemical processes, and could play a significant role in enhancing the efficiency and operability of steam methane reforming processes, particularly those connected to future electricity grids with high penetration of renewables. In SMR, where complex chemical reactions and heat transfer dynamics are tightly coupled, MPC will serve as a powerful tool for optimizing system performance. By using real-time measurements and predictive models, MPC can dynamically adjust operating parameters such as temperature, pressure, and inlet flow rates, to ensure optimal hydrogen production and minimize energy consumption. Although a great deal of work has been carried in the modeling and control of traditional fired SMR systems, significant knowledge gaps remain in the electrification of SMR reactors. For example, a computational study was

conducted by [204] using a robust CFD model for proposing an MPC for a traditional fired SMR process, and the results were compared to a PI controller. In [219], a computationally efficient closed-loop system with a gain-scheduled MPC was introduced for a steam methane reformer, using a first-principle model for a fired reforming tube reactor to represent process dynamics. The gain-scheduled MPC, considering critical parameters such as outlet methane concentration and temperature, demonstrated adaptive operation, outperforming a PID controller and offering energy savings of 3-5% [219]. This not only improved the overall process efficiency but also enabled better responses to disturbances and variations in operating conditions. The importance of MPC in SMR lies in its ability to increase yield and reduce energy costs, thereby contributing to the sustainability and economic viability of hydrogen production through steam methane reforming. It can be envisioned that MPC has an even bigger role to play in the implementation of future electrified SMR systems. In electrified systems, in addition to temperature, pressures, and flow rates, there is a need for the dynamic optimization of currents and voltages while ensuring the process operation to be within a safe operating regime. In this work, we present our preliminary efforts towards the construction, digitalization, modeling and control of an electrified steam methane reforming reactor.

6.2 Preliminaries

6.2.1 Nomenclature

Definitions of variables used in the modeling of the reactor:

- A_i: Pre-exponential factor for adsorption constant K_i of gas species i [Pa^{-1} for $i = CH_4, H_2, CO$

and unitless for $i = H_2O$]

- A_j: Pre-exponential factor for rate coefficient k_i for reaction j [$mol \cdot Pa^{0.5} \cdot (kg - cat \cdot s)^{-1}$

  for $j = 1$ (SMR reaction), $mol \cdot (Pa \cdot kg - cat \cdot s)^{-1}$ for $j = 2$ (WGS reaction)]

- C_i: Concentration of species i [$mol \cdot m^{-3}$]

- C_{p_i}: Specific heat capacity of gas species i [$J \cdot (mol \cdot K)^{-1}$]

- E_j: Activation energy for reaction j [$J \cdot mol^{-1}$]

- F: Total molar flow of gases [$mol \cdot s^{-1}$]

- F_i: Molar flow of gas species i [$mol \cdot s^{-1}$]

- K_i: Adsorption constant of gas species i [Pa^{-1} for $i = CH_4, H_2, CO$ and unitless for $i = $

  H_2O]

- K_j: Equilibrium constant for reaction j [Pa^2 for $j = 1$ (SMR reaction), unitless for $j = 2$

  (WGS reaction)]

- k_j: Reaction rate constant of reaction j [$mol \cdot Pa^{0.5} \cdot (kg - cat \cdot s)^{-1}$ for $j = 1$ (SMR reaction),

  $mol \cdot (Pa \cdot kg - cat \cdot s)^{-1}$ for $j = 2$ (WGS reaction)]

- P_i: Partial pressure of gas species i [Pa]

- q: Outlet volumetric flowrate [$m^3 \cdot s^{-1}$]

- q_i: Outlet volumetric flowrate of gas species i [$m^3 \cdot s^{-1}$]

- r_j: Rate of reaction for reaction j $[mol \cdot (kg \cdot s)^{-1}]$

- R: Universal gas constant $[J \cdot (mol \cdot K)^{-1}]$

- $\bar{R}$: Alloy tube resistance $[Ohm]$

- T: Reactor temperature $[K]$

- T_{p_i}: Temperature of inlet gas species i $[K]$

- T_s: Temperature of the surroundings $[K]$

- UA: Overall heat transfer coefficient times the heat transfer area $[J \cdot (s \cdot K)^{-1}]$

- V: Reactor volume $[m^3]$

- W: Catalyst weight $[kg]$

- ΔH_{r_j}: Heat of reaction j $[J \cdot mol^{-1}]$

- $\dot{m}_{p_i}$: Mass flow rate of gas species i $[kg \cdot s^{-1}]$

- ρ_i: Density of the gas species i in the reactor $[kg \cdot m^{-3}]$

6.2.2   Process Overview

The overall goal of the joule-heating steam methane reforming process is to convert methane

to produce emissions-free hydrogen gas in an electrically heated steam methane reformer. Instead

of utilizing a conventional fossil fuel-based heating source, electrical heating is used in our work

to heat up the reactor by applying an electric current directly through the reactor tube. In our experimental setup, methane, steam, and argon gases are flown into the reactor tube under certain temperature and pressure to react and produce hydrogen gas. The overall reaction can be demonstrated by the two independent reactions,

$$\text{Steam reforming}: CH_4 + H_2O \rightleftharpoons 3H_2 + CO, \quad \Delta H_{298} = 206.1 kJ \cdot mol^{-1} \tag{6.1a}$$

$$\text{Water gas shift}: CO + H_2O \rightleftharpoons CO_2 + H_2, \quad \Delta H_{298} = -41.15 kJ \cdot mol^{-1} \tag{6.1b}$$
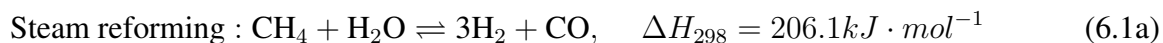
The first reaction (Eq. 6.1a) is the reforming reaction which converts methane and water to carbon monoxide and hydrogen. The second reaction (Eq. 6.1b) is the water-gas shift reaction, which converts carbon monoxide and water to carbon dioxide and hydrogen gas. The overall reaction is endothermic due to the combination of a strongly endothermic reforming reaction and a slightly exothermic water-gas shift reaction. In general, the intrinsic reaction kinetics and mass transport phenomena occurring in the reactor need to be considered when determining the reactor dynamics. In this work, we focus more on the reaction kinetics since the internal and external diffusion resistances of catalyst particles does not appear in the intrinsic reaction rates. Considering the reaction kinetics, the rate determining step is the activation of methane, since it has a stable structure that demands higher energy to break the C-H bond. [192, 193, 194, 195, 196, 197]. In order to catalyze the activation of methane in our experiment, a highly active Ni-based catalyst is used. Furthermore, the experiment is also performed under a high temperature to overcome the energy barrier and increase the reaction rates.

The electrical heating provides a radially near-uniform heat supply with a very small temper-

ature gradient that helps to prevent carbon formation by keeping the gas temperature close to the equilibrium temperature [200]. Also, the temperature gradient between the inner and outer wall of the reactor is very small in contrast to burner-heated reactors, which helps to reduce the thermal stress and extends reactor lifetime [200]. This also means that the thermal gradient across the washcoated catalyst is very small, which improves the catalyst utilization. [200] also suggests that electrically heated SMR reactors can reach the same conversion as industrial burner-heated tubular reformers, but with a reactor volume up to 100 times smaller.

The reactions shown in Eq. 6.1 are complex reactions, as they are parallel in H_2O and series in CO. Furthermore, all the reactants and products are in gas phase. As a result, modeling the reaction kinetics for this system is inherently challenging since, due to the presence of complex reactions, the conversion cannot be used for modeling. In our simulations, the concentrations of hydrogen, methane, water vapor, carbon dioxide, carbon monoxide, and argon are the outputs that depend on the current flowing thorough the reactor, and the current is manipulated to adjust the temperature of the reactor. Prior to the experiment, pure H_2 at high temperature is flown through the tubular reactor to activate the nickel catalyst. Following the H_2 flow, Ar is flown to remove the excess hydrogen remaining in the system before CH_4 and steam are flown. The experimental process flow diagram is shown in Figure 6.2.

We built an experimental steam methane reforming setup at UCLA, whose details are presented in [148]. A similar setup was built by [200] and a series of experiments were conducted to understand the concentration and temperature distribution better inside the reactor. In our work, we initially focus on the development of a nonlinear first-principles-based model that incorporates kinetic rate parameters extracted from experimental data obtained from our experimental reactor.
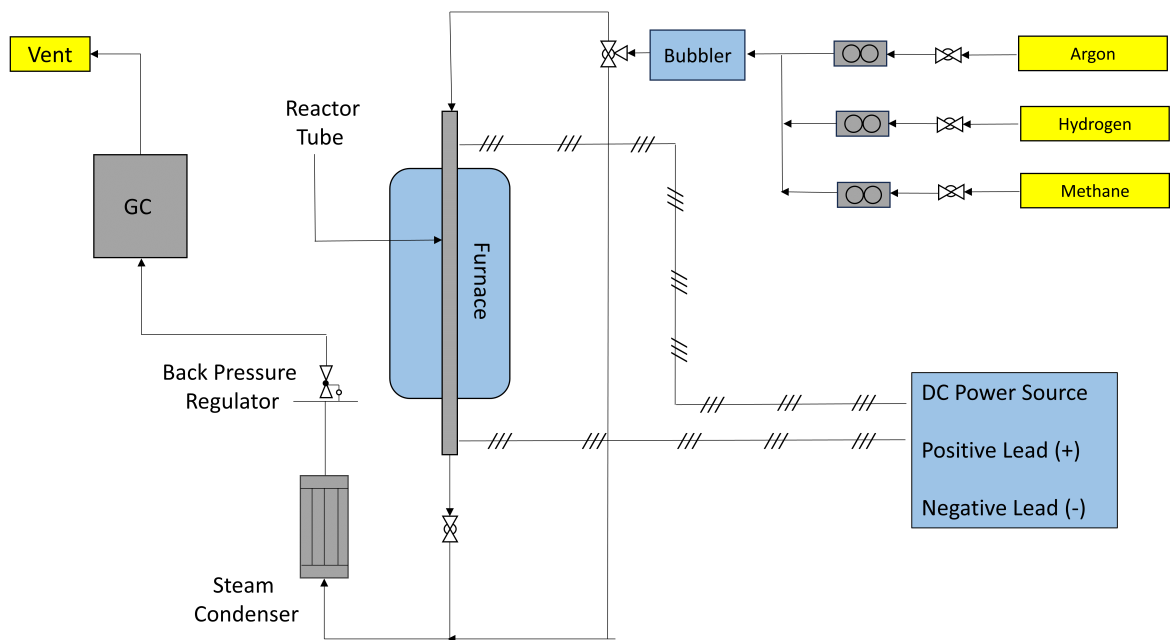
Figure 6.2: Process flow diagram for experimental joule-heating steam methane reforming process.

This model is subsequently used in the present work for the development and evaluation via numerical simulations of a model predictive controller. Eventually, we aim to experimentally implement the model predictive controller to control the joule-heated SMR reactor. Our longer term goal is to use these models as a starting point to build controllers for the more complex process of SMR in a proton membrane reactor setup (where hydrogen production via SMR and hydrogen removal via a proton membrane occur in the same unit) in our laboratory.

6.3 Digitalization of the Experimental Setup

6.3.1 Overview

An experimental setup is built at UCLA for two steam methane reforming reactors. This experimental setup renders it possible to conduct experiments with a proton membrane reactor and a joule heating steam methane reformer. Both reactors have the same inlet gas streams and the outlets are quantified using a gas chromatogram. The inlet flow gases can be directed to either of the processes by arranging the valve configuration. This study focuses on electrically heated steam methane reforming. The majority of the lab equipment for both processes are common. The main difference between both processes is the source of applied potential. The proton membrane reactor uses a potentiostat (Metrohm 302N) to adjust the current flowing through the reactor for separation of hydrogen through a $BaZr_{0.8-x-y}Ce_xY_yO_{3-\sigma}$ (BCZY) membrane, while the joule-heating setup uses a power supply to flow current to heat up the reactor, replacing fossil fuel-based heating [146]. The experimental setup at UCLA is shown in Figure 6.3.

Both setups are digitalized through a connection to a common Laboratory Virtual Instrument Engineering Workbench (LabVIEW) interface. Through LabVIEW, it is possible to manipulate actuators and read data from the sensors in real time. The previous experience with LabVIEW for the electrochemical CO_2 reduction reactor discussed in Chapter 3 considerably accelerated the building of the new interface. It is possible to collect data and perform real-time control with the experimental steam methane reforming setups through LabVIEW.
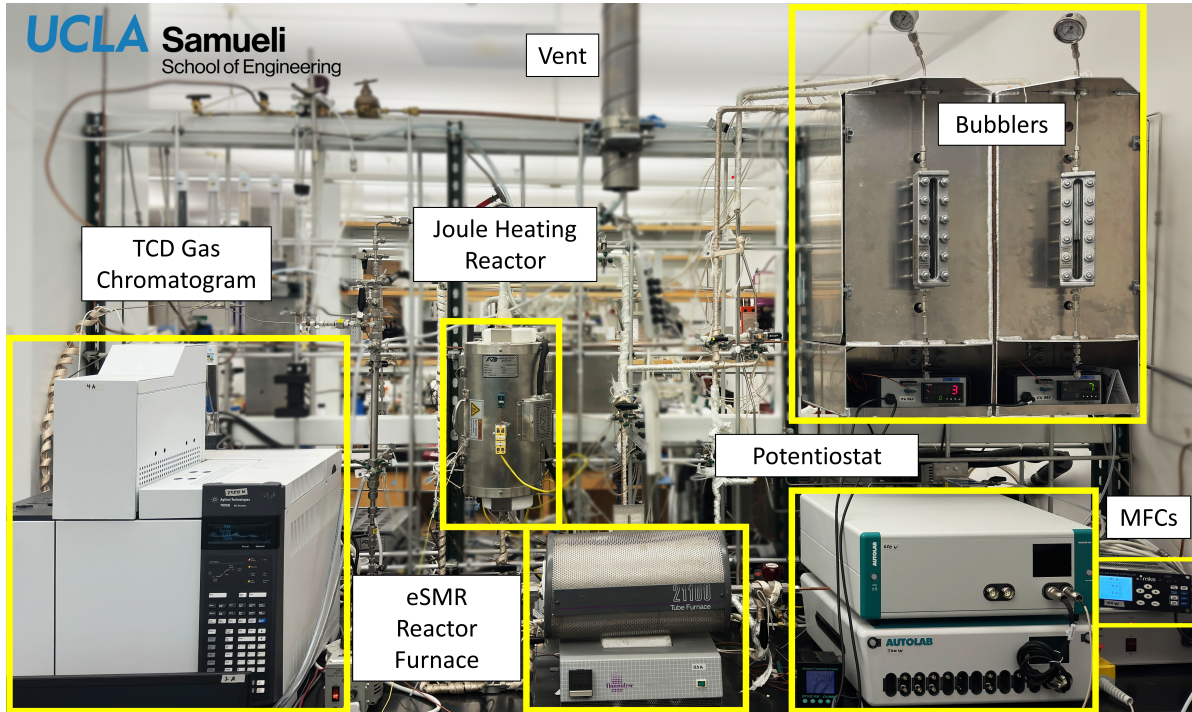
Figure 6.3: Experimental setup for steam methane reforming processes at UCLA.

6.3.2 Sensors and Actuators

The experimental setup involves constant change of parameters such as temperature, pressure, current, etc. In order to measure these changes, multiple sensors, including thermocouples (Omega K-type) and pressure transducers (Omega PX359 - 1KAI), were installed. There are three thermocouples in the system. The first is placed in the middle of the wall on the joule-heated tubular reactor. The second is placed on the inside the steam box. The third one is placed on the wall of the gas flow pipe that is right before the inlet to the reactor.

There is one pressure transducer used for the joule-heated steam methane reforming process, which measures the pressure of the flow system. This pressure can be adjusted and kept constant through back pressure regulators (Equilibar) attached to the system. The aforementioned ther-

mocouples, pressure transducers, and back pressure regulators are digitalized through a National Instruments Compact Rio. A Compact Rio is a reconfigurable, industrial grade data collection system that can work with LabVIEW.

The gas flowrates are set through MKS mass flow controllers (MFC) controlled by MKS 946 Vacuum system controller. It is possible to set flowrates for 5 gases, and the unit is standard cubic centimeters per minute ($sccm$). The MFCs are also connected to the LabVIEW interface and their setpoint can be changed in real time.

The SMR system is equipped with two steam generators that supply steady inlet streams of water vapor to the joule heating and proton membrane reactors. Each steam box houses a bubbler encased within fiberglass thermal insulation. Using a K-type thermocouple, an Arcon temperature actuator, and electrical heating tape, a bubbler is set to a desired steam-to-carbon (s/c) ratio via temperature control. At a specific water temperature, the Antoine equation provides the vapor pressure of steam in the bubbler which is effectively the partial pressure of steam in the inlet gas mixture. The temperature sensor and Arcon actuator amount to PI control over the thermodynamic equilibrium of the inlet gas mixture and liquid water phase in each of the bubblers.

The tubular reactor is the main component of the joule-heated SMR setup. The tubular reactor is shown in Figure 6.4. The reactor tube (Goodfellow Corporation: 72.8% Fe, 22% Cr, 5% Al, 0.1% Y, and %0.1 Zr alloy) has a length of 500 mm, an outer diameter of 6 mm, an inner diameter of 5.4 mm, and is wrapped in a fiberglass based insulation layer to prevent heat losses (not shown in Figure 6.4) [148, 146]. In addition to this layer, there is an insulation furnace (Applied Test Systems, 321C-75-8-12) around the reactor tube. The furnace has a ceramic foam material as an extra layer of insulation. The furnace is used to provide thermal insulation only for these

experiments and was not used to provide heat. Power cables from the power supply are connected to each end of the reactor tube. Using the power supply a potential is applied across the axial direction of the reactor driving a current through the metal tube. In this configuration, heat is uniformly generated along the length of the tube by the passage of electrons through the resistive metal reactor tube. The power supply shown in Figure 6.5. A Chroma programmable DC power supply (62012P-40-120) is the main actuator in the experimental setup. It is connected to the LabVIEW interface through a driver provided by the manufacturer. It is possible to set the applied potential through the power cords. It is also possible to set a current setpoint, and the power supply will accordingly adjust the applied potential to obtain the given current setpoint. To control the joule-heating SMR setup, the power supply can adjust the current flowing through the reactor to provide the heat necessary for the reactions.

Fabrication of the washcoated reactor tubes (procedure with which catalyst is deposited on the tube walls) began with pretreatment of the FeCr Alloy tube followed by application of the coating in a multi-step process. Pretreatment was conducted by running a gentle flow of air through the tube while being heat treated at 950 °C for 10 hours. The purpose of the pretreatment was to form a uniform oxide layer on the surface of the tube for the washcoat to adhere to. All high temperature changes including pretreat and calcination used a ramp rate of 1.4 °C to prevent crack formation in the washcoat or oxide layer. After pretreatment, Zr powder was mixed with 20 mL of deionized water whose pH was adjusted to 10 by adding a drop of 25% aqueous ammonium hydroxide. The solution was mixed with a magnetic stir bar at 300 rpm for 30 minutes, followed by bath sonication for 20 minutes. Immediately after sonication, the slurry was inserted into the tube using a pipet until the tube was filled. The tube was oriented vertically and plugged at the bottom with a nipple

Figure 6.4: Reactor tube and insulation furnace.

designed to slowly release the liquid upon applying a slight pressure inside the tube. Once filled,

a syringe pump was attached to the top of the tube with a Swagelok fitting. The syringe pump

forced a controlled 20 $ml \cdot min^{-1}$ flowrate of air into the top of the tube, forcing the slurry to

drain through the nipple at a steady rate. A total of 50 ml of air was forced through the tube in

this way to ensure all excess slurry was drained. Coated tubes were dried at 110 °C for 1 hour

before calcination at 500 °C for 1 hour. Multiple layers were added to achieve the desired loading
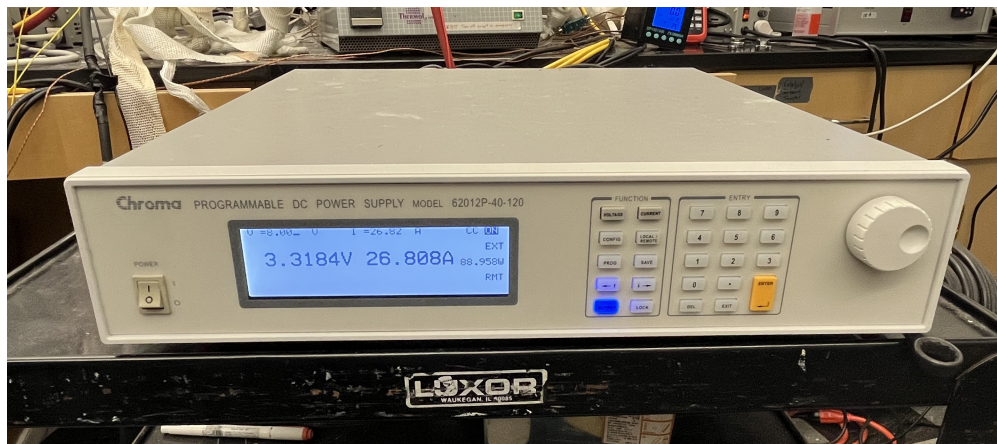
Figure 6.5: Power supply connected to the tubular reactor for electrical heating.

by repeating the process just described. Catalyst addition was done by wet impregnation of the washcoat using a solution of 10 g of nickel nitrate hexahydrate dissolved in 15 ml of deionized water. Similar to how the washcoat was applied, the impregnation solution was added to the tube and allowed to set for at least 1 minute to allow the solution to fill the pores in the washcoat. The solution was drained, and air was gently blown through the tube to remove excess solution. After air drying for 1 hour, the tube was dried at 110 °C for 1 hour followed by calcination at 500 °C for 1 hour. By this technique, a stable washcoat was obtained with a total washcoat loading of 39.8 mg.

An Agilent Technologies 7890B gas chromatogram is used to measure the gas phase products in real-time. A thermal conductivity detector (TCD) is used to quantify each gas product. The product gases first go through a condenser to separate water vapor from the remaining gas products. After condensing the steam, H_2, CO_2, CO, and unreacted CH_4 flows into the GC. It takes 15 minutes for each gas sample analysis. Then, it takes 3 minutes to cool down the GC for a new injection. Thus, the GC can take one gas sample every 18 minutes.

In our setup, the GC measurements are initiated automatically using an external Python code. The main algorithm behind automated GC analysis is described in Chapter 3. After each GC run is finalized, the results are automatically processed by calculating the areas underneath each gas species peak and comparing it to previously calibrated peaks for each gas. An example of GC peaks is shown in Figure 6.6.

Remark 29 *The calculation of the concentrations from the GC analysis are 15 minutes delayed. This situation is a challenge for dynamic processes. However, the present work only uses steady state experiments to evaluate the reaction kinetics parameters.*
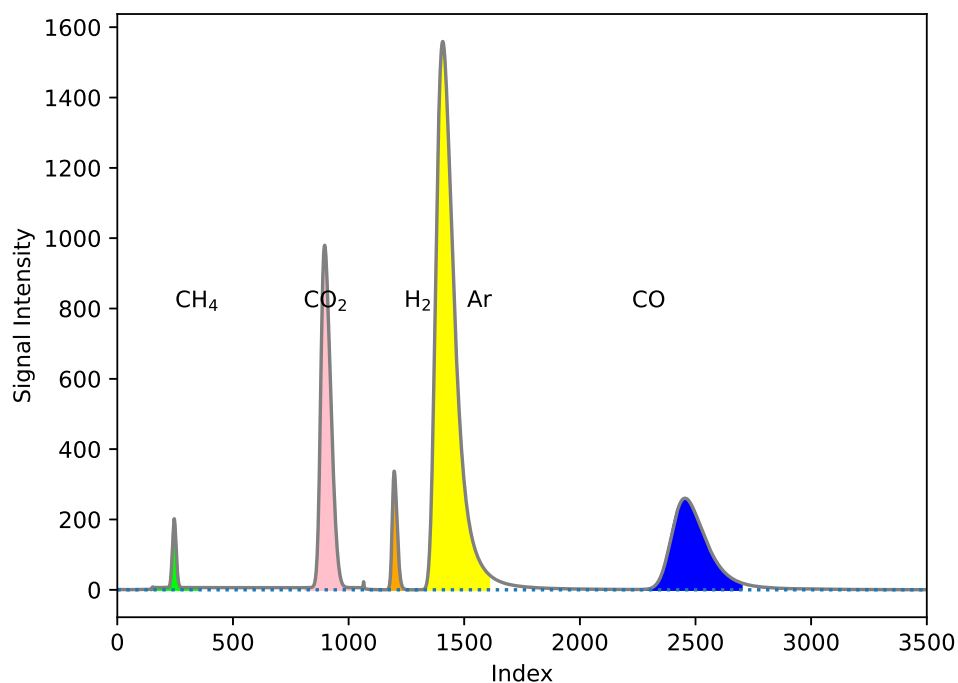
Figure 6.6: Gas chromatogram peaks for quantifying the gas phase products.

6.3.3 Smart Manufacturing Innovation Platform Connection

The LabVIEW interface is connected to Smart Manufacturing Institute's (CESMII) Innovation platform (SMIP). All the data generated from the setup is sent to the platform securely through the query language GraphQL [61]. The profile for the SMR system is shown in Figure 6.7. It is organized and hierarchical; all the attributes are defined under the equipment and each attribute has endpoints to store relevant data.

SMIP will also be used to implement a model predictive controller scheme in the future. Chapter 5 demonstrated the use of SMIP in real time control by transferring data between the lab computer operating LabVIEW and another computer with solver licenses. Thus, SMIP is planned to play a key role in implementing the MPC designed in this chapter on the experimental setup.

Figure 6.7: The SMR system profile on SMIP.

6.4   First-Principles Modeling

The SMR system is challenging to model due to the complex nature of the SMR reactions, all species being in gas phase, transfer phenomena in the reactor, and spatio-temporal variations in the tubular reactor giving rise to partial differential equations. Due to these challenges, the tubular reactor is generally modeled with CFD simulations, which are computationally expensive and not practical to use for real-time predictive control [116, 94]. [95] modeled the SMR process as a burner-heated 1-dimensional fixed-bed tubular reactor surrounded by a large-scale furnace using first-principles equations and industrial data for online deployment of the model. The radiative heat transfer was modeled using temperature from various locations on the tube in conjunction with the Hottel zone method. As the scale of the process setup gets larger, the effects of the transport phenomena becomes more important. In this work, we propose a simplified modeling approach that can be used for a small scale experimental reactor and can be implemented in real-time for predictive control. To this end, the modeling options for approximating the tubular reactor as a continuously stirred tank reactor (CSTR) are explored.

The first-principles model is based on the mole balance of each species in a CSTR. However, compared to a liquid phase CSTR, the gas phase CSTR is harder to model using first-principles because the volumetric flow rate changes with reactions due to the gas phase stoichiometry change after reactions. The reactor is operated at 1 bar and between 600–1000°C. Due to low pressure and high temperature conditions, the ideal gas law is assumed to hold. As a result, in the reactor, $PV = nRT$ must hold at all times, where P is the pressure, and n is the total number of moles. Also, since the experimental setup is a flow system, $Pq = FRT$ must also hold for the flow

calculations, where q is the volumetric flowrate and F is the molar flowrate.

Eq. 6.2 demonstrates the equations used for the calculation of the reaction rates. The kinetic parameters and mechanism are taken from [215] and [1]. Each species has an adsorption coefficient that is a function of temperature. The reaction rates are written in terms of partial pressures due to them occurring in the gas phase.

$$r_{1,SMR} = \frac{k_1}{P_{H_2}^{2.5}} \cdot \frac{P_{CH_4} \cdot P_{H_2O} - \frac{P_{H_2}^3 \cdot P_{CO}}{K_1}}{(DEN)^2} \tag{6.2a}$$

$$r_{2,WGS} = \frac{k_2}{P_{H_2}} \cdot \frac{P_{CO} \cdot P_{H_2O} - \frac{P_{H_2} \cdot P_{CO_2}}{K_2}}{(DEN)^2} \tag{6.2b}$$

$$DEN = 1 + K_{CO} \cdot P_{CO} + K_{H_2} \cdot P_{H_2} + K_{CH_4} \cdot P_{CH_4} + K_{H_2O} \cdot \frac{P_{H_2O}}{P_{H_2}} \tag{6.2c}$$

$$k_j = A_j \cdot \exp\left(-\frac{E_j}{R \cdot T}\right), \quad j = 1, 2 \tag{6.2d}$$

$$K_i = A_i \cdot \exp\left(-\frac{\Delta H_i}{R \cdot T}\right), \quad i = CH_4, H_2O, CO, H_2 \tag{6.2e}$$

In order to model the reactor as a lumped parameter system, it is necessary to write the mass balance equations for each gas. This generates 6 nonlinear ordinary differential equations, as shown in Eq. 6.3. Additionally, since the reaction rates are functions of temperature, it is necessary to have an energy balance, which will be discussed in Sec. 6.5.2.

$$\frac{\mathrm{d}C_{CH_4}}{\mathrm{d}t} = \frac{1}{V_R}\left(F_{CH_4,0} - r_1 \cdot W - q \cdot C_{CH_4}\right) \tag{6.3a}$$

$$\frac{\mathrm{d}C_{H_2O}}{\mathrm{d}t} = \frac{1}{V_R}\left(F_{H_2O,0} - (r_1 + r_2) \cdot W - q \cdot C_{H_2O}\right) \tag{6.3b}$$

$$\frac{\mathrm{d}C_{CO}}{\mathrm{d}t} = \frac{1}{V_R}\left((r_1 - r_2) \cdot W - q \cdot C_{CO}\right) \tag{6.3c}$$

$$\frac{\mathrm{d}C_{H_2}}{\mathrm{d}t} = \frac{1}{V_R}\left(F_{H_2,0} + (3 \cdot r_1 + r_2) \cdot W - q \cdot C_{H_2}\right) \tag{6.3d}$$

$$\frac{\mathrm{d}C_{CO_2}}{\mathrm{d}t} = \frac{1}{V_R}\left(r_2 \cdot W - q \cdot C_{CO_2}\right) \tag{6.3e}$$

$$\frac{\mathrm{d}C_{Ar}}{\mathrm{d}t} = \frac{1}{V_R}\left(F_{Ar,0} - q \cdot C_{Ar}\right) \tag{6.3f}$$

where $F_{i,0}$ is the inlet molar flowrate of species i.

Remark 30 *In the experiment, a relatively short, in terms of axial length, reactor and good thermal insulation are used. Therefore, a lumped parameter system behavior is assumed when the first-principles model is derived, which means there is no spatial variation in temperature and concentration inside the reactor taken into account in the model development. Furthermore, the reaction is expected to consume more heat in the inlet section of the tubular reactor [200], indicating that most of the conversion occurs near the reactor inlet and, for the remainder of the tube, the temperature and concentration profiles do not vary significantly in the axial direction, thereby justifying further the use of a lumped parameter modeling approach.*

6.4.1 Constant Pressure and Temperature Case

The constant pressure and temperature case is considered as a starting point. In this case, the total concentration in the reactor, C_T, is constant according to the ideal gas law,

$$C_T = C_{CH_4} + C_{H_2O} + C_{CO} + C_{H_2} + C_{CO_2} + C_{Ar} = \frac{P}{RT} \tag{6.4}$$

Due to the species being in gas phase, the outlet volumetric flowrate, q, is a function of temperature and cannot be assumed constant for all temperatures. Since q appears as a new variable, one new equation should be introduced, which is Eq. 6.4. This equation originates from the ideal gas law since $C_T = F/q$, where F is the total outlet molar flow rate, and indicates that the total gas concentration in the reactor is a function of only temperature and pressure. As a result, the total concentration inside the reactor does not change under constant temperature and pressure, and its differential with respect to time can be set to zero as follows:

$$\frac{dC_T}{dt} = \frac{dC_{CH_4}}{dt} + \frac{dC_{H_2O}}{dt} + \frac{dC_{CO}}{dt} + \frac{dC_{H_2}}{dt} + \frac{dC_{CO_2}}{dt} + \frac{dC_{Ar}}{dt} = 0 \tag{6.5}$$

where the differentiation of the total concentration term is written as the summation of concentration differentials with respect to time for each of the 6 gas species, which are already written explicitly in Eq. 6.3. Substituting the individual species' mass balances into Eq. 6.5, we obtain the relation,

$$\frac{dC_T}{dt} = \frac{1}{V_R} \left(F_{CH_4,0} + F_{H_2O,0} + F_{H_2,0} + F_{Ar,0} + 2r_1 W - qC_T \right) \tag{6.6}$$

whose right-hand side can be equated to zero to yield

$$F_{T0} + 2r_1 W = q \frac{P}{RT} \tag{6.7}$$

from which the final expression for the outlet volumetric flow rate can be obtained as

$$q = \frac{F_{T0} + 2r_1 W}{\frac{P}{RT}} \tag{6.8}$$

This equation shows that the outlet flowrate is a function of the reforming reaction rate. After replacing all occurrences of q in Eq. 6.3 by Eq. 6.8, the system of differential equations for constant temperature can be solved with numerical integration and then compared via experimental results. In the experimental setup, the constant temperature and pressure conditions can be achieved by controlling the power supply to regulate temperature through current and using pressure regulators.

6.4.2 Variable Temperature Case

The SMR reactor can be heated by electricity. Specifically, the current across the reactor can be manipulated to heat the reactor tube, which will change the temperature of the reactor. A higher temperature can produce a higher amount of hydrogen and increase methane conversion. Hence, it is necessary to derive the dynamic model of the case with variations in temperature. However, in this case, the total concentration changes according to the change in temperature. Therefore, the expression of the volumetric flow rate also changes and now depends on the derivative of the temperature with respect to time. The derivation of the expression for the outlet volumetric flow rate is done for the temperature-varying case using the same steps as Eq. 6.4–6.8 and shown in Eq.

6.9.

$$\frac{\mathrm{d}C_T}{\mathrm{d}t} = -\frac{1}{T^2} \cdot \frac{P}{R} \cdot \frac{\mathrm{d}T}{\mathrm{d}t} \tag{6.9a}$$

$$\frac{\mathrm{d}C_T}{\mathrm{d}t} = \frac{1}{V_R} \cdot (F_{T,0} + 2 \cdot r_1 \cdot W - q \cdot C_T) \tag{6.9b}$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = T^2 \cdot \frac{R}{P} \cdot \frac{1}{V_R} \cdot \left(q \cdot \frac{P}{R \cdot T} - F_{T0} + 2 \cdot r_1 \cdot W \right) \tag{6.9c}$$

$$q = \frac{F_{T0} + 2 \cdot r_1 \cdot W}{\frac{P}{RT}} + \frac{V_R}{T} \cdot \frac{\mathrm{d}T}{\mathrm{d}t} \tag{6.9d}$$

6.4.3   Steady State Simulation and comparison with experimental results

The steady state simulation results at different temperatures are shown in Figure 6.9. When the temperature is below approximately 600 °C, the volumetric flows for hydrogen and methane going out of the reactor remain constant, which means there is nearly no reaction. When the temperature is higher than 600 °C, the hydrogen volumetric flow rate starts to increase, and the methane volumetric flow rate starts to decrease. The reason behind this phenomenon is that the reaction rate increases with the increment of temperature.

247
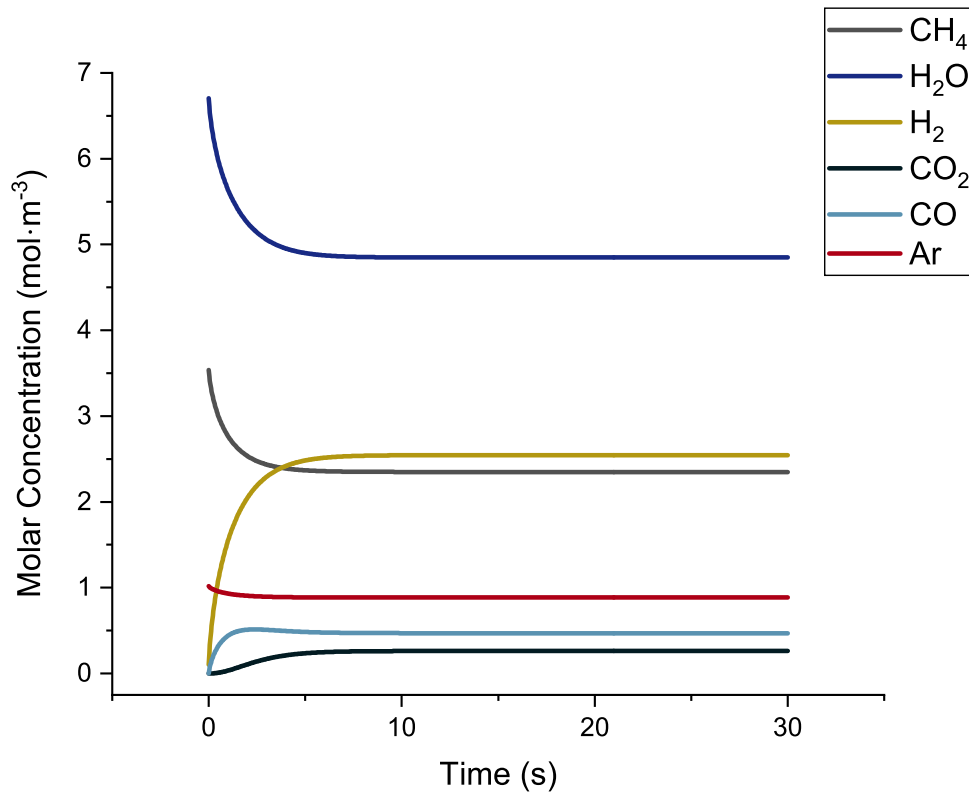
Figure 6.8: Open loop dynamic evolution based on first-principles equations under constant temperature (800 °C) and constant pressure (1 bar).

Remark 31 *In the constant temperature and pressure case, the time to reach the steady state is below 10 seconds, indicating extremely fast dynamics.*

6.5 Model Parameter Estimation Using Experimental Data

6.5.1 Reaction Kinetics Estimation

Steam methane reforming reaction mechanisms are taken from [215]. However, the catalyst used in [215] is different from the catalyst used in our experimental study. As a result, the reaction parameters presented in [215] are not expected to be the same as those in our experiments. The preparation of the catalyst will also impact the distribution of the active sites and the reaction kinetics. Thus, the first step towards modeling the system is to estimate the parameter values in our specific setup.

As discussed in Section 6.4, this experimental setup is modeled as a continuous stirred tank reactor (CSTR; lumped parameter modeling). Using the concentrations at steady state, the lumped parameter model is compared to the experimental observations for gases. The activation energies presented in [215] for reforming and water gas shift reactions were re-calculated to align closely with our experimental concentrations. The comparison between the experimental data and the model calculations after adjusting the activation energy of the reforming reaction is shown on Figure 6.9. An optimization problem was established to minimize the difference between calculated pre-exponential factors and activation energies, and the experimental observations. After solving the optimization problem, the pre-exponential factors and activation energies for the reforming reaction are taken to be $4.22 \cdot 10^{16} \, mol \cdot Pa^{0.5} \cdot (kg - cat \cdot s)^{-1}$ and $384.5 \, kJ/mol$ respectively, and

for the water gas reaction are taken to be $4.22 \cdot 10^7 \; mol \cdot Pa \cdot (kg - cat \cdot s)^{-1}$ and $128.9 \; kJ/mol$, respectively.

Remark 32 *Due to the nonlinearity of the optimization problem, different solvers may find different solutions corresponding to different local optima. In our case, the best fit that gives the closest alignment with the experimental data among various Python library $SciPy$ solvers is taken to be the final estimate of the parameters. Following this methodology, the least-squares sequential quadratic programming method was used as the final optimization solver.*

Remark 33 *[200] suggests that the temperature difference between the inlet region and the outlet region of the reactor may reach up to 300 °C. Our approach approximates the tubular reactor as a lumped parameter system and does not account for the spatial temperature gradient. This approach will be improved by modeling with multiple lumped parameter models in series that will use the temperature values from multiple thermocouple sensors attached to the tubular reactor. The reaction kinetic parameters will be calculated accordingly.*
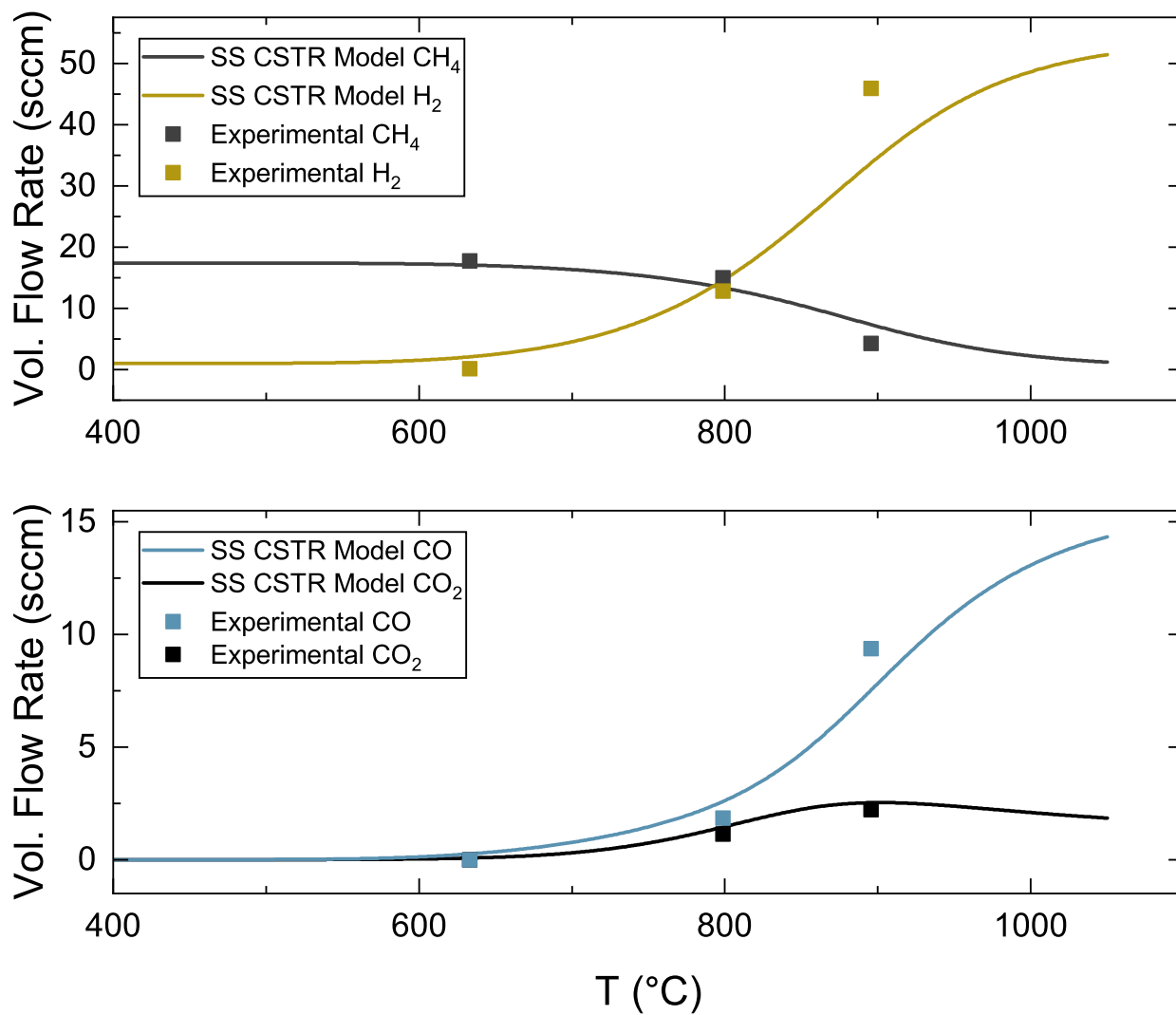
Figure 6.9: Comparison of lumped parameter model steady-state with experimental observations.

6.5.2 Temperature Dependence with Respect to Time

As described in Section 6.4.2, the system of differential equations needs to include one equation for the energy balance. This energy balance can be approximated as follows:

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{I^2\bar{R} + \sum_i \dot{m}_{p_i}C_{p_i}(T_{p_i} - T) - Wr_{SMR}\Delta H_{SMR}(T) - Wr_{WGS}\Delta H_{WGS}(T) + UA(T_s - T)}{\sum_i \rho_i C_{p_i}V}$$

(6.10)

The detailed energy balance of Eq. 6.10 is a modified version of the energy balance presented in [45], adapted for electrically heated steam methane reforming. The heat input is replaced by $I^2\bar{R}$, which is the power supplied by the flow of electrons. This equation requires further experiments to estimate the value for the heat transfer coefficient (U), as well as the coefficients for radiant and convective heat losses to the environment. As the lack of knowledge of these parameters in Eq. 6.10 renders it unusable in practice until further experimental data is available, the temperature change with respect to time is instead approximated using the available experimental data by fitting the data of current and temperature vs time to a first-order dynamic model. The derivation of such a first-order model is shown in Eq. 6.11, starting from the Laplace domain and its transition to time domain.

$$\frac{T'(s)}{I'(s)} = \frac{K}{\tau s + 1}$$

(6.11a)

$$T'(s) = \frac{K}{\tau s + 1} \cdot I'(s)$$

(6.11b)

$$T'(t) = I'(t) \cdot K \cdot (1 - \mathrm{e}^{-t/\tau})$$

(6.11c)

$T'(s)$ and $I'(s)$ are the deviation form notations for temperature and current, respectively, while K and τ are the process gain and time constant, respectively. If the K and τ values can be extracted from the experimental setup by fitting the temperature dynamics following a step change in the current, the time-derivative of the temperature can be written as

$$\frac{\mathrm{d}T}{\mathrm{d}t} = -\frac{T - T_{ss}}{\tau} + K \cdot \frac{I - I_{ss}}{\tau} \tag{6.12}$$

where T_{ss} and I_{ss} are the initial steady-state values. Eq. 6.12 is simpler and, in this work, is used in lieu of Eq. 6.10. With this approach, the reaction heat generation, power supply efficiency, and heat transfer coefficients are accounted for in the dynamic behavior approximation via K and τ. In order to estimate the values of K and τ, it is necessary to apply a step change in current and record the dynamic behavior of the temperature. Thus, when the current was at 32 A, and the system was at steady state, the current was reduced to 0 A while the system was active, and the corresponding data shown in Figure 6.10 was collected. In addition to this, steady state temperature values are recorded at various currents, and the data is shown in Figure 6.11.

The data shown in Eq. 6.10 is fitted to a first-order model of the form of Eq. 6.11c to get the process gain (K) and process time constant (τ) values, which were calculated to be 14.54 K/A and 284 seconds, respectively. The first-order model fit and experimental variation of the temperature with respect to the step change in current are compared in Figure 6.12.

Remark 34 *The step change in current is aimed to be large to capture a broader range of temperature change. However, to avoid harming the catalyst morphology, the change is made in the cooling direction since rapid, large increases in the temperature are detrimental to the catalyst.*
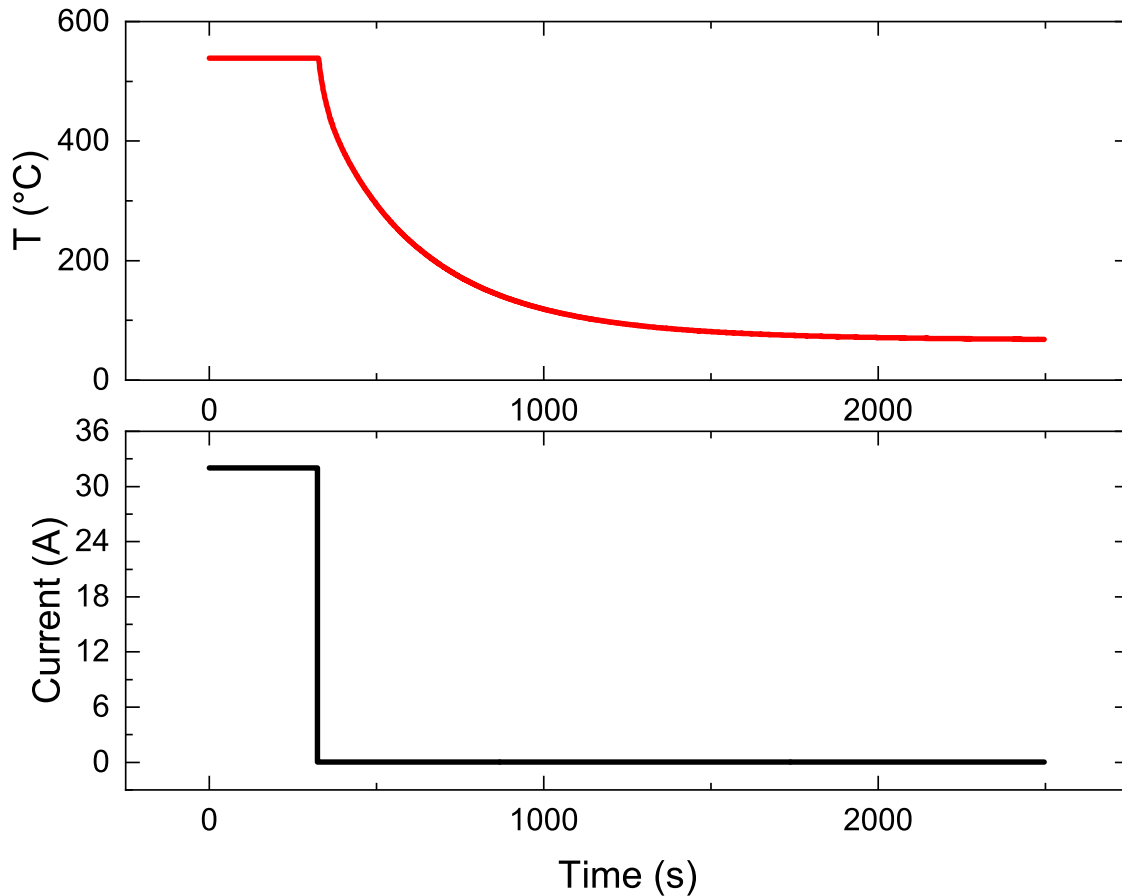
Figure 6.10: Dynamic change in temperature with respect to a step change in current.

Remark 35 *Figure 6.11 demonstrates that the relation between steady-state current and temper-ature is a second order polynomial and mildly nonlinear. The approach mentioned in this section is for first-order linear processes. Thus, this approach can be improved with an incorporation of other first-order models for higher temperature ranges or can be replaced by a data-driven model, such as a recurrent neural network. This would also account for the reaction dynamics at higher temperature regimes.*

Remark 36 *The first-order model will be valid for a specific inlet flow of CH_4. The amount of CH_4 in the inlet is pivotal for the reaction kinetics. Thus, the first-order model parameters should*
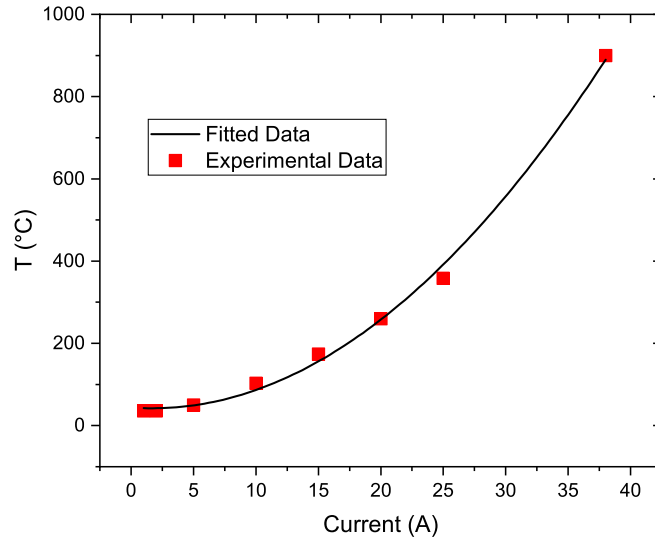
Figure 6.11: Temperature values against current setpoints at steady state.

be calculated for various CH$_4$ inlet conditions for a more comprehensive model.
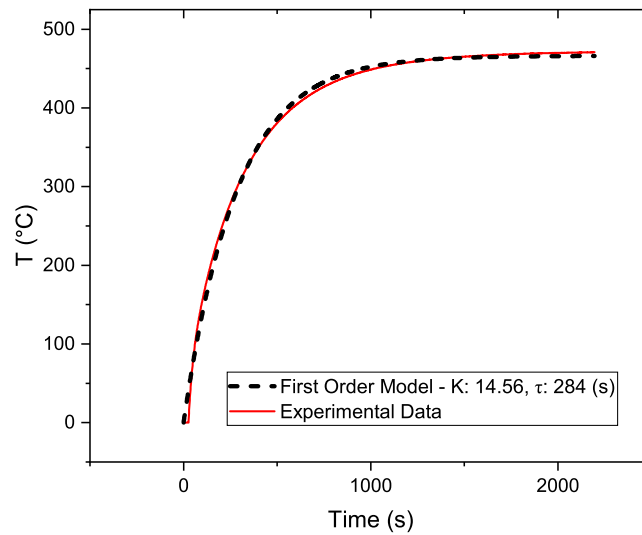
Figure 6.12: Comparison of first order dynamic model with experimental observations.

6.6 Feedback Control

Our work aims to build a feedback control architecture for the experimental SMR setup that is robust to disturbances. It is possible to decide the best controlling strategies by accounting for the limitations of the process, since first-principles and data-driven dynamic temperature variation models are available to simulate the experimental behavior. For this purpose, a PI control scheme is compared to a model predictive controller. In the following simulations, the pressure is assumed to be constant at 1 bar, while the temperature is varied to regulate the reaction rates to drive the H_2 production. The implementation of the constant pressure in the experimental setup in future work will be realized by using a back pressure regulator.

6.6.1 Tuning of a Proportional Integral Controller and Model Predictive Controller

PI controllers are based on feedback sensor data and do not require a process model. However, the parameters of the PI controller must be tuned. The PI control equations are as follows:

$$u = K_C \cdot \left[(y_{sp} - y) - \frac{1}{\tau_I} \cdot \int_0^t (y_{sp} - y) \ \mathrm{d}t \right] \tag{6.13a}$$

$$u = I - I_s \tag{6.13b}$$

$$y_{sp} = C_{H_2,\,sp} - C_{H_2,s} \tag{6.13c}$$

$$y = C_{H_2} - C_{H_2,s} \tag{6.13d}$$

$$0A \leq I \leq 70A \tag{6.13e}$$

The actions of the PI controller are limited to be below 70 A of current, since currents higher than 70 A will be detrimental to the catalyst and cause coking.

The model predictive controller takes the form of the following optimization problem:

$$\mathcal{J} = \min_{u} \int_{t_k}^{t_{k+N_h}} L(\hat{x}(t), u(t)) \; \mathrm{d}t \tag{6.14a}$$

$$\text{s.t.} \quad \hat{x}(t) = ODE(x(t), u(t)) \tag{6.14b}$$

$$L(\hat{x}(t), u(t)) = A(\hat{x}(t) - x_{sp})^2 + B(u(t) - u_{sp})^2 \tag{6.14c}$$

$$t \in [t_k, t_{k+N_h}) \tag{6.14d}$$

$$|u(t_k) - u(t_{k-1})| \leq 0.1 \tag{6.14e}$$

$$0A < u(t) < 70A \tag{6.14f}$$

where $L(\hat{x}(t), u(t))$ is a cost function to be minimized, ODE is the nonlinear dynamic process model derived in Sec. 6.4, $\hat{x}(t)$ is the state prediction over the horizon using the ODE model, A and B are tunable weight parameters, and x_{sp} and u_{sp} are the setpoints for the H_2 concentration and current, respectively. Similar to the PI controller, the optimization problem is aimed to produce outputs within the bounds of 0 and 70 A. However, since an MPC is inclined to drive the process very fast and hit the input bounds very early for a quick response, an additional constraint is placed that bounds consecutive current changes to a maximum of 0.1 A per second. This is also important for the catalyst morphology, since a slow increase in current will not harm it.

The MPC will try to minimize the quadratic cost function over a horizon of 10 seconds. The estimated current setpoint value (u_{sp}) is calculated by using the steady-state equations, which are obtained by equalizing the mass balance equations shown in Eq. 6.3 to 0. Here, the hydrogen

concentration setpoint is fixed, and the other gas concentrations and temperature are calculated using the process operating conditions. The current value corresponding to the calculated temperature is estimated from the correlation shown on Figure 6.11. The weight parameters A and B for the quadratic cost function are taken to be 100 and 0.01, respectively. The MPC problem will be solved in real time in the future work on the experimental setup, thus requiring the MPC problem to be solvable within the sampling period. To ensure this, the MPC optimization problem will be solved every second using a sequential quadratic programming (SQP) solver, which is a computationally-efficient solver [12].

All closed-loop simulations start from the same steady state that is described in Figure 6.8 under constant temperature at 800 °C at 40 A and constant pressure at 1 bar. After it reaches the steady state, it will remain there until the control action starts at the 100th second. The initial H_2 steady-state concentration will be 1 mol/m^3, and the controller is expected to drive the H_2 concentration to 4.5 mol/m^3. This number is 3 times the initial steady-state concentration and is a sufficiently large change to show that the controller is successful.

The MPC simulation results are shown in Figure 6.13. After the setpoint change is introduced at the 100th second, the controller starts to increase the temperature at 0.1 A per second to quickly drive the process towards the setpoint. Around 300 seconds, the controller reaches its peak value at 60 A, and it can be seen that the process is very close to the setpoint. After this point, the controller starts to reduce the current slowly to finalize the process evolution to the setpoint, and the process settles at a steady state after around 600 seconds, where the temperature and current reach their respective steady states of 890 °C and 46 A. The MPC is solved quickly, drives the process output to the desired setpoint without offset, and all the input constraints are respected.

258

Remark 37 *The reaction reaches the initial steady state under 10 seconds. However, even with an MPC, it takes more than 220 seconds to reach a new steady state. The reason for this is that the temperature increase caused by the power supply has slower dynamic evolution compared to reaction dynamics.*
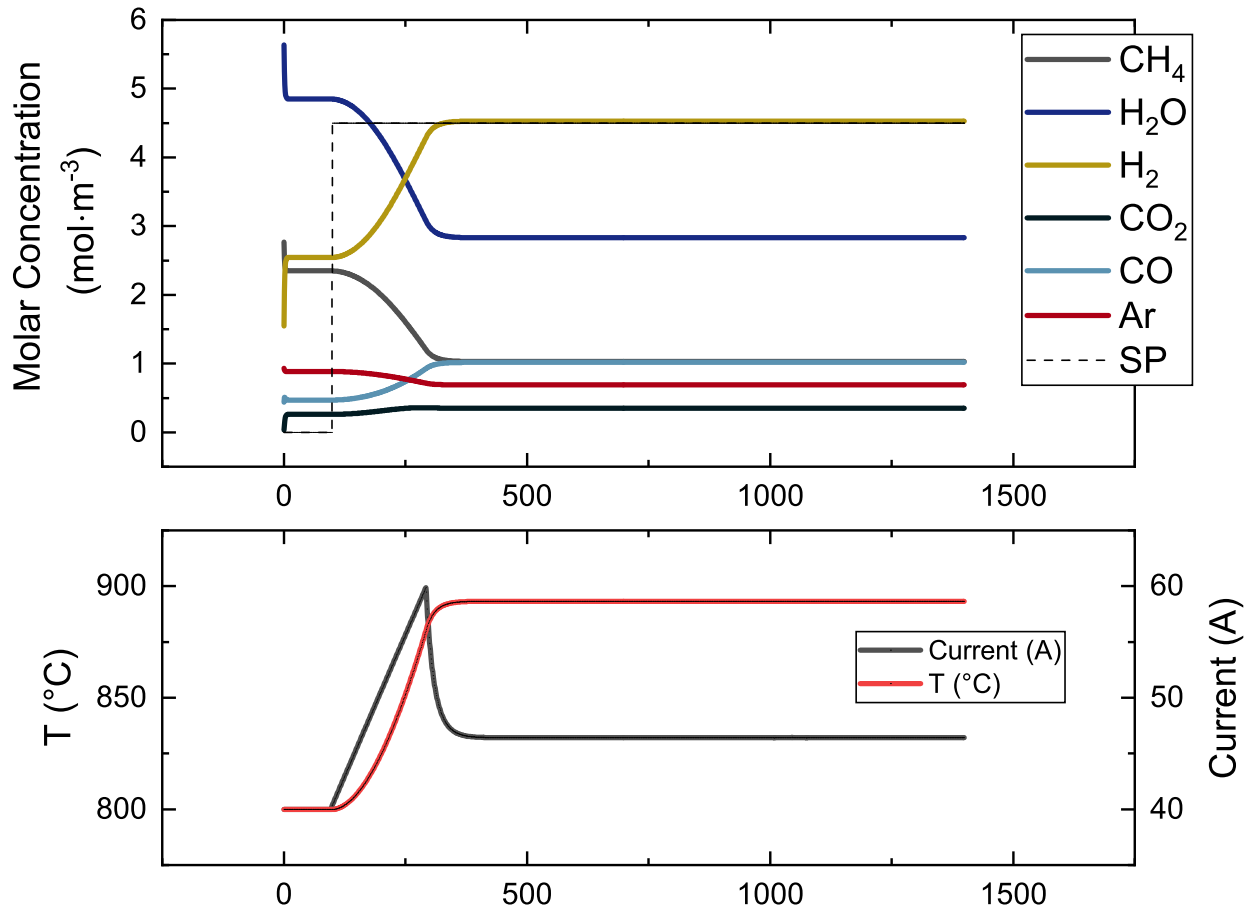
Figure 6.13: Closed-loop response under MPC is fast and without offset while input constraints are respected.

In order to verify that the proposed MPC scheme is highly effective, it is compared with a PI controller. The PI controller is tuned to have no overshoot or oscillations. The parameters for the PI controller are taken to be $K_c = 0.12$, and $\tau_i = 20$. The comparison between the proposed PI controller and MPC is presented in Figure 6.14. Both controllers succeed in driving the process to the setpoint. It takes around 1300 seconds for the PI controller, while this duration was around 220 seconds for the MPC. The effectiveness of the MPC is due to the initial increase in the current input. Thus, the MPC is proven to be fast and efficient.
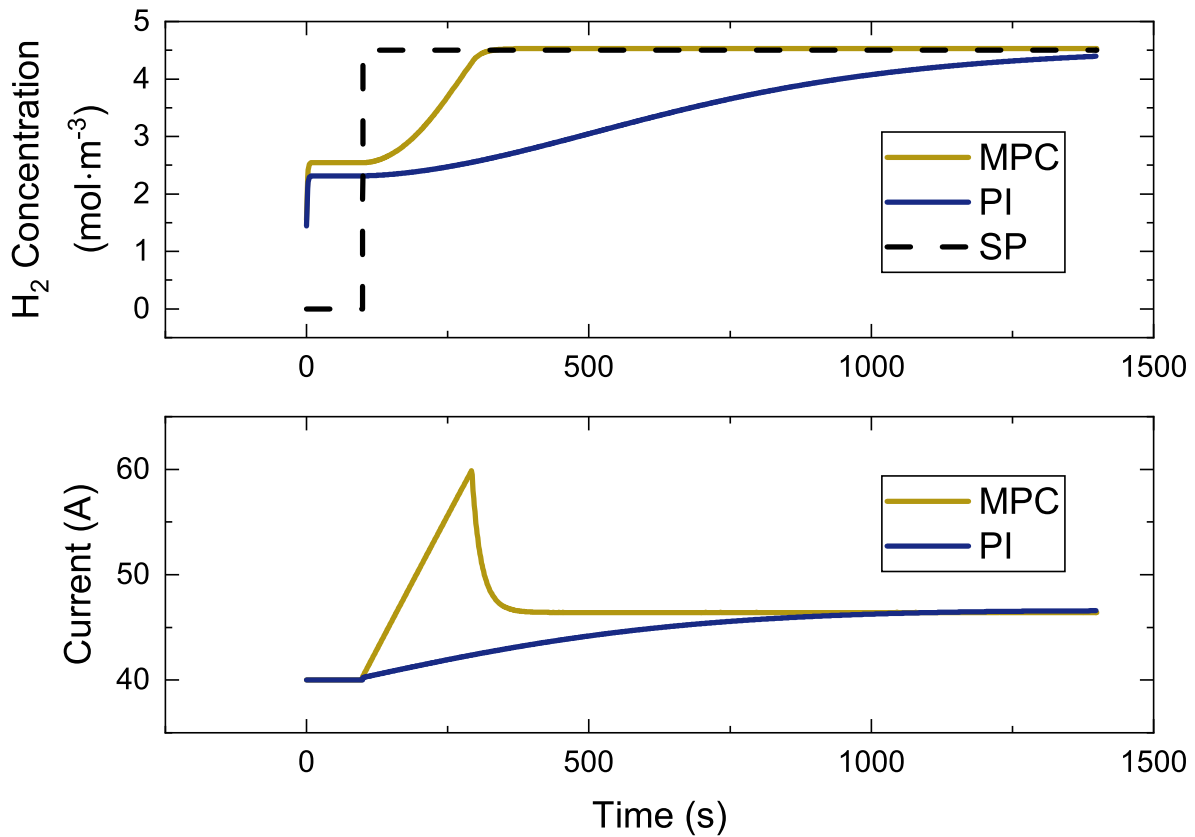
Figure 6.14: Closed-loop response under MPC is superior to the one under PI control.

6.6.2 Disturbance Rejection

In order to prove further effectiveness of the proposed MPC, it is necessary to demonstrate a robust performance against disturbances. One of the possible disturbances in the experimental setup is the steam flowrate. As explained in Sec. 6.3, there is a temperature control box that regulates the temperature of the bubblers. If the control box provides more heat, the steam flowrate going into the SMR reactor increases. Under normal circumstances, the steam-to-carbon ratio is arranged to flow 33 $sccm$ of water vapor into the reactor. In case of a malfunction in the temperature control box, the amount of steam sent to the reactor might increase. In the presence of such a disturbance, the behavior of both the PI controller and MPC are examined.

The MPC performance against a 10% increase in steam feed flowrate is shown in Figure 6.15. Compared to Figure 6.13, it can be seen that the initial water steady state concentration is higher. Similar to the disturbance-free case, the controller gradually increases the current and then slowly decreases it to the setpoint value as the concentration of H_2 gets closer to the setpoint. However, in the +10% disturbance case, it is seen that the current is increased up to 61.93 A, which is also higher than the maximum current for the no-disturbance run. To quantify the differences in the dynamic responses with and without the disturbance in the steam box, we calculate the production of H_2 in terms of (standard) volumetric flow rate at the end of the run. For the case without disturbance, shown in Figure 6.13, the final volumetric flow rate is 32.85 $sccm$, corresponding to a temperature of 893.38°C and concentration of 4.5 mol/m^3. Under the disturbance, if the same temperature of 893.38°C is maintained, the higher water vapor content in the inlet causes the steady-state concentration of H_2 to be reduced to 4.2 mol/m^3, yielding a reduced H_2 production

rate of 31.57 *sccm*. However, under the MPC, using a higher peak current of 61.93 A, the final

steady-state temperature is now adjusted to 913.56°C to compensate for the excess water vapor in

the feed. As a result, the hydrogen concentration once again reaches its setpoint of 4.5 mol/m^3, and

the H_2 production increases to 35.59 *sccm*. This may be due to the extra water vapor in the feed

stream decreasing the partial pressure of CH_4 and, consequently, the rate of the reforming reaction

as per Eq. 6.2a. Thus, in order to reach the same level of H_2 production as the disturbance-free

case, the controller needs to increase the heat provided to the system to boost the reaction rate of

the reforming reaction.
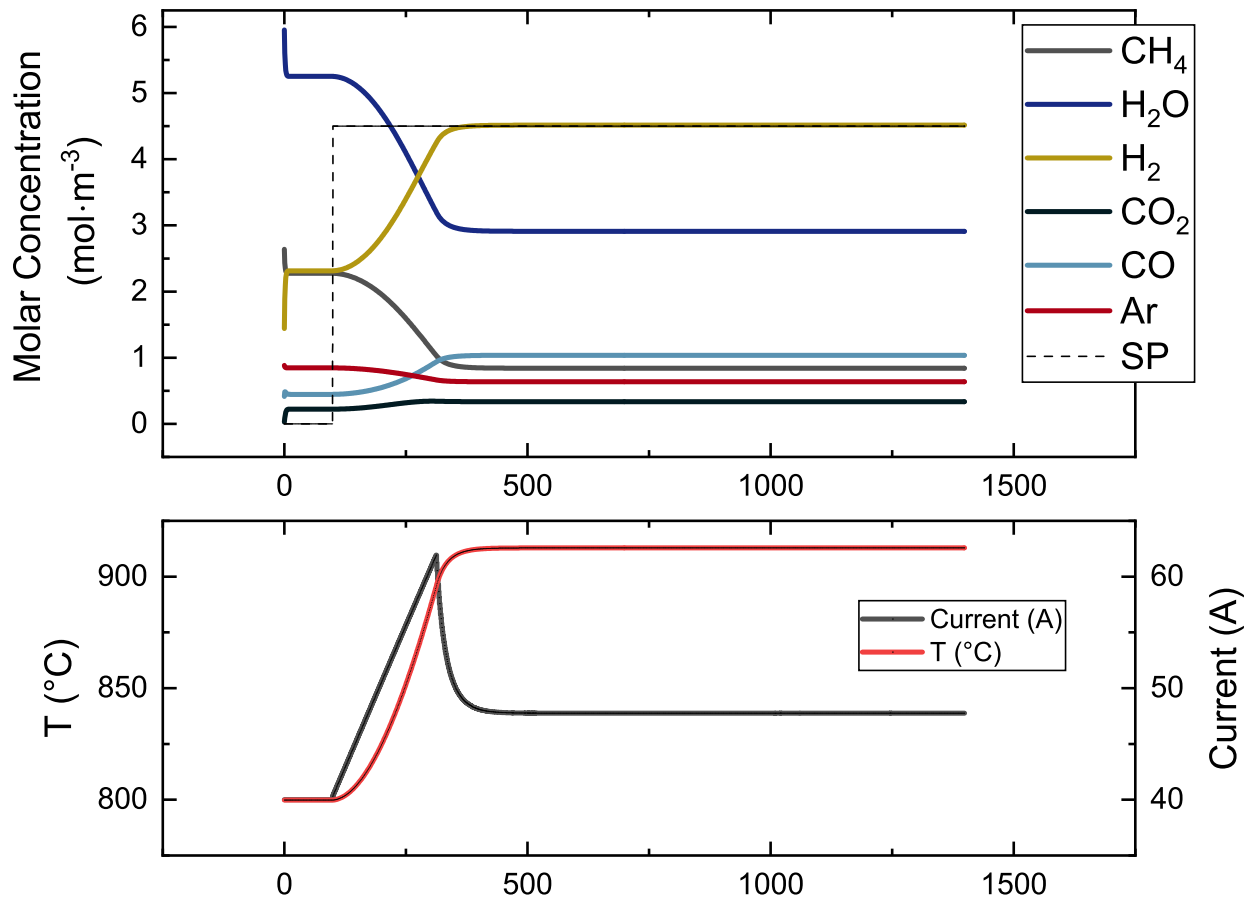
Figure 6.15: Closed-loop response of the MPC under +10% disturbance in vapor feed flow rate.

The behavior of the PI controller is demonstrated in Figure 6.16 against the 10% increase in the steam flowrate case. The same PI controller parameters are used as in Figure 6.14. Although the PI controller manages to drive the process to the setpoint, it takes longer than the MPC. The increase of steam flowrate requires more heat input, thus increasing the time to reach the setpoint from 1300 seconds to 1900 seconds. On the other hand, the time required for the MPC to drive the process to the setpoint under the disturbance had increased from 220 seconds to 370 seconds, showing that MPC is faster than the PI controller, and robust, in the disturbance case.
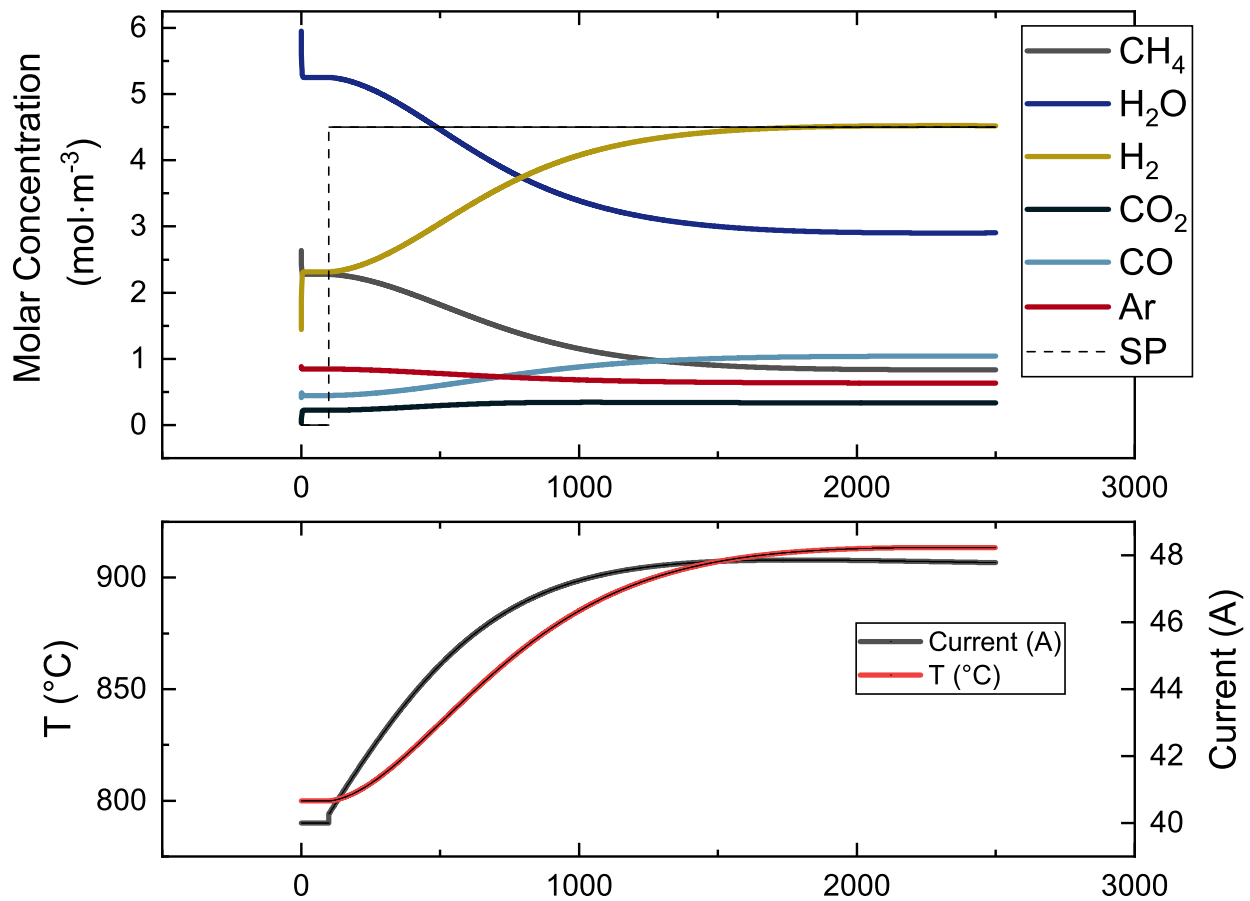
Figure 6.16: Closed-loop response of the PI controller under +10% disturbance in vapor feed flow rate.

Chapter 7

Feedback Control of an Experimental Electrically-Heated Steam Methane Reformer

7.1 Introduction

Steam methane reforming (SMR) is an industrial process that converts methane and steam into hydrogen and carbon dioxide through the chemical reactions shown below:

$$\text{Steam methane reforming}: \text{CH}_4 + \text{H}_2\text{O} \rightleftharpoons 3\text{H}_2 + \text{CO}, \quad \Delta H_{298} = 206.1 \ kJ \cdot mol^{-1} \quad (7.1\text{a})$$

$$\text{Water gas shift}: \text{CO} + \text{H}_2\text{O} \rightleftharpoons \text{CO}_2 + \text{H}_2, \quad \Delta H_{298} = -41.15 \ kJ \cdot mol^{-1} \quad (7.1\text{b})$$
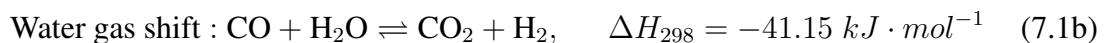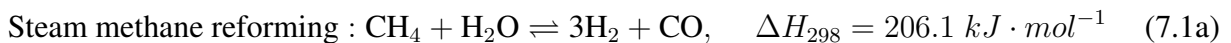
The highly endothermic reforming reaction and the slightly exothermic water gas shift reaction occur in series and parallel. Traditionally, these reactions take place in a packed-bed reactor placed in a fired furnace that burns fossil fuels. About thirty years ago, [168] introduced the idea of an

electrically-heated SMR. Recently, [200] conducted a detailed experimental and modeling study to investigate the process and its potential to replace the conventional furnace heating with electrical heating, and reported that electrical heating leads to lower reactor volumes, causes less waste-heat, and reduces CO_2 emissions if renewable electricity is used. Also, a Joule-heating energy source provides radially uniform heat distribution, causing the gas mixture to be at close-to-equilibrium conditions through a washcoated catalyst. The resistance-heated SMR process is further expected to reduce carbon formation, thus increasing the carbon conversion to hydrogen. The adoption of a Joule-heating process has also a potential to reduce global CO_2 emissions [200]. While the transition to a resistance-heated SMR process is still in its infancy, the current bench scale efforts can offer strategies and explorations to ease the adoption process.

With respect to previous work on this topic, [201] experimentally examined process dynamics including the effect of the heating rate on the start-up phase of the reactor, cyclic heating, and carbon formation at transient and steady-state conditions. [3] focused on improving coil geometry and magnetic field frequency of an induction-heating setup and showed that thermal efficiency can be improved up to 12% compared to their initial bench scale coil geometry. In [223], a Rh/Al_2O_3 catalyst coated with silicon carbide (SiC) layer was washcoated across the reactor tube. The SiC layer increased the resistance for heating, and experimental results were reported to reach full conversion of methane. To explore further use of ohmic heating in reforming processes, [46] moved to a pilot plant phase to test a process that uses biogas to produce syngas.

In addition to endeavors that improve SMR reactor heat flux, process design, and catalysts, one key component that can optimize the electrified SMR process is efficient feedback control to optimize the speed of transition to different setpoints. The SMR reaction, especially in a fired-

heater where the temperature gradients vary significantly in both axial and radial directions, is very complicated and thus very difficult to model. The reactions occur in a tubular reactor, and considering the variation in heat, mass, and momentum transfer coupled with reaction kinetics and temperature- or pressure-dependent variables like volumetric flowrates, such modeling requires solving partial differential equation (PDE) models. Consequently, commercial PDE solvers (e.g., COMSOL, Ansys Fluent) are widely used to model the SMR process. Specifically, [204] simulated the reforming process in a furnace using a detailed computational fluid dynamics (CFD) model, linearized the model using the simulation data, and solved a quadratic programming problem to implement a model predictive control (MPC) scheme to the CFD model. More recently, [177] built an Aspen Dynamic model to generate process data and carried out state-space identification to employ an MPC scheme in a computational study. However, efforts to experimentally implement and evaluate efficient real-time SMR process control are limited. In particular, and to the best of our knowledge, the advanced control of a Joule-heated SMR system with a washcoated catalyst has not been studied.

Chapter 2, Chapter 4, and Chapter 5 have demonstrated successful single-input single-output, multi-input multi-output, and predictive control of experimental electrochemical reactors. Furthermore, Chapter 3 presented how experimental setups can be digitalized using appropriate software and tools. Utilizing our experience in digitalization, modeling, and control, in Chapter 6, we proposed a lumped parameter modeling approach that can be rapidly optimized in real-time. For this lumped parameter model, mass and energy balance equations were derived for varying parameters, and in a detailed computational study, this model was used in an MPC to demonstrate that the H_2 concentration can be effectively driven to the desired H_2 setpoint. However, this compu-

tational work did not account for the experimental challenges, such as unmeasured variables like the molar or volumetric flowrates of steam. Thus, in a recent work [29], we developed a model predictive control method accounting for the issues encountered in the experimental setup, and in the present work, the MPC scheme proposed in [29] and Chapter 6 is implemented experimentally. Specifically, an extended Luenberger observer (ELO) is used to account for the missing feedback parameters and the ELO-based MPC closed-loop performance is compared to the one of a PI control system.

7.2 Preliminaries

7.2.1 Nomenclature

Definitions of variables used in the modeling of the reactor:

- A_j: Pre-exponential factor for adsorption constant K_j of gas species j $[mol \cdot (m^2 \cdot s)^{-1}]$

- A_i: Pre-exponential factor for rate coefficient k_i for reaction i $[mol \cdot Pa^{0.5} \cdot (kg - cat \cdot s)^{-1}$ for i = 1 (SMR reaction), $mol \cdot Pa \cdot (kg - cat \cdot s)^{-1}$ for i = 2 (WGS reaction)]

- C_i: Concentration of species i $[mol \cdot m^{-3}]$

- C_{p_i}: Specific heat capacity of gas species i $[J \cdot (mol \cdot K)^{-1}]$

- F: Total molar flow of gases $[mol \cdot s^{-1}]$

- F_i: Total molar flow of gases of gas species i $[mol \cdot s^{-1}]$

- K_j: Adsorption constant of gas species j $[m^3 \cdot mol^{-1}]$

- k_i: Reaction rate constant of reaction j [$mol \cdot Pa^{0.5} \cdot (kg - cat \cdot s)^{-1}$ for i = 1 (SMR reaction), $mol \cdot Pa \cdot (kg - cat \cdot s)^{-1}$ for i = 2 (WGS reaction)]

- P_i: Partial pressure of gas species i [Pa]

- q: Outlet volumetric flowrate [$m^3 \cdot s^{-1}$]

- q_i: Outlet volumetric flowrate of gas species i [$m^3 \cdot s^{-1}$]

- r_i: Rate of reaction for reaction i [$mol \cdot (m^3 \cdot s)^{-1}$]

- R: Universal gas constant [$J \cdot (mol \cdot K)^{-1}$]

- $\bar{R}$: Cell resistance [Ω]

- T: Reactor temperature [K]

- T_{p_i}: Temperature of inlet gas species i [K]

- T_s: Temperature of the surroundings [K]

- UA: Overall heat transfer coefficient times the heat transfer area [$J \cdot (s \cdot K)^{-1}$]

- V: Reactor volume [m^3]

- Q: Power [W]

- W: Catalyst weight [kg]

- ΔH_r: Heat of reaction [$J \cdot mol^{-1}$]

- ΔH_{r_i}: Heat of reaction i [$J \cdot mol^{-1}$]

- $\dot{m}_{p_i}$: Mass flow rate of gas species i [$kg \cdot s^{-1}$]

- ρ_i: Density of the gas species i in the reactor [$kg \cdot m^{-3}$]

7.2.2 Experimental System and Digitalization

At UCLA, we have established an experimental steam methane reforming setup to investigate the efficiencies of various promising and novel hydrogen production methods. Transforming conventional-heating reformers into Joule-heating reformers is relatively straightforward, only requiring a modification in the heating source, and this work begins an investigation into the efficiencies, modeling, and control strategies of these electrically-heated experimental reforming systems. In order to minimize axial pressure drop, a Ni/ZrO$_2$ washcoat was deposited on the inner wall of the tubular reactor. The experimental and digitalization overview of the reactor are shown in Figure 7.1.
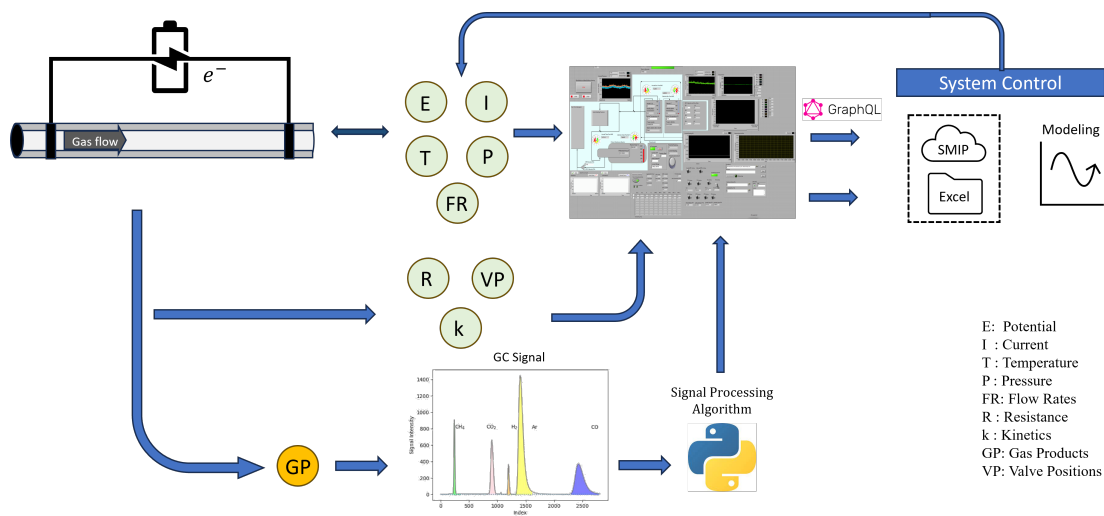
Figure 7.1: Process flow diagram for experimental Joule-heating steam methane reforming process.

The experimental setup involves monitoring various parameters such as temperature, pres-

sure, and current. Multiple sensors including thermocouples and pressure transducers are utilized for measurement. All sensors and actuators are connected to a Laboratory Virtual Instrument Engineering Workbench (LabVIEW) interface, a graphical coding platform that is very convenient for connecting data acquisition and actuation systems. LabVIEW has built-in drivers for various process equipment components, and many companies develop external drivers that can be incorporated into LabVIEW for digitalizing their own equipment. Three thermocouples (Omega K-type) are positioned throughout the system: one on the bottom of the exterior reactor wall (13.5 cm from the reactor outlet), the second one on the top of the exterior reactor wall (34.5 cm from the reactor outlet), and the third on the upstream gas flow pipe that is heated to 150 $^\circ C$ to prevent steam condensation. A single pressure transducer (Omega PX359 - 1KAI) measures reactor pressure and may be maintained at a constant value using a back pressure regulator (Equilibar). Thermocouples, back pressure regulators, and pressure transducers are digitized through a National Instruments Compact Rio, an industrial grade, reconfigurable data acquisition system. The experimental setup is shown in Figure 7.2.

In addition to the sensors and actuators in connection to the Compact Rio, a Chroma programmable DC power supply connects to LabVIEW as well. The reactor-power supply setup controls the current supplied to the reactor and measures the corresponding potential in a closed-loop configuration. The power supply is connected to the LabVIEW interface through external Chroma drivers. The energy given to the experimental setup is shown in Eq. 7.2a below, where Q is the rate of heat supply and the average resistance value ($\bar{R}$) can be found as the ratio of potential,
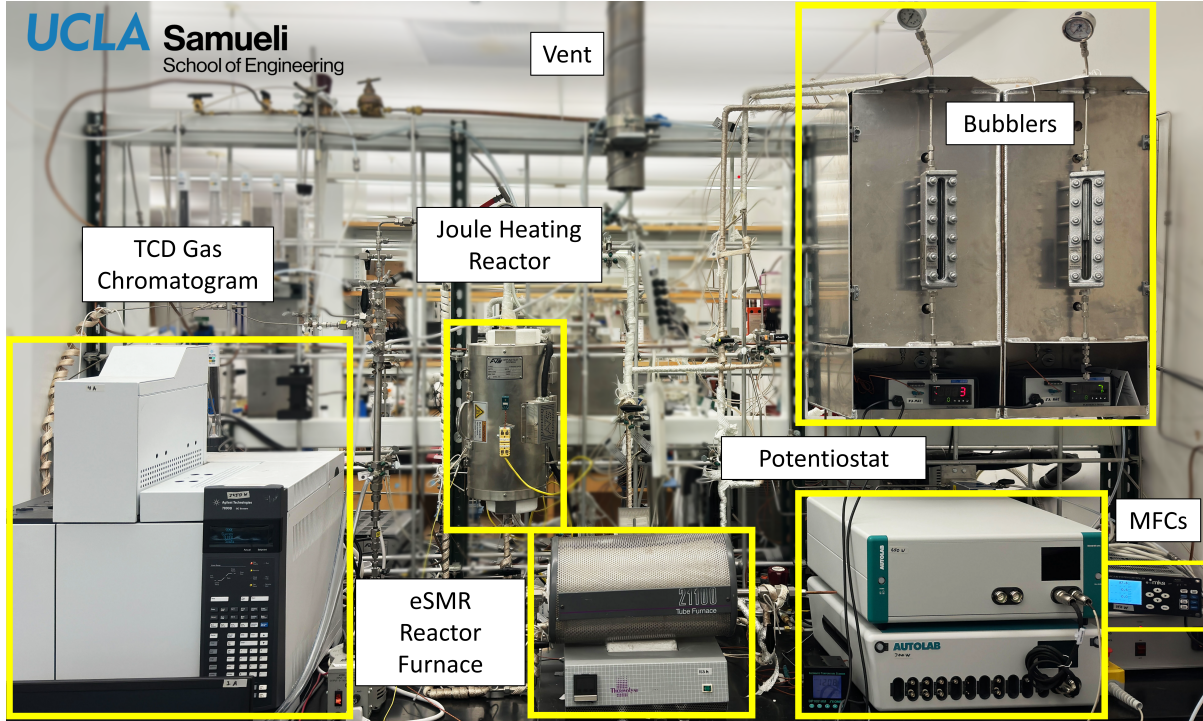
Figure 7.2: Picture of the experimental setup.

E, to current values, I using the entire time series data, as shown in Eq. 7.2b below:

$$Q = I^2 \bar{R} \tag{7.2a}$$

$$\bar{R} = \frac{E}{I} \tag{7.2b}$$

The tubular reactor is the central component of the Joule-heated SMR setup. Constructed with a 72.8% Fe, 22% Cr, 5% Al, 0.1% Y, and %0.1 Zr alloy procured from Goodfellow Corp., it features dimensions of 500 mm length, 6 mm outer diameter, and 5.4 mm inner diameter, as discussed in Chapter 6. The reactor tube is enveloped by fiberglass-based insulation to minimize heat losses, while an additional insulation furnace, equipped with ceramic foam material, provides further heat retention. In this study, the joule-heated system sits inside an Ascon Technologic R38S electric

furnace that can be used to provide heat similar to a conventional fired SMR process, but that is turned off in all the joule-heating experiments presented here, unless otherwise indicated.

Gas phase products are analyzed in real-time using an Agilent Technologies 7890B gas chromatograph equipped with a thermal conductivity detector (TCD). The gas mixture undergoes condensation to remove water vapor before entering the GC, where components including H_2, CO_2, CO, and unreacted CH_4 are quantified. Each analysis cycle lasts 15 minutes, followed by a 3-minute cool-down period. Consequently, the GC can process one gas sample every 18 minutes. Automated GC analysis is triggered by an external Python code, with results automatically processed post-measurement to determine peak areas and compare them to calibrated values for each gas species. The signal processing algorithm used is discussed in detail in Chapter 3. The gas chromatograph does not directly measure concentration or molar flowrate, rather it measures the molar percentage of the gases in the injection volume. This is based on the initial calibration of the GC, where various known molar ratios of gases were sent to the GC, and corresponding peak areas were recorded. After multiple gas mixtures were calibrated, the peak areas and corresponding percentages were fitted to power functions, which are used in real-time for quantifying the gas phase products. The sum of all percentages must be equal to 100% for an accurate gas quantification.

The inlet flowrates provided to the system are adjusted by a mass flow-meter (MKS). The flow-meter unit is standard cubic centimeters per minute ($sccm$). Trace amounts of Ar gas is flown through the system during each experiment to track molar flowrate changes in the outlet gas mixture. Argon is a noble, inert gas that does not react with other gases during the SMR process. As a result, the inlet molar Ar flowrate will be equal to outlet molar Ar flowrate, except for the very brief period of reactor startup.

The mass flow-meter is calibrated before each experiment. A specific flowrate input for each inlet gas (e.g., CH_4, H_2, Ar) that will be used for the experiment is entered on the flow-meter input panel and the corresponding GC injection output (in %) is recorded. First, the percentage of Ar that comes from the automated GC code is accepted to be the Ar basis. The sum of inlet gases (in $sccm$) shown on the mass flow controller is accepted to be the flow basis ($flow_{basis}$), and this is a constant value throughout the experiment for a constant inlet flowrate experiment. With each GC measurement, the Ar basis is divided by the Ar percentage in the injection volume to estimate the flow factor ($flow_{factor}$). In each GC measurement, the Ar peak area (and thus percentage) might change, however, since the molar flowrate of Ar will not change, Eq. 7.3 below can be used to estimate the molar flowrate of each species:

$$F_i[sccm] = \frac{GC_i[\%]}{100} \times flow_{basis}[sccm] \times flow_{factor}, \;\; i = CH_4, CO, H_2, CO_2, Ar \quad (7.3)$$

The flow factor is accounting for the change in the entire molar flowrate compared to the initial total flowrate by adjusting the molar fraction of the constant flowrate of Ar in the injection volume. The conversion for molar flowrate from mol/s to $sccm$ is shown in Eq. 7.4 below:

$$F\left[\frac{cm^3}{min}\right] = 1000 \left[\frac{cm^3}{L}\right] \times 22.4 \left[\frac{L}{mol}\right] \times F\left[\frac{mol}{s}\right] \times 60 \left[\frac{s}{min}\right] \quad (7.4)$$

The SMR system is equipped with two steam boxes that can supply steady inlet streams of water vapor to the electrically heated tubular reactor; however, only one unit is needed for the investigated setup. Each steam box houses a bubbler encased within fiberglass thermal insulation. Using a K-type thermocouple, an Arcon temperature actuator, and electrical heating tape, the bub-

273

bler is set to a desired steam-to-carbon (s/c) ratio via temperature control. The Antoine equation, shown in Eq. 7.5 below, estimates the vapor pressure of steam in the bubbler at a specific water temperature. The temperature sensor and Arcon actuator amount to PI control over the thermodynamic equilibrium of the inlet gas mixture and liquid water phase in the steam box. The Antoine equation is of the form,

$$\log_{10}(P) = A - \frac{B}{T + C} \tag{7.5}$$

where the temperature (T) is in $°C$, and pressure is in bar. The corresponding A, B, and C values are 8.14, 1810.9 $[°C]$, 244.5 $[°C]$ for temperatures above $100 °C$ [150].

All the sensors and actuators mentioned are connected to a LabVIEW interface shown in Figure 7.3. The LabVIEW interface gets real-time data from the experimental setup, such as pressure, temperatures, potential, current, standard volumetric flowrates, gas concentrations, and can send signals to experimental equipment to modify parameters such as current setpoint, the system back pressure, standard volumetric flowrates and steam box temperature setpoint. The Compact Rio is run through a Field Programmable Gate Array (FPGA) script written in LabVIEW and the signals read through a 32-bit LabVIEW script. A Compact Rio cannot run with 64-bit LabVIEW, and a 32-bit LabVIEW cannot run Python scripts through a Python node. As a result, we run a 32-bit script to acquire data from the FPGA script connected to the Compact Rio and record the data to a text file each second. Simultaneously, the main script reads the text file to transfer the sensor data so that a Python script embedded into the LabVIEW interface can process the data. The built-in PI control function is used for all the PI-based control demonstrations in this research. For a model predictive control scheme, a code written in Python is incorporated into the

LabVIEW interface.
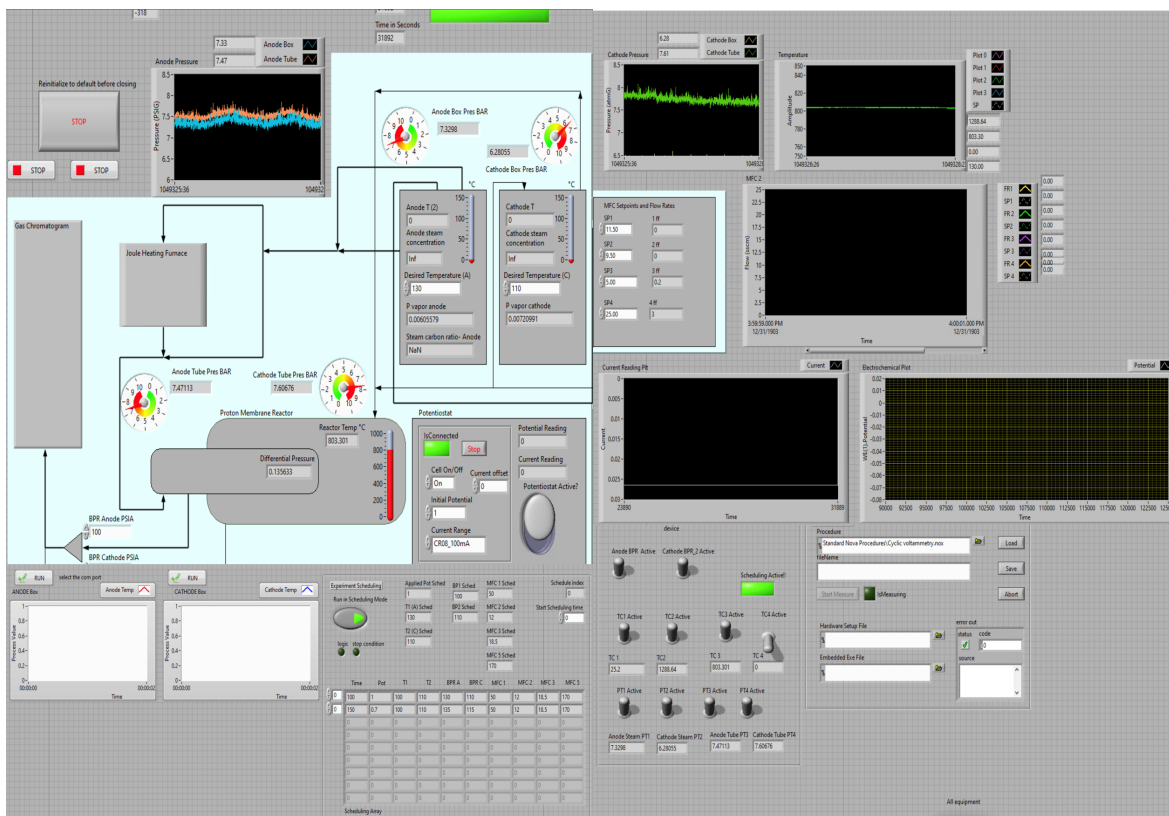
Figure 7.3: LabVIEW interface that connects sensors and actuators to the computer.

In Chapter 6, we have proposed a modeling approach and computationally demonstrated that this modeling approach can be used in a model predictive controller to operate the SMR system in an optimized way. In the present study, the modeling and control approaches are demonstrated to be experimentally effective. Still, there remains experimental challenges that prevent the application of the unmodified computational approach presented in Chapter 6. Firstly, the gas products coming out of the reactor need to be cooled down to be processed by the GC. Water must also be removed from the GC feed, since the peaks associated with water overlap with Ar and H_2 in the gas products signal. To this end, the outlet stream is exposed to a condenser that is cooled

with a cold water stream and brings the gas mixture temperature to room temperature. This means that the unreacted steam in the tubular reactor is condensed and is not quantified by the GC. Any first-principle based model would need the steam flowrate to initialize the model or correct the model with respect to sensor measurements. Thus, in an experimental implementation, we need to account for the missing steam flowrate. Another difficulty is that the volumetric flowrate is needed to convert molar flowrates into concentrations through Eq. 7.6 for reaction rate calculations. The volumetric flowrate can be measured by flow sensors, such as a bubble meter; however, the experimental flow-meters do not operate at temperatures higher than $100\,^\circ C$, and the outlet temperature of the SMR setup is expected to vary between 500 and $900\,^\circ C$.

$$C_i = \frac{F_i}{q} \tag{7.6}$$

A final challenge is that the GC measurements are delayed by 18 minutes (15 minutes for gas separation, elution, quantification, and peak processing, and an additional 3 minutes for cooling of the GC). This makes it difficult to correct the model in real time and requires a solution to incorporate past measurements into the real-time modeling scheme. All of the aforementioned complications required a creative methodology, discussed in the following sections, to successfully implement model predictive control.

7.3 Catalyst Synthesis

7.3.1 FeCrAl Tubing Pretreatment

All procedures for the Ni/ZrO$_2$ synthesis process and washcoat application were adopted from [148]. Prior to the application of the ZrO$_2$ support and Ni catalyst, a Goodfellow FeCrAlloy © tube was washed with acetone and rinsed with deionized water to remove any debris or other contaminants. The tube was heated to 950 $^\circ C$ in the furnace at a ramp rate of 1.4 $^\circ C/min$ where it was kept at the setpoint for 10 hours to calcinate the tube in air before being cooled to 30 $^\circ C$ at the previously mentioned ramp rate. During the heating process, slow dry air flowed through the system. The oxidation of FeCrAl reformer tubing is known to produce an adhesion-enhancing alumina layer on the inner and outer wall surface above 900 $^\circ C$ [62]. The color change caused by the oxidation process was evident on the outer wall of the tubing in Figure 7.4.

Figure 7.4: FeCrAl alloy tube after first oxidation procedure at 950 $^\circ C$ for 10 hours.

7.3.2 Washcoat Slurry Preparation and Application

With repect to the washcoat slurry preparation, 15 ml of deionized water (Milli-Q IQ7000, Milliporesigma) was added to a 50 mL beaker housing a magnetic stir bar. The pH of the water

was adjusted between 9-10 with ammonium hydroxide, (25 % NH_3, 99.99 % metal basis, liquid, Alfaesar) added dropwise. The pH-adjusted DI water was used to prepare a 21 wt % solution of zirconium (IV) oxide (powder, 5 um, 99 % trace metals basis, Sigma Aldrich). The magnetic stirrer (Fisherbrand^{TM} Ultra Thin Magnetic Stirrer) was set at 500 rpm and 4g of ZrO_2 powder was added to the beaker and allowed to stir for 30 minutes. After stirring, the slurry was sonicated in a Tuttnauer Clean and Simple Ulrasonic 60 kHz sonicator for 20 minutes.

The tube was removed from the furnace and mounted vertically using clamps. A tee valve was installed at the top of the tube and a drain nipple was secured at the tube's bottom. The slurry was then poured down the inside of the tube and allowed to drain. After the slurry application, a PTFE tube was secured to the top of the tee and air was flown with a syringe pump at a rate of 1 ml/min for an hour. The tube was then inserted into the furnace and heated to 100 $°C$ at a ramp rate of $1.4°C/min$. The tube was heated at this temperature for 1 hour, then the temperature was ramped up to 500 $°C$ (1.4 $°C/min$) and the tube was calcinated for 1 hour. The temperature was ramped down to 30 $°C$ at the same ramp rate. Five washcoats were prepared following the steps mentioned above. A final layer of the ZrO_2 support was added to the tube by leaving the slurry inside the tube for 10 hours before drainage, followed by air drying and calcination. The slurry setup and washcoat tubes are shown in Figures 7.5 and 7.6.
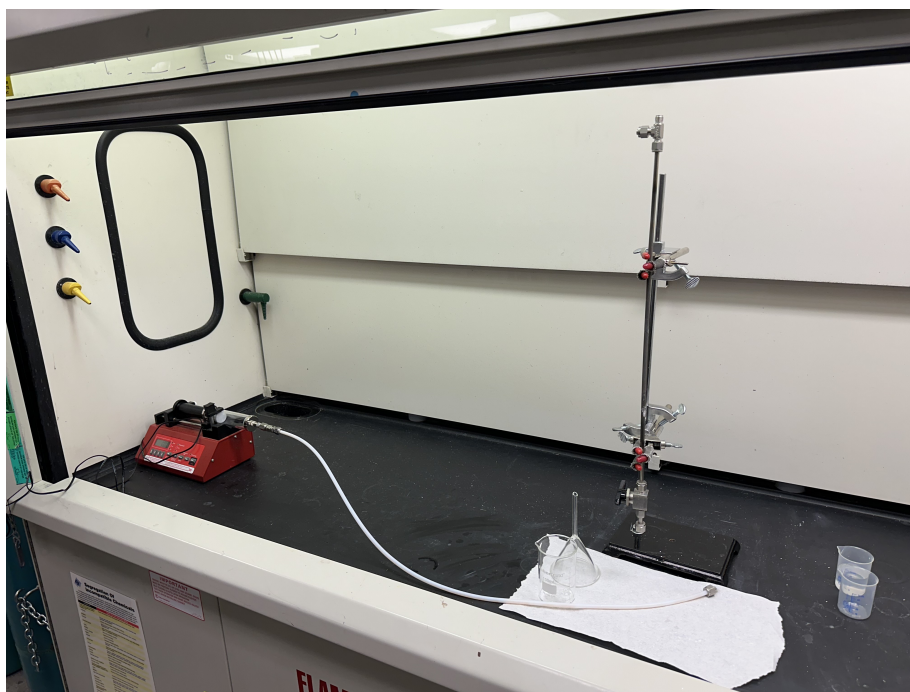
Figure 7.5: Slurry application setup.

Figure 7.6: Wet ZrO_2 washcoat monolith at the end of FeCrAlloy reformer tube prior to calcination and Ni catalyst embedment.

7.3.3 Catalyst Preparation

After calcination of the fifth ZrO_2 monolith, a 40 wt % nickel (II) nitrate hexahydrate (99.999% trace metals basis, Sigma Aldrich) was prepared and stirred in a 25 mL deionized aqueous solution for 5 minutes at 400 rpm. The solution was dropped into the vertical FeCrAl alloy tube via pipette and was allowed to sit for 1 minute before being drained, dropwise, into a 100 mL beaker. Once the first nickel monolith was applied, the tube was air-dried for 30 minutes before undergoing calcination at $110°C$ for 1 hour, then at 500 $°C$ for 1 hour (1.4 $°C/min$). A second nickel monolith was added to the reactor tube and remained in the tube for 12 hours. Following the draining of any residual nickel nitrate, calcination was performed once more with the previous procedure. The estimated loading of Ni on ZrO_2 is 50-80 mg.

7.3.4 Catalyst Reduction

A proper reduction technique was developed to initialize the catalytic activity of Ni/ZrO_2 before each experiment. For reforming, reduction of the Ni surface sites with high temperature H_2 reverts any metal oxides into their original metallic forms, thereby regenerating the catalyst. For this reason, the reduction procedure was performed 12 hours prior to every steady-state, PI-control, and MPC experiment.

An ATS cylindrical heating element, coupled with a PI temperature controller, served as the heating element for the reduction procedure. The first step of the procedure increased the temperature of the reactor tube to 110 $°C$ (at a rate of 5 $°C$/min). After 1 hour at $110°C$, the PI controller ramped up the reactor temperature to $850°C$ (at a rate of 5 $°C$/min) where it remained for 4 hours.

Subsequent cooling to $20°C$ occurred at the previous ramp rates. Constant flows of N_2 and H_2 (96 $sccm$ and 64 $sccm$, respectively) were maintained throughout the temperature schedule. After reduction, and prior to the start of any experiment, the catalyst was preserved with low-flowing N_2 and H_2 (30 $sccm$ and 17.7 $sccm$, respectively).

7.4   Feedback Control for Experimental Data Collection

The experimental SMR system requires a controlled increase of the temperature to prevent harming the catalyst. At the same time, it offers an opportunity for controlling gas phase concentrations with a PI controller, since the GC measurements are needed to generate data that can be used for model development.

7.4.1   Temperature Control for Data Collection

As discussed in Chapter 6, the temperature increase should be limited to $6°C/min$ to prevent catalyst degradation. If the reformer temperature rapidly increases or decreases, a change in the crystal structure of the catalytic monolith may occur by sintering and bring about changes to the Ni/ZrO_2 surface and bulk morphologies. The suggested constraint is around $6°C/min$ and the temperature ramping trend should be linear. Since the temperature and current relationship is not linear (explained in detail in Section 7.6), a constant increment in current would cause more than 6 $°C/min$ at the higher temperature range. Thus, PI control can manipulate the power supply current to increase/decrease the reformer temperature in gradual increments.

An additional constraint is used to limit the gain of the current-controller in order to generate

a linear increase in the reformer temperature when approaching the temperature setpoints. If the current-controller is too aggressive, a 6 $^\circ C/min$ temperature setpoint increase may result in a significant initial jump in current, and 60-80% of the final setpoint value is achieved in the first few seconds of controller action. In this case, the controller continues to slowly decrease the current over the next few minutes to remain at the temperature setpoint. Even though the average ramp rate may be within the $^\circ C/min$ constraint, the release of large amounts of electrons during the initial time steps of controller action leads to a sharp increase in the temperature derivative, violating the temperature rate of change constraint, and possibly inhibiting Ni surface sites by promoting sintering processes. Thus, to keep the temperature derivative constant at 6 $^\circ C/min$, the current should not change more than 2 A in one time-step. As it is explained in Section 7.6.2, 1 A increase in the current can lead to an increase of up to 39 $^\circ C$ (if a final steady-state is reached). However, the way that the controller works is that it first causes a sharp increase in the current within 1 to 2 A, after which the current gradually decreases to preserve a linear change in temperature. This behavior is illustrated in Figure 7.7.

7.4.2   Steady-state Data Collection

Following the tuning of a controller to linearly increase the reactor temperature, experimental data at specified temperatures were collected to determine the SMR reaction kinetics and to quantify radial heat transfer. A range of thermal conditions were designated as steady-state setpoints. The temperature was slowly increased to those temperatures and multiple GC injections were made at a steady-state temperature, and subsequently, the PI controller drove the process to the new temperature target. To limit the deactivation of Ni/ZrO$_2$ catalyst, the PI controller was used
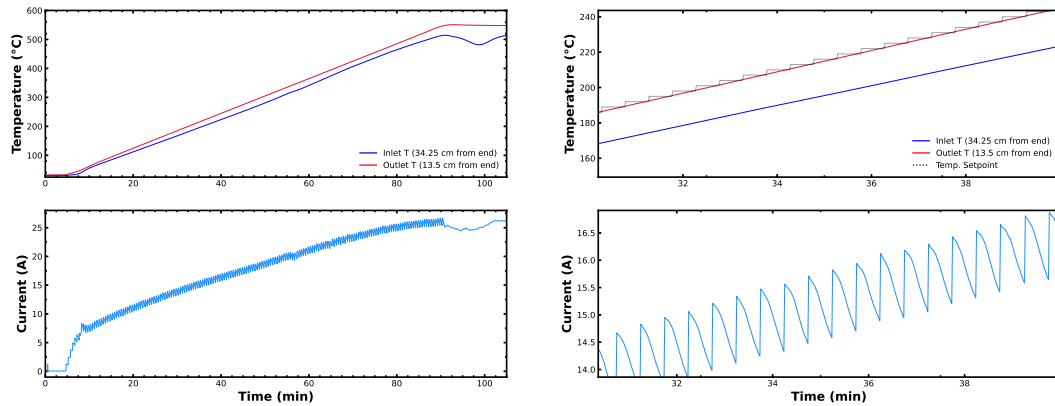
(a) Start-up phase reactor temperature increase.

(b) Zoomed in setpoint change for linear temperature increase.

Figure 7.7: Linear temperature increase under control and corresponding current manipulation.

to drive the outlet temperature of the reactor from 550 to 800 $°C$ in increments of 50 $°C$ gradually over a certain time window. The corresponding current ranges between 23 A to 34 A depending on the heat transfer coefficient. However, Figure 7.8 demonstrates that, after the controller drives the reactor temperature to a new setpoint temperature, the current is gradually decreased to keep the temperature at the setpoint. This would suggest that the system is not at a thermal equilibrium as long as the current keeps decreasing. The main reason for this is the fiberglass insulation that surrounds the tubular reactor. The heat conduction from the surface of the reactor to the fiberglass is very slow since fiberglass has a very low heat conduction coefficient. As a result, the increase on the fiberglass surface temperature is not as fast as the temperature increase on the reactor outer wall. When the reactor wall is driven to a steady-state temperature, the insulation is at a lower temperature due to slow heat conduction. As the controller keeps the reactor wall temperature constant, the fiberglass slowly continues increasing in temperature, resulting in lower heat loses as time progresses. Hence, the controller reduces the current input to the reactor to keep it at a steady

283

temperature. This is an indication that the heat transfer coefficient for the heat lost to the surroundings is not constant. Also, a large temperature difference in excess of $50\,^{\circ}C$ was seen between the reformer's inlet and outlet after the start of the endothermic SMR reactions that consume heat at the inlet section.
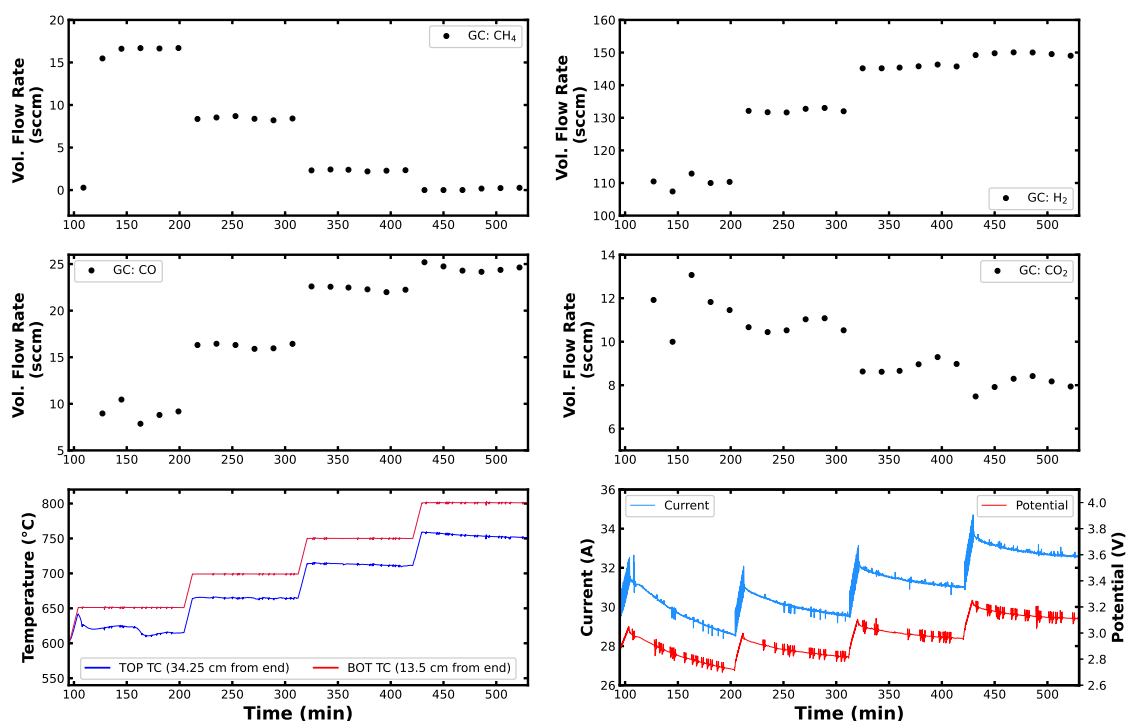
Figure 7.8: Steady-state data collection with a Joule-heated energy source for experimental parameter calculations. Temperatures range from $650\,^{\circ}C$ to $800\,^{\circ}C$ for the outlet thermocouple. Each steady-state condition was maintained for 110 minutes.

A feed mixture of CH_4, H_2O, H_2, and Ar (39.47/119.5/17.7/6.47 $sccm$) was sent to the reactor at 1 atm for all steady-state and dynamic control experiments. Each steady-state temperature was maintained for 110 minutes, giving ample time for kinetic equilibrium to occur and for the minimization of thermal gradients in the reactor insulation. The high temperature data collection

experiment is shown in Figure 7.8. Six flowrate samples from the GC were averaged at each temperature to determine the steady-state effluent flowrates. When the outlet temperature of the reformer tube was $500\,^{\circ}C$, the conversion of methane was 7%, compared to 99% at $800\,^{\circ}C$. Above $650\,^{\circ}C$, the reverse WGS reaction is favored, leading to greater CO selectivity over CO_2. Figure 7.9 confirms this activity and the maximum selectivity towards CO_2 production occurs at $650\,^{\circ}C$ when the ratio of CO_2:CO is 1.31 (Figure 7.10b). The average hydrogen production rate at 800 $^{\circ}C$ was 149.6 $sccm$, and the average absolute errors between the computational model of Chapter 6 and laboratory results in Figure 7.9 were 1.59, 3.43, 1.45, and 2.47 $sccm$ for CH_4, H_2, CO, and CO_2, respectively. Most of the variability in H_2 measurements occurred at the $650\,^{\circ}C$ and 700 $^{\circ}C$ steady-states. The high error is attributed to increasing catalyst deactivation coupled with axial and radial thermal gradients introduced by catalytic inhibition. Experimental errors in the GC total mixture percentage also peaked around the $650\,^{\circ}C$ and $700\,^{\circ}C$ steady-states, leading to additional variability in the measurements of all gas species at these temperatures. The GC total mixture percentage should equal 100% for a perfectly calibrated system. In practice, it is found that the total mixture percentage is somewhere between $101-105\%$.

The thermodynamic efficiencies for steady-state conversion of CH_4 at 650, 700, 750, and 800 $^{\circ}C$ outlet temperatures are given in Table 7.1. CH_4 is initially an energy carrier and the efficiency of transforming CH_4 into H_2 is an essential metric for the overall SMR process. Heats of reaction for steam methane reforming were calculated using the heat capacities (C_P) for all reactants and products of the first SMR reaction. Similarly, the enthalpy required to heat the inlet gases to the average reformer temperature was calculated by integrating their respective C_P values over the temperature differential between the laboratory conditions and the inside of the reformer. The

285
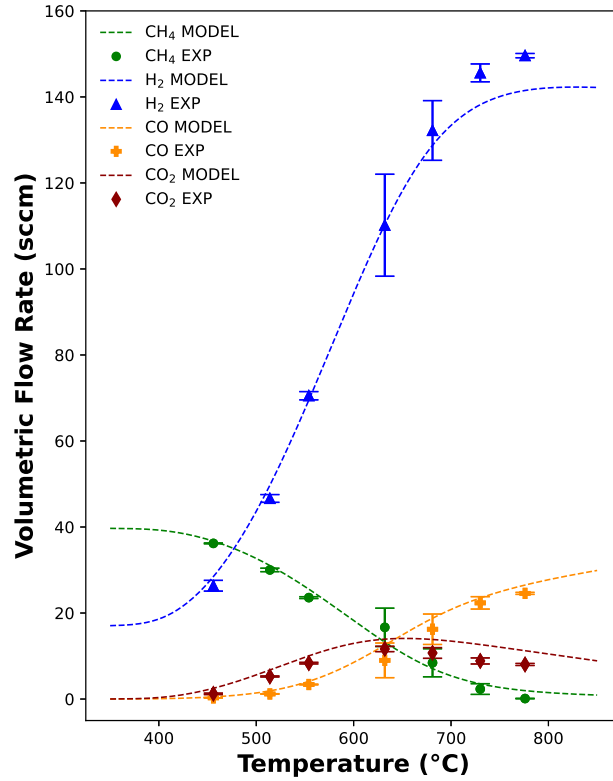
Figure 7.9: Steady-state experimental data and model predictions over the 400 $^{\circ}C$ to 800 $^{\circ}C$ temperature range. For the experimental dataset, average steady-state temperatures are reported. Error bars represent the standard deviations of volumetric flowrates.

efficiency calculation is given by the following equation:

$$e_{ff} = \frac{\Delta H_{\text{T, SMR}} \times \left(CH_{4, \text{ Molar flowrate In}} - CH_{4, \text{ Molar flowrate Out}}\right) + \Delta H_{\text{Inlet Gasses}}}{Average\ Power\ Input} \times 100\% \quad (7.7)$$

where the thermodynamic efficiency of CH_4 conversion is equal to the ratio of the heat of reaction for the first SMR reaction at a given steady-state temperature times the molar flowrate of converted CH_4 plus the energy requirement to heat the inlet gasses to the steady-state temperature over the average power input. An optimal thermodynamic efficiency of 10.69% was achieved for the elec-

Table 7.1: Thermodynamic reaction efficiency for steady-state CH_4 conversion rates.

	Average Reformer Temperature ($°C$)			
	632	**681**	**730**	**776**
$\Delta H_{T, SMR}$ (kJ/mol)	232.46	235.5	239.01	242.81
CH_4 Conversion Energy Requirement ($Watts$)	4.79	5.98	6.82	7.09
ΔH for Inlet Gases ($Watts$)	2.74	2.96	3.21	3.43
Average Power Input ($Watts$)	84.51	86.58	93.81	103.39
CH_4 Thermodynamic Reaction Efficiency (%)	**8.91**	**10.33**	**10.69**	**10.18**

trified SMR system at the 730 $°C$ steady-state condition. This corresponds to an outlet temperature of 750 $°C$, an average CH_4 conversion of 94.1%, and a CO_2:CO ratio of 0.396. The 681 and 776 $°C$ steady-states generated efficiencies over 10% as well, signifying the relative optimality of the 700 to 800 $°C$ outlet temperature range.

The experimental electrified SMR system includes an ATS split-tube electric furnace. To confirm the proper workings of the Joule-heated reformer, the electric furnace was used as a standard for comparison. The heating profile of the electric furnace is axially parabolic, with the center of the furnace holding the peak temperature set by the Watlow PI controller using an Omega K-type heavy-duty thermocouple for temperature sensing. As a result, the heat profile was at a minimum at the inlet and outlet of the reformer. Except for the ATS specification limits of 1100 W, 115 V, and 9.6 A, the real-time power output of the electric furnace was not known during steady-state data collection. In spite of this, it was expected that the steady-state CH_4 conversion and CO_2:CO selectivity should be comparable to the Joule-heated energy source given that the average temperatures of the two heating elements were within 20 $°C$. The CH_4 conversion for both heating elements was 100% at 800 $°C$ and 94% at 750 $°C$, indicating identical catalytic performance at the upper operational limits of the reformer. At 650 and 700 $°C$, the electric furnace converted 5% and 3%

more CH_4 than the Joule-heated furnace which was within the range of experimental error (Figure 7.10a). Thus, the Joule-heated reformer adequately supplied energy to the outer reactor shell in the form of heat. Additionally, the CO_2:CO selectivity for the heating elements was comparable for all steady-state measurements (Figure 7.10b), except for 650 $^\circ C$ which produced selectivities of 1.35 and 1.04 for the Joule-heating system and the electric furnace system, respectively. A difference in shape of the heating profiles of the heating elements may have caused the deviation in CO_2 products between the two experiments. It is also possible that catalyst deactivation may have influenced the SMR reaction rates during the electric furnace experiment.
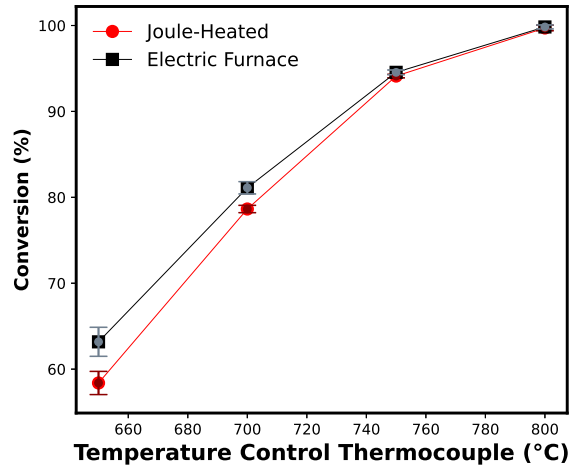
(a) Average steady-state conversion when the temperature ranges from $650\,^{\circ}C$ to $800\,^{\circ}C$.
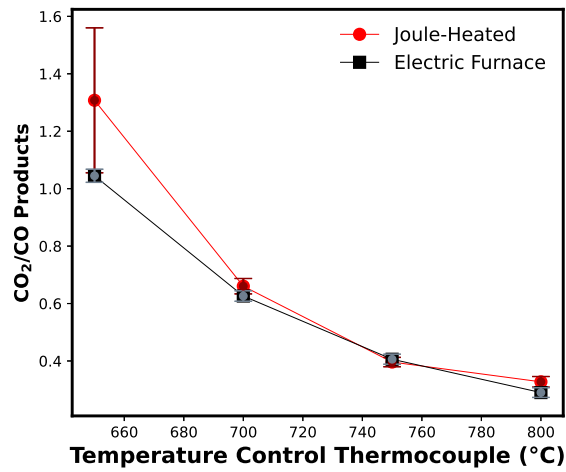
(b) CO_2/CO product selectivity over the same temperature range.

Figure 7.10: Steady-state data collection from two energy sources: Joule-heated setup and electric furnace setup.

7.5   Modeling

7.5.1   Lumped Parameter Model

Real-time MPC of the SMR system requires a model that can be solved sufficiently fast in real-time on the order of seconds. SMR reactions occur in a tubular flow reactor, which would require solving partial differential equations (PDEs) to account for spatio-temporal evolution of variables such as species concentrations and temperature. Solving PDEs require commercial CFD software (e.g., Ansys Fluent, Comsol, etc.) and the solution time may be on the order of hours to days depending on the model detail. Thus, building an MPC based on a CFD model is not appropriate for real-time control purposes. In order to efficiently solve the mass and energy balance equations in real-time, the flow reactor is approximated as a continuously stirred tank reactor (CSTR) modeled by a lumped parameter ordinary differential equation (ODE) system, which is much faster to solve with methods like Runge-Kutta or Explicit/Implicit Euler. This brings a trade-off between the accuracy of the ODE model solution and numerical simulation speed. A lumped parameter approach sacrifices some accuracy in the resulting model as it does not account for spatial variations of the process state variables but it enables real-time solution calculations with a sufficient accuracy. Below, the lumped parameter model equations are given; the derivation of the lumped model can be found in Chapter 6.

The reaction mechanisms of SMR on Nickel-based catalyst were investigated by [215], from

which the reaction rate equations were as follows:

$$r_{1,SMR} = \frac{k_1}{P_{H_2}^{2.5}} \cdot \frac{P_{CH_4} \cdot P_{H_2O} - \frac{P_{H_2}^3 \cdot P_{CO}}{K_1}}{(DEN)^2} \tag{7.8a}$$

$$r_{2,WGS} = \frac{k_2}{P_{H_2}} \cdot \frac{P_{CO} \cdot P_{H_2O} - \frac{P_{H_2} \cdot P_{CO_2}}{K_2}}{(DEN)^2} \tag{7.8b}$$

$$DEN = 1 + K_{CO} \cdot P_{CO} + K_{H_2} \cdot P_{H_2} + K_{CH_4} \cdot P_{CH_4} + K_{H_2O} \cdot \frac{P_{H_2O}}{P_{H_2}} \tag{7.8c}$$

$$k_j = A_j \cdot \exp\left(-\frac{E_j}{R \cdot T}\right), \quad j = 1, 2 \tag{7.8d}$$

$$K_i = A_i \cdot \exp\left(-\frac{\Delta H_i}{R \cdot T}\right), \quad i = CH_4, H_2O, CO, H_2 \tag{7.8e}$$

and the mass balances for our electrified SMR system are as follows:

$$\frac{dC_{CH_4}}{dt} = \frac{1}{V_R} \left(F_{CH_4,0} - r_1 \cdot W - q \cdot C_{CH_4}\right) \tag{7.9a}$$

$$\frac{dC_{H_2O}}{dt} = \frac{1}{V_R} \left(F_{H_2O,0} - (r_1 + r_2) \cdot W - q \cdot C_{H_2O}\right) \tag{7.9b}$$

$$\frac{dC_{CO}}{dt} = \frac{1}{V_R} \left((r_1 - r_2) \cdot W - q \cdot C_{CO}\right) \tag{7.9c}$$

$$\frac{dC_{H_2}}{dt} = \frac{1}{V_R} \left(F_{H_2,0} + (3 \cdot r_1 + r_2) \cdot W - q \cdot C_{H_2}\right) \tag{7.9d}$$

$$\frac{dC_{CO_2}}{dt} = \frac{1}{V_R} \left(r_2 \cdot W - q \cdot C_{CO_2}\right) \tag{7.9e}$$

$$\frac{dC_{Ar}}{dt} = \frac{1}{V_R} \left(F_{Ar,0} - q \cdot C_{Ar}\right) \tag{7.9f}$$

The solution of the above equations require to express the molar flowrate (F_i) in terms of concentration (C_i) and volumetric flowrate (q). Specifically, the volumetric flowrate will vary based on temperature as in Eq. 7.10 below:

$$q = \frac{F_{T0} + 2 \cdot r_1 \cdot W}{\frac{P}{RT}} + \frac{V_R}{T} \cdot \frac{dT}{dt} \tag{7.10}$$

7.5.2 Model Initialization

Molar and volumetric flowrates for steam are not measured via measurement sensors. Still, using the available flow equations, the following equations can be solved simultaneously with a good initial guess for the steam molar flowrate:

$$0 = q - \frac{FRT}{P} \tag{7.11a}$$

$$0 = q - \frac{F_{T0} + 2 \cdot r_1 \cdot W}{\frac{P}{RT}} - \frac{V_R}{T} \cdot \frac{\mathrm{d}T}{\mathrm{d}t} \tag{7.11b}$$

where F is the total flow rate, Eq. 7.11a is the ideal gas law and Eq. 7.11b is the expression of the volumetric flowrate. The first reaction rate (r_1) is a function of gas species partial pressures, including steam, which is shown in Eq. 7.8a. Eq. 7.12 below can be utilized to calculate the partial pressure of each species required in the rate equation:

$$P_i = \frac{F_i}{F_T} P \tag{7.12}$$

where F_i and P_i are molar flowrate and partial pressure of each gas species, respectively. These equations can be used to initialize the ODE solver of the process dynamic model at a steady-state. The molar flowrates are obtained from the GC-based calculations, while temperature values are measured by the two thermocouples. Once the system reaches steady-state, the total volumetric flowrate (q) and the steam flowrate (F_{H_2O}) are calculated, and the ODE solver is initialized to

calculate the rest of the process variables.

7.5.3 Parameter Estimation using Experimental Data

Our previous computational study in Chapter 6 suggested an approach for using a lumped parameter model in a model predictive controller that does not use the thermodynamic energy balance for temperature estimation, rather it uses a data-driven first-order dynamical model that relates the applied current to the reactor temperature. However, due to the nonlinear nature of heat transfer in steam methane reforming, first-order linear dynamic models can only capture the reactor temperature behavior within the region where the linear model is valid. To create a more accurate model over the entire operational temperature range, a comprehensive energy balance is used in the present work. Using a lumped parameter assumption, the energy balance takes the following form:

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{I^2\bar{R} + \sum_i \dot{m}_{p_i} C_{p_i}(T_{p_i} - T) - Wr_{SMR}\Delta H_{SMR}(T) - Wr_{WGS}\Delta H_{WGS}(T) + UA(T_s - T)}{\sum_i \rho_i C_{p_i} V}$$

$$(7.13)$$

where the $I^2\bar{R}$ term represents the heat given to the system by the power supply and can be read from the sensors. The reaction enthalpies at specific temperatures can be calculated mathematically using the Shomate equation [164] which accounts for the heat capacity and formation enthalpies at standard state (298 K, 1 atm). The reactor is cooled by the surrounding ambient temperature, which is accounted for in the $UA(T_s - T)$ term. Thus, the only unknown in the energy balance is the UA term, which is the overall heat transfer coefficient times the heat transfer area. In a perfectly insulated system, the U term would be zero and the system would not lose any heat to

the surroundings. In that case, there must be a cooling stream, such as a jacket, around the reactor to reduce the reactor temperature when necessary. Furthermore, experimental systems are not ideal systems, and despite the use of an insulation layer in our reactor, there are heat losses to the surroundings. The UA term can be estimated by fitting experimental temperature data to the mass and energy balances, Eq. 7.9 and Eq. 7.13, respectively. To fit the data, steady-state operating data, such as the experimental data shown in Figure 7.8, can be used directly without de-noising. Figure 7.8 illustrates the conditions for four steady-states at 650 $^\circ C$, 700 $^\circ C$, 750 $^\circ C$, and 800 $^\circ C$. Thus, this data is fitted to nonlinear mass and energy steady-state balances to estimate the UA value [29]. As a result of the data fitting procedure explained in [29], the UA term is estimated to be 0.116 $J \cdot (s \cdot K)^{-1}$. Finally, in our previous work [29], we demonstrated the fitting of the experimental data to the reaction kinetics equation described by [215] to estimate the pre-exponential factor and activation energy, and the results showed very close correspondence.

7.5.4 Model Evaluation

The model predictive controller will use the lumped model described in Section 7.5.1. This model assumes that the mixing is perfect and the temperature and concentration profiles are uniform everywhere in the reactor. Note, in a tubular reactor, the reactions occur in the axial direction and neither the temperature nor the concentrations are uniform. The gas mixture starts reacting in the inlet section of the reactor where the highest heat consumption occurs due to the endothermic SMR reaction. This causes a significant temperature drop at the inlet section of the reactor which is evident when analyzing thermocouple values over the axial length of the reactor. The maximum temperature difference over this length is around 100 $^\circ C$. This brings about the question of which

temperature to use in the lumped model. One effective approach is taking a weighted average of top and bottom thermocouples to be the lumped model temperature.

To find meaningful weights for averaging the reactor temperature, available data from flowrate feedback control were used. In this instance, various weights were used to find the best matching outlet flowrate and temperature prediction. Since the heat transfer coefficient term (UA) was known, providing the experimentally recorded current values to the energy balance equation (Eq. 7.13) was sufficient to estimate the reactor temperature. The estimated temperature was subsequently compared to the weighted average of experimentally recorded inlet and outlet thermocouple values.

To compare the model predictions with recorded thermocouple temperatures and GC concentrations, the numerical ODE solver that integrates the mass and energy balances (Eq. 7.9, Eq. 7.13) is provided with an initial condition. However, the gas chromatogram provides only molar flowrates of CH_4, H_2, CO, CO_2, and Ar. Steam concentration is not measured in the reactor effluent, as steam is condensed before the GC feed. As a consequence, initializing the ODE solver requires estimates for the outlet steam molar and volumetric flowrates. The two equations shown in Eq. 7.11 were solved simultaneously for a given steady-state to get an estimate of outlet steam molar and volumetric flowrates. Once F_{H_2O} and q values are estimated, the solver can be initialized.

In each experiment, multiple GC injections are taken at each steady-state (5-6 injections per steady-state temperature). If the experiment performs flowrate control, the controller is initialized after multiple GC injections at the outlet section temperature of 550 $^\circ C$ steady-state. To find the top and bottom thermocouple weights, the model is initialized at the steady-states and inte-

grated with respect to the recorded experimental current values. After trying many combinations of weights for the top and bottom thermocouples, the optimal weights are estimated to be 60% of the top thermocouple value and 40% of the bottom thermocouple value. This also corresponds to the vicinity of the experimental steady-state H_2 flowrate value at the weighted temperature. The overall temperature of the reactor for model calculations is based on these weights. Since the majority of CH_4 conversion occurs at the inlet section of the reactor, giving more weight to the top thermocouple provides a better representation of the reactor heat profile when compared to taking the arithmetic mean of the two thermocouple values. The model predictions with the respective weights are shown in Figure 7.9, for one of the feedback control experiments later explained in Section 7.6.1. This model takes only recorded experimental current as an input, calculates temperature values, gives continuous predictions of the gas specie concentrations. The predictions are compared to discrete GC measurements in Figure 7.11.

Remark 38 *The UA term must be calculated with respect to the weighted average temperature. Thus, each weight tested with the experimental data must calculate a unique UA value. The data fitting of UA term shown in Section 7.5.2 is calculated for the weights obtained in this section.*

Carbon Formation Effect

In any SMR system, a common disturbance process is the formation of carbon (coke) on the catalyst throughout the reactor. A carbon layer is formed on the catalyst and blocks available surface sites, decreasing the catalytic performance [111]. The side reactions that take place in Eq.

296
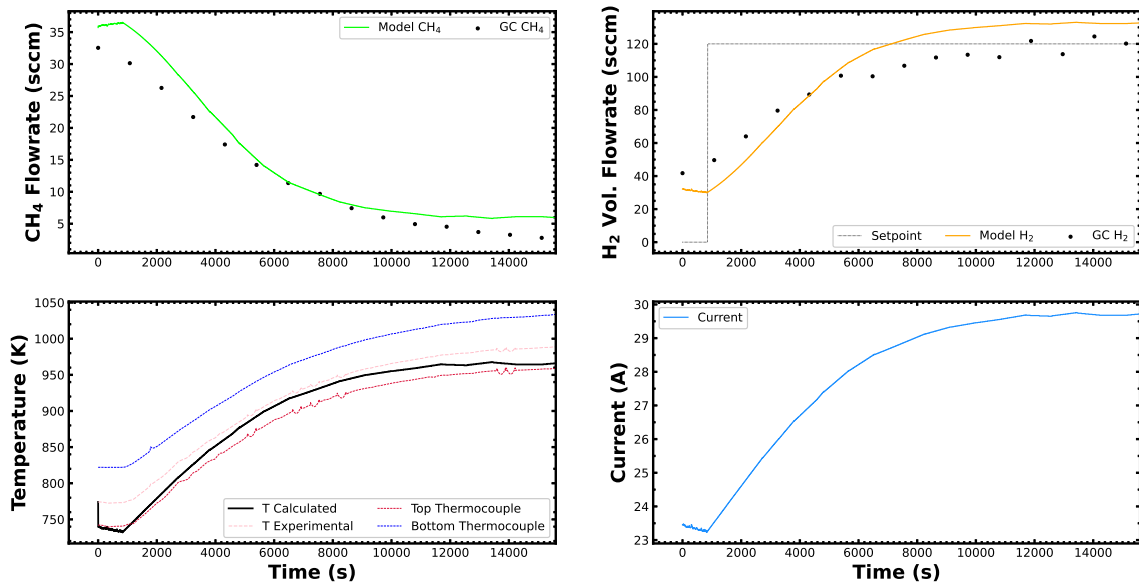
Figure 7.11: Model predictions for outlet CH_4 and H_2 molar flowrates when experimental current is provided.

7.14 cause coke formation.

$$CH_4 \rightleftharpoons 2H_2 + C, \quad \Delta H_{298} = 75 \ kJ \cdot mol^{-1} \tag{7.14a}$$

$$2CO \rightleftharpoons CO_2 + C, \quad \Delta H_{298} = -172 \ kJ \cdot mol^{-1} \tag{7.14b}$$
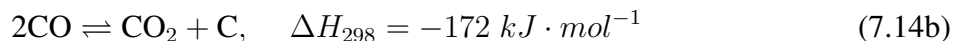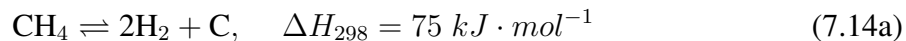
This initially formed carbon, C_α, is very active and some carbon atoms are transformed to C_β. For C_β, atoms may be vaporized and the remaining portion can diffuse within the catalyst. This diffusion can make the carbon nucleate and precipitate at the back of the catalyst, causing the catalyst to lift, triggering fragmentation of nickel crystallite [220]. Coke formation is therefore detrimental to the catalyst and hard to avoid. When the carbon atoms are counted in the inlet stream (from CH_4) and outlet stream (from CO, CO_2, and unreacted CH_4). Figure 7.12 demonstrates the

difference for the steady-state data collection experiment in Figure 7.8.
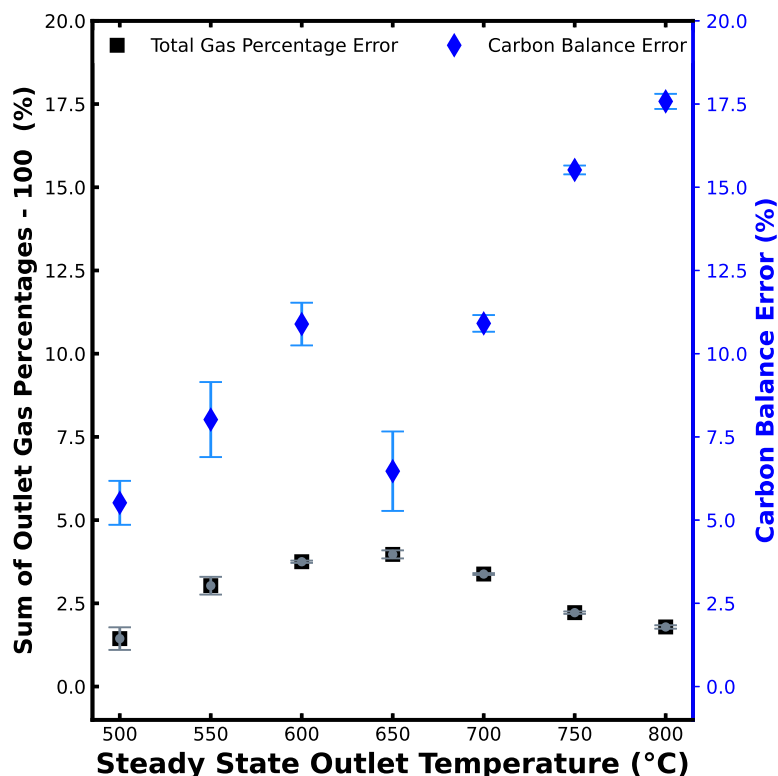
Figure 7.12: Average carbon balance and GC errors for the steady-state data collection experiments shown in Figure 7.11.

The carbon deposition effect can be reduced using methods like the gasification of the carbon layer to refresh the catalyst. As a part of the SMR system experimental procedure, H_2 and steam are used to gasify the carbon layer on the reactor catalyst for 5 minutes at the beginning and end of each experiment. However, coke formation may have caused deviations in modeling results due to carbon losses in the carbon mass balance for experimental data. Also, this phenomenon is very challenging to quantitatively model, and furthermore, it is difficult to eliminate coke while the SMR process is already running. [48] and [6] worked towards measuring the rate of carbon formation and proposed the rate equations shown in Eq. 7.14, which were developed under certain

298

assumptions. For example, [48] assumed the decomposition of CO (Eq. 7.14b) to carbon was dominant. However, both our steady-state and dynamic data do not support this assumption. If the coke formation is mainly caused by CO according to Eq. 7.14b, CO_2 flowrate estimation from the model should be smaller. However, CO_2 has larger estimated flowrate values when compared to the experimental data shown in Fig 7.13, implying that this assumption may not be valid for our process.
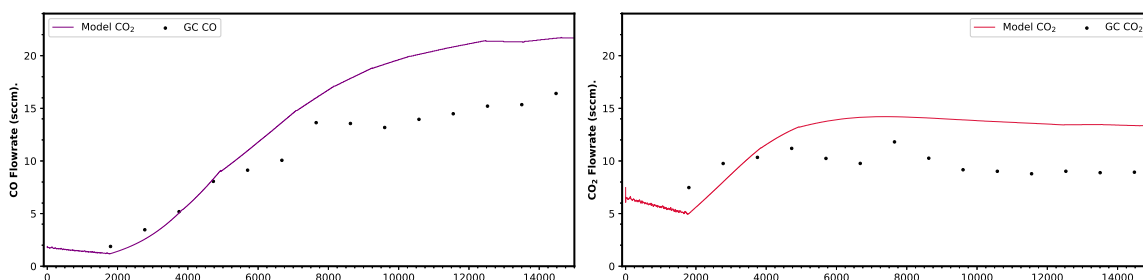
Figure 7.13: Model predictions for CO and CO_2 when experimental current is provided. The CO_2 flowrate estimations demonstrate some deviation from the experimental results, which might be due to coke formation at higher temperatures.

7.6 Feedback Control of Hydrogen Molar Flowrate

7.6.1 Experimental PI Control

In order to evaluate the efficiency of a model predictive controller in terms of speed of response, a control study with only sensor feedback is first conducted. In this control scheme, the controlled variable is the outlet H_2 flowrate and the manipulated variable is the current. As the SMR is an overall endothermic process, the equilibrium constant will increase as temperature increases, and the process will produce more H_2. The temperature controller drives the process from room temperature to the first steady-state at 550 $^\circ C$ in the bottom temperature, which corresponds

to 500 $^\circ C$ in terms of the weighted average temperature. Then, the H_2 flowrate controller takes over and drives the process to the 120 $sccm$ setpoint decided in [29]. Theoretically, it requires 102.3 $sccm$ of H_2 produced from the reaction (given that 17.7 $sccm$ of H_2 is in the inlet flow rate to the reactor). According to the stoichiometric ratio of CH_4 and H_2 in Eq. 7.8, the minimum CH_4 consumption is 25.575 $sccm$ if 4 moles of H_2 is produced from 1 mole of CH_4 (64.8 % conversion), and maximum is 34.1 (86.4 % conversion) $sccm$ if 3 moles of H_2 is produced from 1 mole of CH_4 (excluding WGS reaction), respectively. However, it is important to note that this theoretical conversion range is calculated without considering the coke formation.

The inlet flowrates of CH_4, H_2O, H_2, and Ar remained unchanged from the steady-state experiments to maintain a gas hourly space velocity (GHSV) of 1000. Finally, after the setpoint is reached and GC injections are taken for at least 324 minutes, the temperature controller takes over the process once more upon re-initialization in order to decrease the reactor temperature to ambient conditions. The GC data points are delayed by 15 minutes and updated every 18 minutes. Thus, the state seen by the controller is constant at the GC output for 18 minutes. However, the integral term in the controller keeps integrating the error, causing a continuous increase in the controller's current output being sent to the power supply on a per second basis. The controller was tuned in [29] using a computational process model. The final parameters found from this simulation were $K = 0.0012$ $A/sccm$, and $\tau_I = 79$ s and have been adopted by the concentration PI controller. The results from the first PI control experiment are shown in Figure 7.14. The control behavior is very consistent with the proposed control strategy in [29]. The controller drives the process to its 120 $sccm$ setpoint without violating the current ramp rate and keeps the process at said setpoint with only slight fluctuations around the target hydrogen production value. The H_2 flowrate control
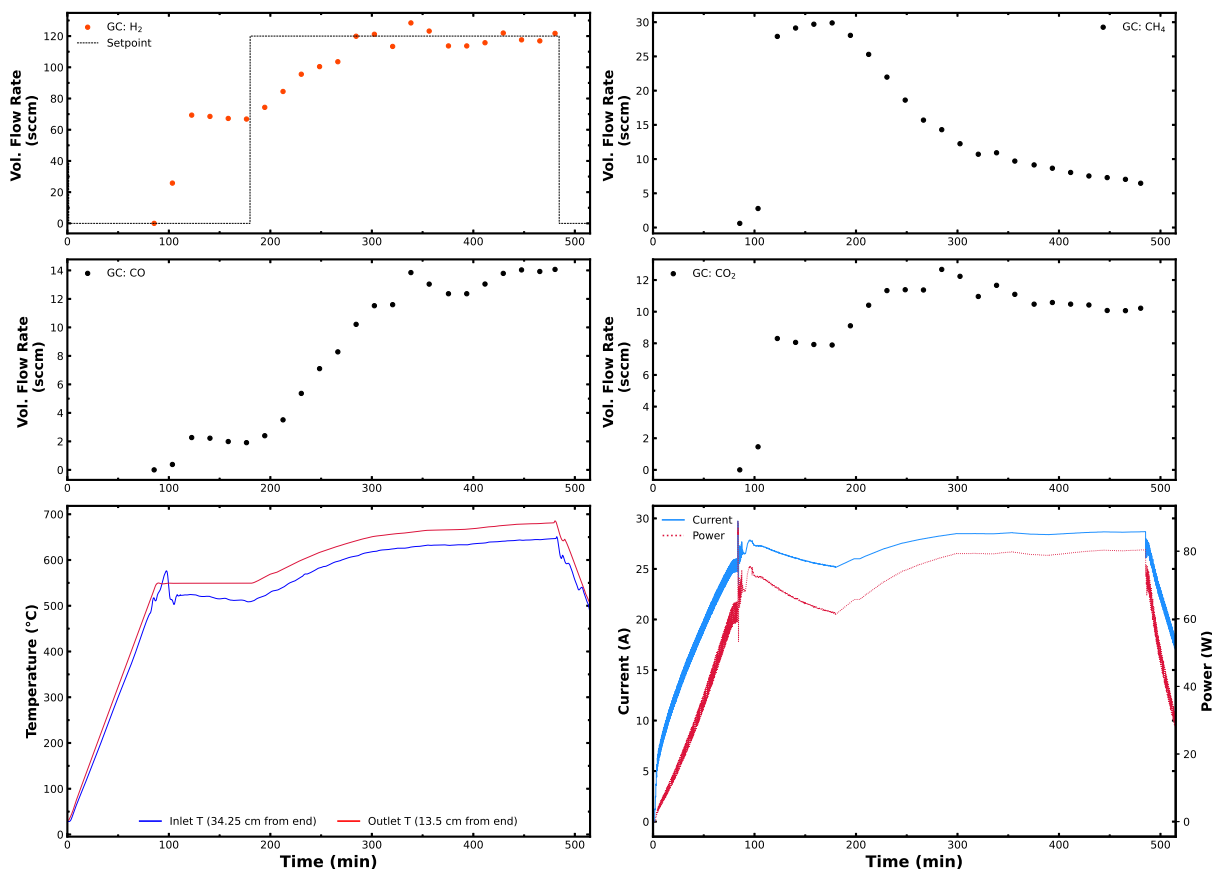
Figure 7.14: First experiment under PI control based on GC feedback only.

experiment is repeated three times to show that this PI control scheme can drive the process at the desired setpoint. The second experiment (Figure 7.15) confirms the ability of the PI controller to drive the process to the desired setpoint.

Figure 7.17 demonstrates the error between the three PI control experiments and the closed-loop model under PI control prediction. This is an indication that the model would perform fairly well. The slight dynamic mismatch between 0 and 200 minutes might be due to the lumped parameter modeling. However, the mismatch at steady conditions between 300 and 400 minutes after the control starts might also indicate some other phenomena, especially for the third PI experiment.

While interpreting the result, it is crucial to note that the reverse water gas shift reaction is
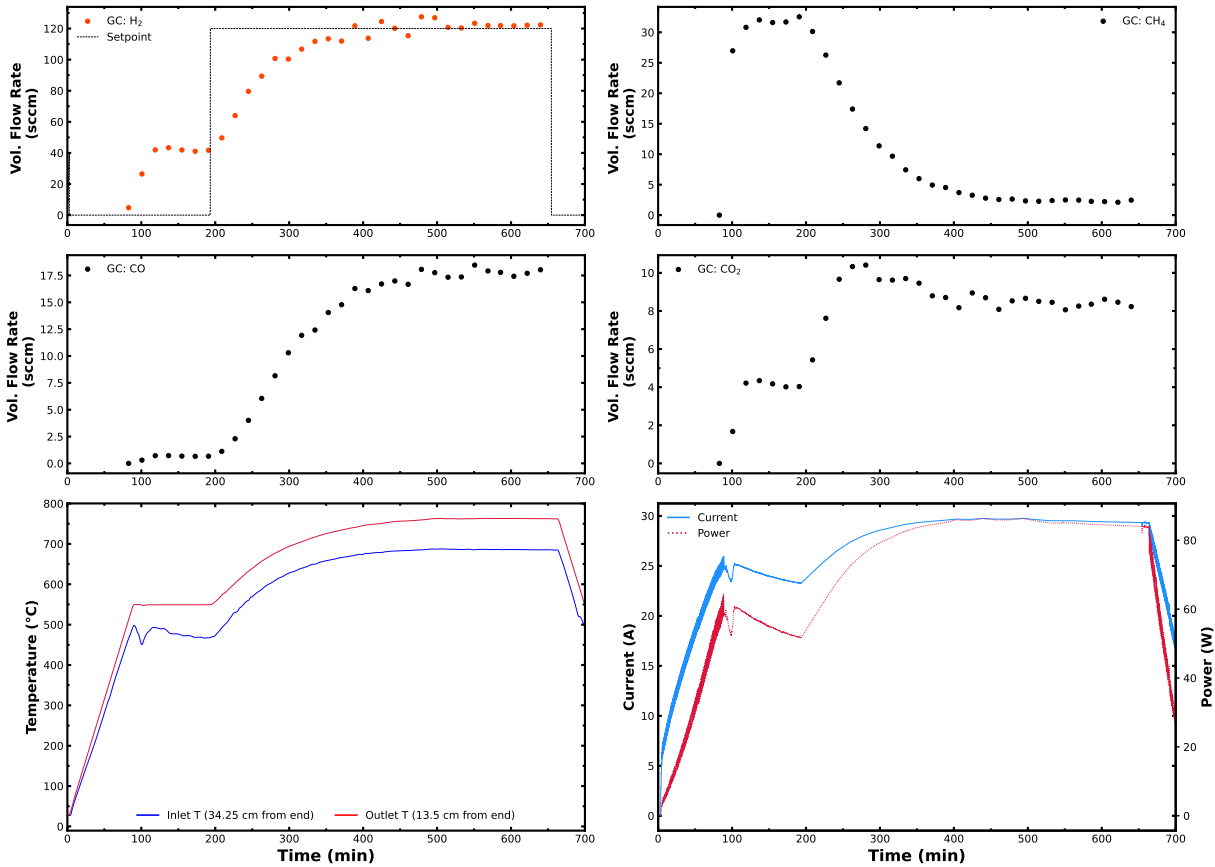
Figure 7.15: Second experiment under PI control based on GC feedback only.

favored around the setpoint. WGS reaction is slightly exothermic, and as the temperature increases the equilibrium shifts toward producing more CO and less CO_2 and H_2. Thus, at this point, the H_2 production is mainly maintained by the reforming reactions. At lower temperatures, 1 mole of methane produces roughly 4 moles of H_2, and as the reverse WGS takes over, 1 mole of methane produces 3 moles of H_2. However, as the temperature increases, the reforming reactions become faster and still produce more H_2 compared to lower temperatures. This phenomenon can be seen in Figure 7.9. Around 650 $^{\circ}C$, the CO_2 flowrate is decreasing while the CO flowrate is increasing. Also, after 650 $^{\circ}C$, the slope of the H_2 flowrate production decreases as temperature increases. Thus, in the PI experiments, the setpoint of 120 $sccm$ is usually at a transition area for WGS and
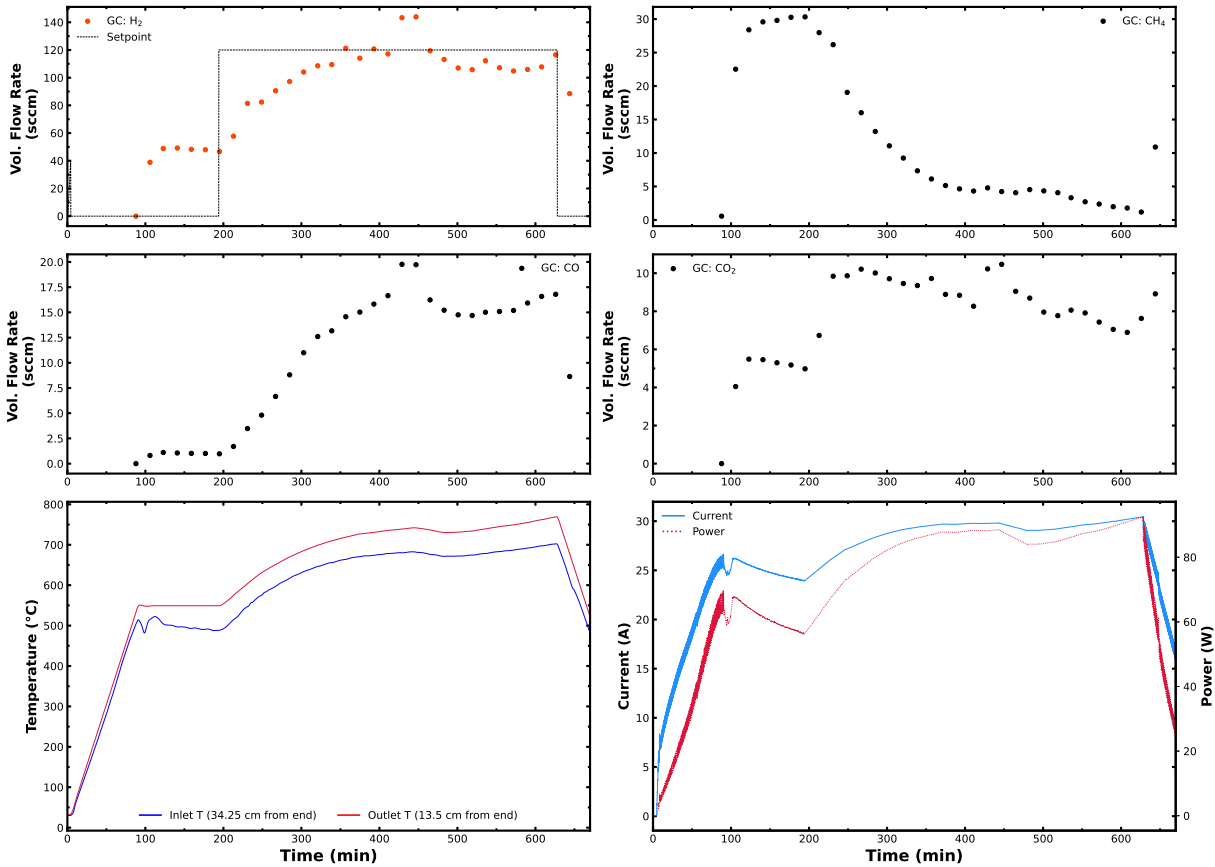
Figure 7.16: Third experiment under PI control based on GC feedback only.

reverse WGS. Hence, after this point, the controller increases the reactor temperature to produce more H_2, converting more methane.

It is also very important to keep in mind that the scatter plots for each gas specie demonstrates when the GC measurement was taken and not when this measurement value was received by the controller. Thus, the controller keeps driving the process based on the previous GC result, and the integral term integrates the previous error. If the GC measurement is very close to the setpoint, the controller tends to keep constant. However, as mentioned in Section 7.4.2, the constant current increases the temperature due to delayed heating of the fiberglass. Thus, while the controller thinks that the process is at steady-state, the gradual heating of the fiberglass is causing a temperature
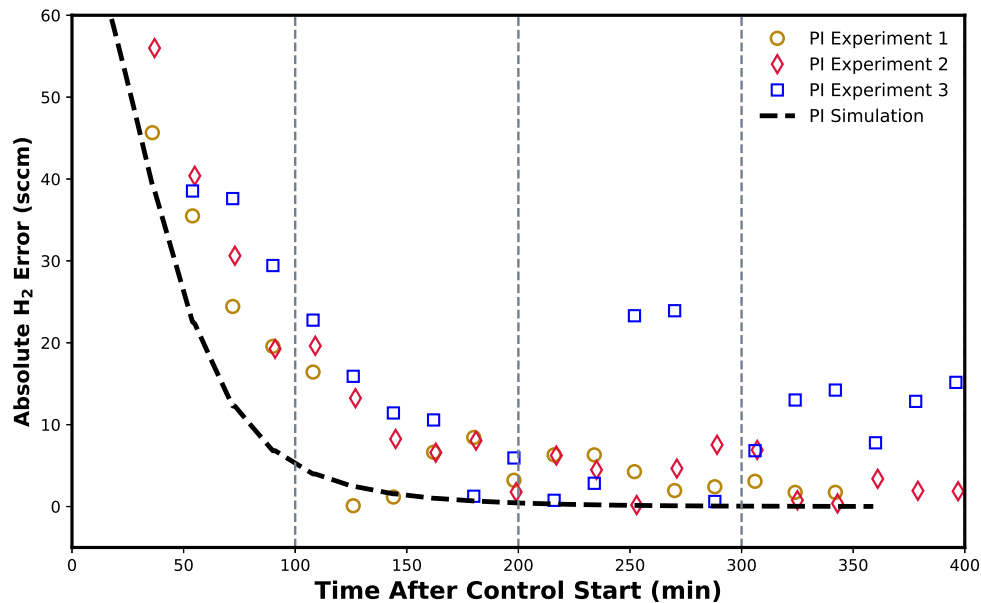
Figure 7.17: Absolute error of H_2 ($sccm$) production with respect to simulated closed-loop response under PI control.

increase in the reactor, thus a surge in the H_2 concentration, even though the current seems to be constant. Moreover, the controller can start decreasing the current only after the GC measurement is obtained, meaning that the current was used for further increase of the H_2 concentration when it should have decreased. One clear example of this is the H_2 flowrate increase in Fig 7.16 at 420th and 450th minutes. To compensate for this increase, the controller quickly decreases the current, and this delayed control brings the process to a state before the reverse water gas shift reaction is favored. The further increase of the current triggers the reverse water gas shift and slows down the H_2 increase. This can be seen from the CO_2 concentration trend change at 265th minute, where the reverse WGS is triggered for the first time, and the 420th minute when the WGS reaction trend changes one more time.

The third experiment (Figure 7.16) shows more variation as quantified in Table 7.2 in H_2 mo-

304

Table 7.2: Avg. absolute errors for hydrogen production (sccm) under PI control for different time intervals (min).

Time (min)	Exp. 1		Exp. 2		Exp. 3	
	Avg. Err. (sccm)	St. Dev.	Avg. Err. (sccm)	St. Dev.	Avg. Err. (sccm)	St. Dev.
0-99	31.29	11.67	36.57	15.57	41.96	14.14
100-199	5.99	6.02	9.59	6.14	11.31	7.52
200-299	4.25	2.07	4.62	2.78	10.29	12.19
300-399	2.18	0.79	2.54	2.39	11.64	3.48

lar flowrate around the setpoint. When CH_4 conversion is checked in Figure 7.18, the H_2 flowrate is below the setpoint, and yet, methane conversion approaches 100%. At this point, side reactions such as carbon formation (which is explained in Section 7.5.4) or catalyst deactivation might be happening in addition to reverse WGS shift reaction becoming more dominant. However, the PI controller still boosts the current to bring the H_2 concentration to the target value. These phenomena are not as severe in the first two PI control experiments, mainly because of the experimental conditions. The lumped parameter steady-state model suggests that the process settles at 120 $sccm$ H_2 production rate around 650 $°C$ (temperature weighted average). However, depending on the experimental variability of parameters (e.g., catalyst activity, coke formation, etc.), the steady-state temperature and reverse WGS triggering varies. The first PI control settles at the steady-state around 630 $°C$, and these temperatures are 715 $°C$ and 695 $°C$ for the second and third PI control experiments, respectively.

The difference between the inlet carbon flowrate (coming from CH_4) and the outlet GC measurements' carbon flowrate (coming from CO, CO_2, and unreacted CH_4) is shown in Figure 7.19 with respect to time and temperature. Overall, an increase in reactor temperature causes a greater consumption of methane, and increases coke formation. At similar high temperatures, the third PI control experiment produced slightly higher amounts of carbon, which might indicate that the
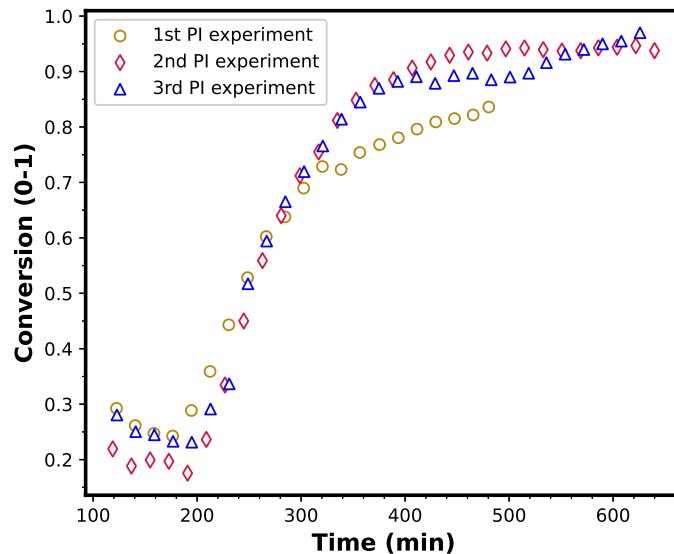
Figure 7.18: CH$_4$ conversion for the 3 PI control experiments.

catalyst was more deactivated compared to the first two experiments. Instead of producing more H$_2$, the catalyst favors more carbon formation. Hence, a higher conversion does not mean higher H$_2$ production, since methane might be consumed for coke production.

The temperature increment is targeted to be kept under 6 $^\circ C$ per minute to preserve the catalyst activity. Thus, the PI controller was tuned to keep the catalyst under this constraint at all times. Figure 7.20 displays the change in temperature each minute during the control experiments. The temperature change each minute is around 2.5 $^\circ C$. A more aggressive controller gain could have been used for the PI for a faster response. However, even with the current parameters, the PI controller causes an oscillation in each experiment. Thus, a higher K_c value would lead to higher-amplitude oscillations which is undesired. Furthermore, these oscillations would become more severe as the deactivation and coke formation effects become more significant.

The PI control experiments demonstrate that a PI controller with delayed measurements re-
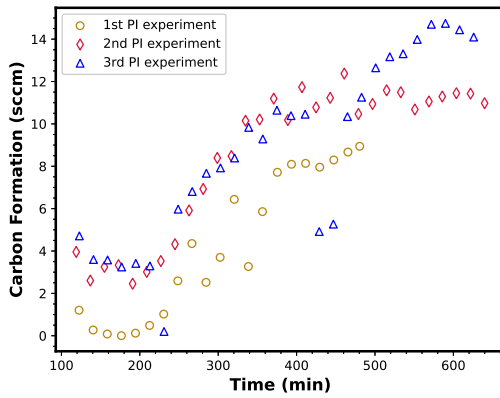
(a) Coke formation with respect to temperature.
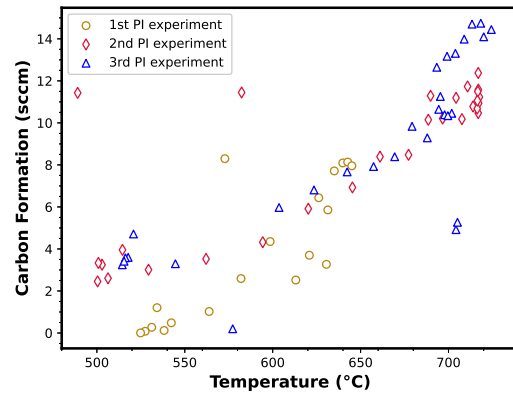
(b) Coke formation with respect to temperature.

Figure 7.19: Difference between the inlet and outlet carbon atom flowrates.
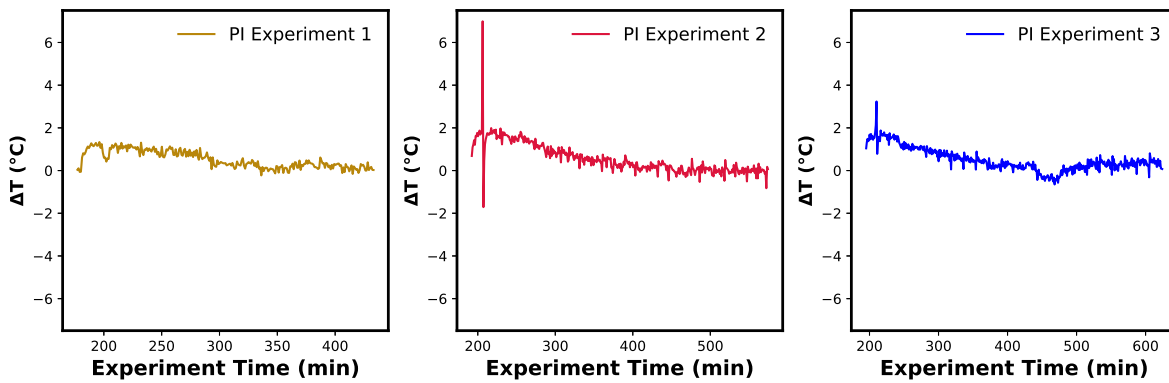
Figure 7.20: Change in the reactor outlet temperature over time for all PI experiments.

quires between 120 to 200 minutes to drive the process to the setpoint. Section 7.6 below focuses

on the improvements on the response time using an estimation-based model predictive controller.

7.6.2   Model Predictive Control

The reaction kinetics are very fast for SMR reactions. The computational work done by [29] demonstrates that the process is expected to settle at a steady-state within 1 second at 1000 GHSV. Thus, even for a slight temperature change, the time required for reaching a steady-state is very short. However, the allowed current increase rate prevents controllers from going to higher target temperatures to reach the desired production rates faster than the ramp rate limits. PI controllers might not provide the most optimal current input to the power supply at each time step. Instead, an MPC would ensure that the most optimal input value is calculated at each time step without violating the constraints. Thus, we examine the behavior of an MPC to make sure that this process is driven to the desired H_2 production rate in the most efficient way possible. The model estimation and optimization are computationally inexpensive, such that all the calculations can be made on LabVIEW in one second. However, the LabVIEW script runs many calculations in one loop. Consequently, if a model solution calculation takes more than one second, there are delays on the code execution. To prevent these time delays in computing the control actions, the MPC is designed to make one calculation every 5 seconds.

The most important constraint given to the MPC is the 6 $°C/min$ temperature change rate. This constraint is therefore embedded into the MPC optimization problem. To do this, the behavior of the temperature against current was examined experimentally. Using the process model explained in the previous section, a current against expected temperature graph is generated and fitted to a 2nd order polynomial. The resulting polynomial is $T(I) = 0.7I^2 + 2.64I + 290.7$ with an R^2 value of 0.99, and the data fit is shown in Figure 7.21a. The MPC operates between 700-1100

K, and thus, when zoomed into this operational region, it is possible to check if a linear constraint can be given. The data points in this region were fit to a linear function via regression, and the resulting function was found to be $T(I) = 39.4I - 186.2$, which is illustrated in Figure 7.21b. The data points show close alignment with the linear model. Thus, within this operation region, it is safe to assume that a 1 A increase in current causes a 39.4 $^\circ C$ increase in temperature. Since the MPC is designed to make a calculation every 5 seconds, the maximum allowed increase in current corresponds to 0.13 A to satisfy the 6 $^\circ C/min$ temperature rate of change constraint used in the MPC optimization problem.
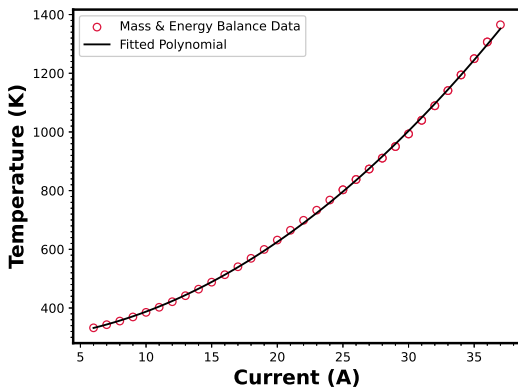
(a) Polynomial fitting of mass and energy balance generated data points.
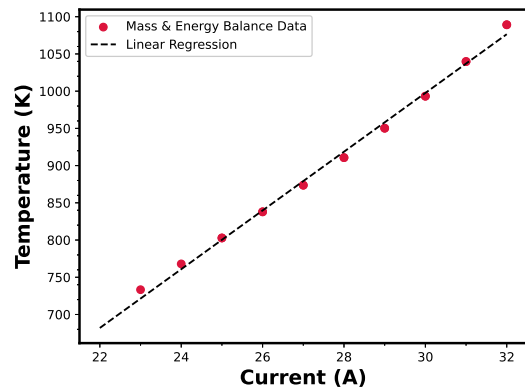
(b) Linear behavior of current temperature behavior between 700-1100 k.

Figure 7.21: Current and temperature relation fitted for MPC current constraints.

Extended Luenberger Observer

In the MPC architecture, the model predictions are updated with feedback obtained from real-time process measurement data. However, the measurement sensors do not yield volumetric and steam molar flowrates. In this type of a feedback control problem, methods like Kalman Filter [78] or Luenberger observer [104] help to combine the process model and sensor feedback values to

obtain an estimate of the state of a system. Specifically, an extended Luenberger observer can be developed for a dynamic system of the form:

$$\dot{x} = f(x, u) \tag{7.15a}$$

$$y = h(x) \tag{7.15b}$$

where x is the state vector, u is the control input vector, $f(x, u)$ is a nonlinear vector function, y is the measurement vector, and $h(x)$ is the transformation vector function that transfers the state value to the measurable physical information. Specifically, the extended Luenberger observer has the following form:

$$\dot{\hat{x}} = f(\hat{x}, u) + K_{ELO}(y - h(\hat{x})) \tag{7.16}$$

where $\hat{x}$ is the estimated state vector, y is the measured output, and K_{ELO} is the observer gain matrix designed to ensure the convergence of the estimated states $\hat{x}$ to the true states x.

In the implementation of the ELO to the experimental setup, the model predictions are corrected by adding the difference between estimated values and sensor values, all multiplied by a tuned gain value. In our work, tests on model predictions were performed to see if the model's predictions could be updated using only temperature values (sampled each second) and H_2 flowrate values from a GC analysis (sampled every 18 minutes). In this way, the missing values are estimated from the model and the correction terms bring all the estimations closer to their real values. The ELO state estimates are subsequently sent to the MPC to optimize the current input for the next sampling time (5 seconds). It is also important to note that the process model in MPC is not

corrected by an estimation error term as future measurements are not available; rather, the ELO is used to calculate the process state variables needed to initialize the process model used in the MPC. The detailed derivation of the equations for ELO using the process model and the observer gain matrix is presented in [29]. The experimentally tuned gain matrix (K_{ELO}) used in this work is as follows:

$$K_{ELO} = \begin{bmatrix} -0.01 & 0.05 & -0.1 & 0.7 & 0.01 & 0.04 & 100 \\ 0.8 & -0.60 & 0.2 & 2 & 0.005 & 0 & 0 \end{bmatrix} \tag{7.17}$$

where the first row corresponds to the correction tuning values for the difference in temperature and its effect on each of the species and the temperature. Each column shows the tuned values that influence CH_4, H_2O, CO, H_2, CO, Ar and temperature. The difference in H_2 does not affect the temperature ODE, since the first row of K_{ELO} mainly accounts for the temperature correction. The second row represents the correction that is implemented for the difference between H_2 production estimation and GC value that corresponds to 15 minutes prior. This 15-minute delay causes jumps in the ELO predictions. Also, since the H_2 values correspond to those from 15 minutes before, the ELO gain matrix should not have large gain to correct for the gas species at the current time step. Finally, since the H_2 concentration is controlled and the only sensor feedback for H_2 is obtained from the GC, the ELO output for H_2 is tuned to be very close to the last GC measurement. Using

the ELO, the MPC optimization problem implemented on our experimental setup is of the form:

$$\mathcal{J} = \min_{I} \int_{t_k}^{t_{k+N_h}} L(\bar{x}(t), I(t)) \, \mathrm{d}t \tag{7.18a}$$

$$\text{s.t.} \quad \dot{\bar{x}}(t) = ODE(\bar{x}(t), I(t)), \ \bar{x}(t_k) = \hat{x}(t_k) \tag{7.18b}$$

$$\bar{F}_{H_2}(t) = h(\bar{x}(t)) \tag{7.18c}$$

$$L(\bar{x}(t), I(t)) = A(\bar{F}_{H_2}(t) - F_{H_2,sp})^2 + B(I(t) - I_{sp})^2 \tag{7.18d}$$

$$t \in [t_k, t_{k+2}) \tag{7.18e}$$

$$|I(t_k) - I(t_{k-1})| \leq 0.013 \tag{7.18f}$$

$$0 \ A < I < 40 \ A \tag{7.18g}$$

where the allowed current change is limited to $0.013 \ A$ per 5 seconds, the current range is bounded between 0 and $40 \ A$, $N_h = 2$ is the prediction horizon length, $h(\bar{x}(t))$ is the transformation needed to calculate the hydrogen outlet flow rate from the dynamic model states, the process model is the ODE solver and the optimized cost function is the weighted summation of quadratic errors between the H_2 molar flowrate and the current state estimated by the model, and the current value and the steady-state current estimation at the desired setpoint. The A and B values in Eq. 7.18d are taken to be 1 and 0.01, respectively. The constraint on the rate of change of the current is imposed so that the temperature in the reactor does not change so fast that the catalyst activity is compromised.

Remark 39 *The sensor feedback for other gas outlet flowrates could be incorporated in the ELO to improve the accuracy of the predictions. However, this would require tuning of more parameters in the gain matrix and was not needed in the present experimental implementation due to the*

achieved closed-loop performance.

Adjustments to Experimental Conditions

We calculated in the beginning of Sec. 7.6.2 that the maximum amount of current increase is 0.013 A from the energy balance equation. However, the experimental setup is vulnerable to phenomena that can cause discrepancies between the model and experimental outputs, such as carbon formation, catalyst deactivation, change in catalyst activity, and delayed heating of insulation layer. Thus, in the experimental setup, the same H_2 production rate can be reached at slightly different temperatures. As a result, if the process control system relies entirely on the model, it may end up not reaching the setpoint experimentally even though the model calculates that the process is at steady-state. In order to handle this type of situation, an integral term was added to the MPC output to help the controller drive the process output to the setpoint. Considering that the integral term will further increase the MPC output current, the maximum allowed current increment in the MPC formulation was limited to a lesser amount than 0.013. In this case, the current input at each time step was calculated using the following equation:

$$I(\hat{t}) = I_{initial} + \sum_{t=t_0}^{\hat{t}} \Delta I_{MPC} + \frac{1}{\tau_I} \int_{t_0}^{\hat{t}} e(t) \, dt \qquad (7.19)$$

where $\hat{t}$ is the current time step, $I_{initial}$ is the current recorded when the MPC starts, I_{MPC} is the current calculated by the MPC only, and $e(t)$ is the error term between the setpoint and the ELO H_2 output. In this case, a τ_I value is selected, and the maximum amount of the current increment provided by the MPC is limited to 0.008 A, which would modify the constraint in Eq. 7.18f. The

integral error term is integrated with respect to the ELO output at each time step. The τ_I value is chosen to be 434086 s, which is very high, to make sure that the impact of the integral term is small. The main reason for this is that the ELO is tuned such that the H_2 estimation is very close to the previous GC measurement. Hence, until the GC is updated, the H_2 prediction does not increase, causing a significant accumulation of error. Also, it is not desirable for the integral term to take a huge step and violate the current increase constraint. Finally, it is important to point out that an alternative to the addition of an integral term in the MPC control action would be to adopt an offset-free MPC scheme. In the present case, the impact of the integral term is really small (owing to the choice of the gain parameter of this term) and only has an effect as the process output approaches the steady-state; please see the experimental results in the next subsection.

In Section 7.4.2, it was discussed that the insulation layer undergoes a delayed increase in temperature change. This implies that the heat from the reactor wall is conducted through the insulation layer, and then heat is dissipated to the surroundings at 25 $^\circ C$ through convective heat transfer. In the energy balance equation (Eq. 7.13), the heat loss is represented with the $UA(T_s-T)$ term, which only accounts for the convection, and is a linear expression. Since the nonlinear conduction heat transfer across the insulation layer is not represented here, it should be represented by a different method for accurate MPC calculations. In order to do that, the heat transfer coefficient times the surface area term in the process model is recalculated every 500 seconds using the recorded reactor temperature data. This helps to lump convection and conduction into the energy equation and improve MPC predictions. To update the UA term, temperature derivatives are calculated from data history using a centered finite-difference method. Following this, the corresponding temperatures, currents, potentials and ELO predictions are evaluated. These parameters

are used to minimize the distance between the experimental heat differential and Eq. 7.13 to find the most optimal UA term through an SQP (sequential quadratic programming) solver. As the UA term changes, the I_{sp} term in the MPC formulation (Eq. 7.18d) that corresponds to the set-point temperature changes as well. Thus, when the UA is recalculated, the procedure mentioned in Section 7.6.2 is repeated with the new UA value used in the process model to update the current setpoint.

Experimental data that the sensors are collecting are very noisy compared to simulation data. For example, the resistance changes often during the experiment, due to oxidation of the current collectors. The temperature signals obtained from the setup are not smooth due to environmental factors and sensor noise. In order to mitigate the effects of the noise, a Savitzky-Golay filter was applied with a polynomial order of 2 and window length of 40 seconds on current, temperature, and resistance values. Savitzky-Golay filter fits a second order polynomial to the last 40 seconds of data and uses the fitted polynomial data coefficients to estimate weights for smoothing the data [154]. The process model takes data points on a per-second basis, and the noise from the sensors causes noisy predictions. In addition to that, the ELO correction term might cause an amplification of the noise that would be reflected to the output predictions. Thus, de-noising the aforementioned signals will cause less fluctuating predictions in the subsequent gas flowrate estimations.

Experimental Implementation of MPC

The experimental real-time implementation of the proposed MPC control scheme is conducted with the same initial condition and setpoint as the PI control experiments for a fair comparison. We carried out two experiments under MPC using the same operating conditions and MPC

tuning parameters. The MPC sampling time is 5 seconds in the experimental implementation and the MPC was always solved within this time constraint. The first MPC experimental result is presented in Figure 7.22 and the second one is presented in Figure 7.23. From these two figures, it can be seen that the MPC successfully drives the process to the setpoint in both experiments, establishing the reproducibility of this experimental MPC implementation. Furthermore, it can be seen that it takes around 4 GC measurements for the output to reach the setpoint, which corresponds to around 72 minutes of process operation time, which is a significantly shorter approach to the setpoint than all the PI control experiments. Upon regulation at the setpoint, the GC results exhibit less variability around the setpoint compared to the output under PI control. With respect to the behavior of the control system, the current values calculated by the MPC change smoothly with time, leading to a smooth change of the temperature evolution in the reactor. Finally, we note that the chattering in the power signal is due to the noisy behavior of the voltage and current measurement of the heating system and it is best to view the power consumption in terms of a sufficiently long moving average that would yield the average value over time. We decided to present the instantaneous power consumption in the plots, as it is the primary data, and one can easily determine the average power from these results.
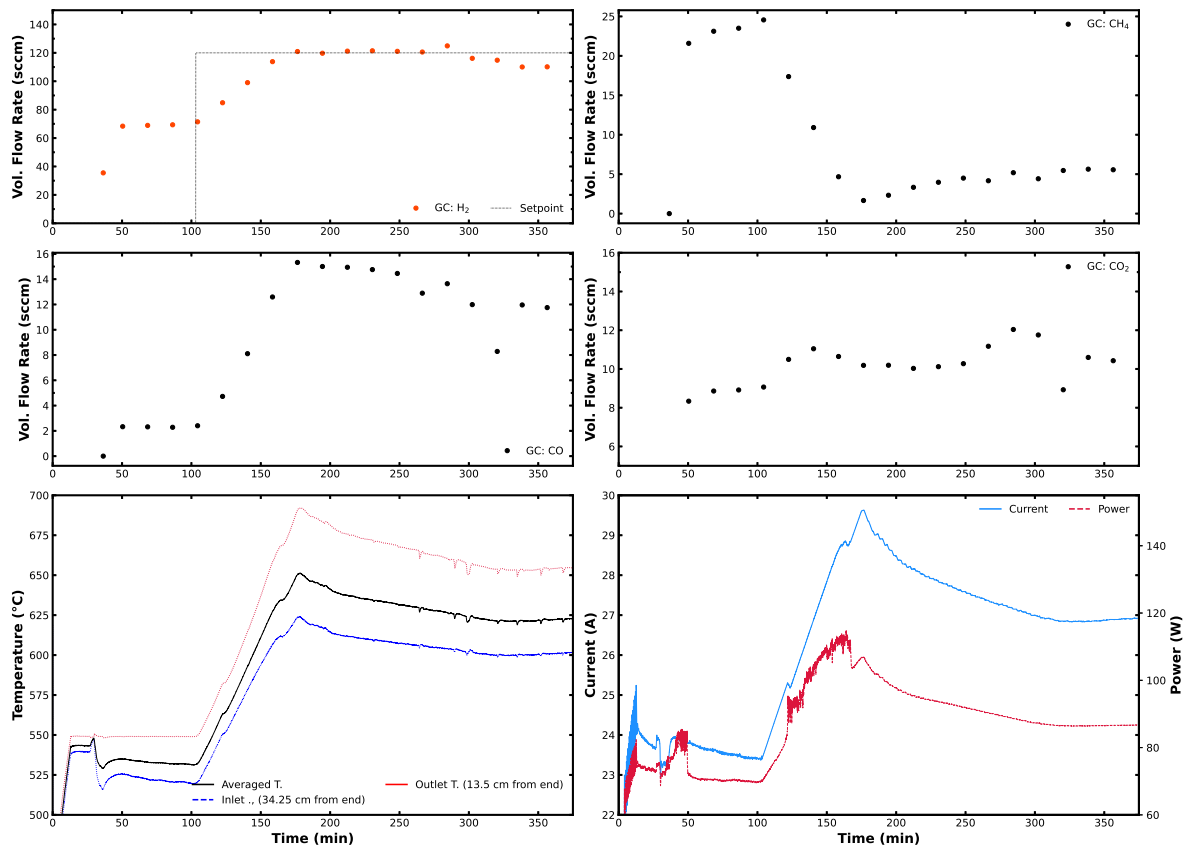
Figure 7.22: First closed-loop experiment under MPC.
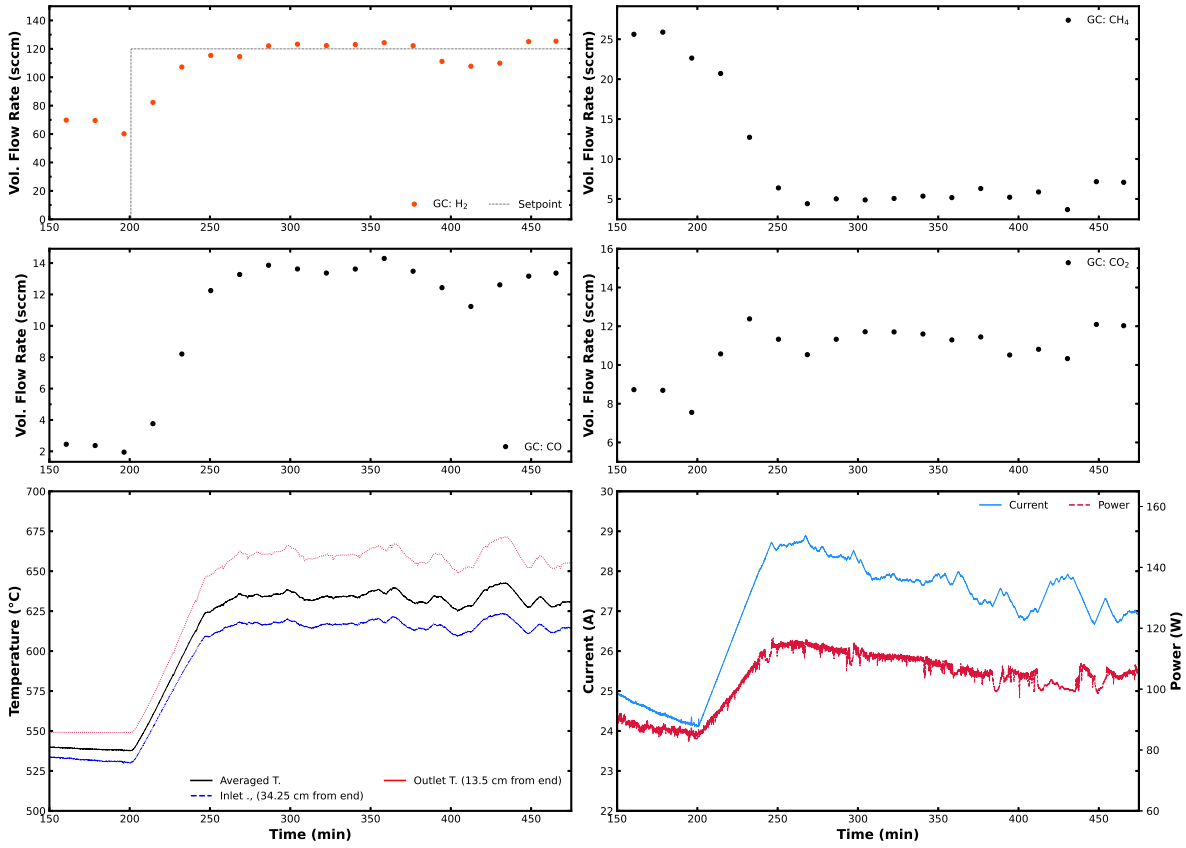
Figure 7.23: Second closed-loop experiment under MPC.

Chapter 8

Conclusions

This dissertation discussed the application of machine learning, and lumped parameter modeling and advanced control strategies for an experimental electrochemical reactor and an experimental electrically-heated steam methane reformer. Prior to experimental implementation of the models and controllers, the experimental setups are digitalized using smart manufacturing guidelines on a LabVIEW interface. After the digitalization, an SVR-based hybrid model was used to control C_2H_4 concentration with one PI-controller controlling surface potential despite the catalyst deactivation and selectivity shift above a certain surface potential. Following this work, a MIMO control scheme is proposed to control C_2H_4 and CO concentrations using two PI controllers controlling surface potential and catalyst rotation speed using an LSTM-based process model that behaved like a soft sensor to generate frequent concentration feedback. Then, the 2 PI MIMO scheme is improved to an MPC control with improvement in the LSTM model. For real-time implementation, the MPC model is linearized in real-time using a Koopman Operator, enabling a

319

quadratic programming optimization. Predictive control shows a control time 3 times faster than 2 PI controllers. After the electrochemical CO_2 reduction process control proposals were finalized, the electrically-heated SMR process was modeled with a lumped-parameter model using a derivation for variable temperature change. An MPC scheme was proposed, however, the experimental implementation posed challenges regarding the missing feedback of volumetric flow rates and water concentration from the GC. This issue was overcome by introducing an extended Luenberger observer to combine the lumped-parameter model with available GC and temperature sensor values for a perfect control of the Hydrogen concentration by manipulating temperature through electrical current.

Chapter 2 presented the implementation of support vector regression in an electrochemical reactor to represent process variables and concentration relationships and proposed an approach to combine the output of the SVR model-based gas-phase C_2H_4 concentration estimator with GC measurements to control the concentration of ethylene in the reactor. As the electrochemical CO_2 reduction and the reaction pathways are not fully known, current first-principal approximations are inadequate to set up an efficient control scheme. Furthermore, catalyst deactivation in the reactor was an inherent disturbance that increased process variability. To address this issue, the SVR model was built to model the experimental concentration and catalyst deactivation over an extensive window of operating conditions and was combined with first-principles modeling to predict the gas-phase ethylene concentration. The integral of the current was introduced as an input to contribute to the representation of the degradation of the catalyst, and the overall accuracy of the model was increased with feature engineering. In addition, the GC measurements were introduced to the controller to achieve a fully-closed-loop control scheme. Lastly, the economically-optimized

320

energy-efficient setpoints were calculated, and the controller was shown to be successful, as it could drive the process in the neighborhood of two energy-optimal set-points.

The approach shown in this report for the automation of the UCLA electrochemical reactor could be broadly applicable. For simpler electrochemical reactions involving a single reaction pathway, there is no challenge in selectivity and such systems require only the collection of electrical current data which is generated every second from the potentiostat. When the system produces more than one product, other analytical tools to quantify products should be implemented and automated. In chapter 2, we have demonstrated the online data processing of GC that automatically quantifies gases, which could be readily translated to other electrochemical reactions involving gas products. This approach could be expanded to other analytical and spectroscopic tools such as high-performance liquid chromatography (HPLC), ultraviolet-visible (UV-Vis) spectroscopy, and FTIR. Although the overall implementation process would be similar, it is important to note the intrinsic differences of analytical sensors. Understanding the intrinsic nature (e.g., detection limit, sampling, and response times) of sensors as detailed in this work will be the key to successfully apply the developed automation and control approach to other electrochemical reactor systems.

Chapter 3 demonstrated the digitization of the UCLA experimental electrochemical reactor using the Smart Manufacturing Innovation Platform and discussed the role of smart manufacturing and how it can accelerate sustainable energy research. Smart manufacturing fundamentals and concepts were explained and relevant examples from our previous works were given. The data-dependent nature of the analysis and modeling of the experimental electrochemical reactor (in order to overcome the lack of fundamental process understanding) was emphasized, and the development and incorporation of automated sensors were discussed in detail. In particular, the

321

principles for automating the gas chromatography sensor for real-time experiment automatic operation and feedback control were elucidated. In addition, the chapter explained the SMIP architecture and its capabilities including connectivity, data transfer, and use of Docker containers. These features offered by the SMIP considerably accelerate data acquisition and analysis, as well as machine learning modeling efforts, while keeping the proprietary data safe. The overall approach implemented on the UCLA electrochemical reactor is applicable to other experimental and industrial reactors, as well as other unit operation processes. Once the sensor, actuator and reactor profiles are developed, the automation, connectivity, and contextualization can be adopted by other experimental groups working on other processes with minimal training and effort.

Chapter 4 presented the development and implementation of a recurrent neural network-based modeling framework on an experimental electrochemical reactor. The RNN-based modeling framework was used to represent relationships between process variables and gas product concentrations at the outlet of the reactor. The proposed approach combined the output of LSTM model-based gas-phase ethylene and carbon monoxide concentration estimators with GC measurements to implement multivariable control of the production rate of these two products in the reactor. Steady state data was used to construct a feedforward neural network for the calculation of feasible operating points. Using this model and an optimization model, economically-optimal, energy-efficient set-points were computed. In the absence of first-principle modeling descriptions needed to implement an efficient control system for the reactor, experimental GC measurements and data regression techniques were used to construct probable experimental trajectories for experiments run over lengths of 7 hours while the GC data available is discrete in nature, and only collected every 20 minutes. The enhanced trajectory dataset was used to train the LSTM model,

which is the basis for the gas concentration estimator. In the implementation of the estimator-based multi-input multi-output control system, the real-time GC measurements are fed back to the controller for which the controller parameters have been tuned in different operating regions in order to better respond to the nonlinear nature of the electrochemical process. The estimator-based multi-input multi-output control system was successfully demonstrated to be capable of driving the process outputs to a variety of optimal setpoints for C_2H_4 and CO.

In **Chapter 5**, a procedure to apply neural network model-based MPC to perform real-time multivariable control for an experimental electrochemical reactor is presented; the approach involves on-line linearization of the neural network model and is applicable to broad classes of chemical processes. Specifically, in this study, an LSTM neural network model was used to capture the nonlinear dynamic input-output relationship to control an electrochemical reactor that converts CO_2 to valuable chemical products. The Koopman operator method was found to be able to linearize the LSTM model efficiently (in terms of computational effort) and effectively (in terms of model performance). Based on that method, a systematic approach was developed to linearize a neural network model using linear regression, which is efficient and easy to implement. Open-loop simulations were performed to evaluate the performance of the original LSTM and linearized LSTM models, and the MPC developed based on the linearization of the LSTM model was applied to control the experimental electrochemical reactor. As the closed-loop results demonstrated, the MPC calculates the optimal control actions with reasonable computation cost and successfully drives the process outputs to desired set-point values. Furthermore, a transfer-learning scheme was introduced to account for the data shift problem (due to catalyst activity variability every time a new catalyst is introduced) by updating the LSTM model using new process measurement data.

323

The transfer-learning method was demonstrated to be able to update the original LSTM model with a limited amount of new data and computational resources. Finally, the updated LSTM model and the resulting MPC were demonstrated to resolve the data shift problem by driving the process outputs to the desired set-points in a closed-loop experiment under the new experimental (catalyst) conditions.

In **Chapter 6**, it was presented that steam methane reforming could be made more sustainable through replacing fossil fuel based heating with electricity, and being able to control this process is the key to maximizing the energy efficiency. The experimental setup for a Joule-heated SMR was modeled with first-principle nonlinear dynamic equations with parameters calculated using experimental data. Set-point tracking control was simulated under PI control and MPC. It was demonstrated that MPC leads to an optimal closed-loop response and is robust to disturbances.

Chapter 7 presented the use of steady-state experimental data to develop a lumped parameter model for this process that was used in a model predictive control system. During the experimental procedure, the temperature rate of change increase was kept within a $6\ ^\circ C/min$ limit to prevent thermal damage to the catalyst. The model was incorporated into an MPC scheme used to drive the SMR process to a new H_2 production setpoint. To prove the efficiency of the MPC, PI control experiments were also conducted. While the PI controller successfully drove the H_2 production to the set-point under conditions like catalyst deactivation and coke formation, the MPC was found to be more efficient with a significantly faster approach to the setpoint while respecting control action constraints.

Bibliography

[1] S. Z. Abbas, V. Dupont, and T. Mahmud. Kinetics study and modelling of steam methane reforming process over a NiO/Al_2O_3 catalyst in an adiabatic packed bed reactor. *International Journal of Hydrogen Energy*, 42:2889–2903, 2017.

[2] M. S. Alhajeri, Z. Wu, D. Rincon, F. Albalawi, and P. D. Christofides. Machine-learning-based state estimation and predictive control of nonlinear processes. *Chemical Engineering Research and Design*, 167:268–280, 2021.

[3] M. R. Almind, S. B. Vendelbo, M. F. Hansen, M. G. Vinum, C. Frandsen, P. M. Mortensen, and J. S. Engbæk. Improving performance of induction-heated steam methane reforming. *Catalysis Today*, 342:13–20, 2020.

[4] H. Arbabi, M. Korda, and I. Mezić. A data-driven koopman model predictive control framework for nonlinear partial differential equations. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 6409–6414, 2018.

[5] H. Arbabi and I. Mezic. Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator. *SIAM Journal on Applied Dynamical Systems*, 16:2096–2126, 2017.

[6] U. Ashik, W. W. Daud, and H. F. Abbas. Methane decomposition kinetics and reaction rate over Ni/SiO_2 nanocatalyst produced through co-precipitation cum modified Stöber method. *International Journal of Hydrogen Energy*, 42:938–952, 2017.

[7] M. Autolab. *Autolab SDK User manual*. Methrom Autolab M.B., 2013.

[8] D. Basak, S. Pal, and D. Patranabis. Support vector regression. *Neural Information Processing – Letters and Reviews*, 11:203–224, 11 2007.

[9] A. Beerelli, P. C. Dheeraj, S. Arunkumar, and M. Venkatesan. Automated temperature control in batch reactor using LabVIEW. In *2014 IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–5, 2014.

[10] B. W. Bequette. *Process control: Modeling, design, and simulation*. Prentice Hall Professional, 2003.

[11] B. Bhadriraju, A. Narasingam, and J. S.-I. Kwon. Machine learning-based adaptive model identification of systems: Application to a chemical process. *Chemical Engineering Research and Design*, 152:372–383, 2019.

[12] P. T. Boggs and J. W. Tolle. Sequential quadratic programming. *Acta numerica*, 4:1–51, 1995.

[13] B. Botcha, Z. Wang, S. Rajan, G. Natarajan, S. Bukkapatnam, A. Mananthanwar, M. Scott, D. Schneider, and P. Korambath. Implementing the transformation of discrete manufacturing systems into smart manufacturing platforms. *Proceedings of the ASME International Manufacuturing Science and Engineering Conference*, 51371:V003T02A009, 2018.

[14] A. Boulamanti, J. A. Moya, et al. Energy efficiency and GHG emissions: Prospective scenarios for the chemical and petrochemical industry. *Report 9789279657344, EU Science Hub*, 2017.

[15] G. Brito and M. T. Valente. REST vs GraphQL: A controlled experiment. In *Proceedings of IEEE International Conference on Software Architecture*, pages 81–91, 2020.

[16] M. M. Brubaker, D. D. Coffman, and H. H. Hoehn. Synthesis and characterization of ethylene/carbon monoxide copolymers, a new class of polyketones. *Journal of the American Chemical Society*, 74:1509–1515, 1952.

[17] S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz. Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. *PLOS One*, 11:e0150171, 2016.

[18] C. J. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[19] V. Canuso. Machine learning-based operational modeling of an electrochemical reactor: Handling data variability for experimental data. Master's thesis, University of California, Los Angeles, 2022.

[20] S. Chavan, N. Birnale, and A. S. Deshpande. Design and simulation of model predictive control for multivariable distillation column. In *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pages 764–768, 2018.

[21] A. Chen, X. Zhang, L. Chen, S. Yao, and Z. Zhou. A machine learning model on simple features for CO_2 reduction electrocatalysts. *The Journal of Physical Chemistry C*, 124:22471–22478, 2020.

[22] Y. Cheng, T. W. Karjala, and D. M. Himmelblau. Identification of nonlinear dynamic processes with unknown and variable dead time using an internal recurrent neural network. *Industrial & Engineering Chemistry Research*, 34:1735–1742, 1995.

[23] N. Choudhary and D. N. Saraf. Hydrocracking: A review. *Industrial & Engineering Chemistry Product Research and Development*, 14:74–83, 1975.

[24] P. D. Christofides, J. F. Davis, N. H. El-Farra, D. Clark, K. R. Harris, and J. N. Gipson. Smart plant operations: Vision, progress and challenges. *AIChE Journal*, 53:2734–2741, 2007.

[25] J.-P. Corriou. *Process Control: Theory and Applications*. Springer, second edition, 2004.

[26] C. Cortes, M. Mohri, and A. Rostamizadeh. L2 regularization for learning kernels. *arXiv preprint arXiv:1205.2653*, 2012.

[27] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

[28] E. B. Creel, E. R. Corson, J. Eichhorn, R. Kostecki, J. J. Urban, and B. D. McCloskey. Directing selectivity of electrochemical carbon dioxide reduction using plasmonics. *ACS Energy Letters*, 4:1098–1105, 2019.

[29] X. Cui, B. Çıtmacı, D. Peters, F. Abdullah, Y. Wang, E. Hsu, P. Chheda, C. G. Morales-Guio,

and P. D. Christofides. Estimation-based model predictive control of an electrically-heated steam methane reforming process. *Digital Chemical Engineering*, 11:100153, 2024.

[30] J. Davis, T. Edgar, R. Graybill, P. Korambath, B. Schott, D. Swink, J. Wang, and J. Wetzel. Smart manufacturing. *Annual Review of Chemical and Biomolecular engineering*, 6:141–160, 2015.

[31] J. Davis, H. Malkani, J. Dyck, P. Korambath, and J. Wise. Chapter 4 - Cyberinfrastructure for the democratization of smart manufacturing. In M. Soroush, M. Baldea, and T. F. Edgar, editors, *Smart Manufacturing: Concepts and Methods*, pages 83–116. Elsevier, 2020.

[32] R. De, S. Gonglach, S. Paul, M. Haas, S. Sreejith, P. Gerschel, U.-P. Apfel, T. H. Vuong, J. Rabeah, S. Roy, et al. Electrocatalytic reduction of CO_2 to acetic acid by a molecular manganese corrole complex. *Angewandte Chemie*, 132:10614–10621, 2020.

[33] P. De Luna, C. Hahn, D. Higgins, S. A. Jaffer, T. F. Jaramillo, and E. H. Sargent. What would it take for renewably powered electrosynthesis to displace petrochemical processes? *Science*, 364:3506, 2019.

[34] D. W. DeWulf, T. Jin, and A. J. Bard. Electrochemical and surface studies of carbon dioxide reduction to methane and ethylene at copper electrodes in aqueous solutions. *Journal of The Electrochemical Society*, 136:1686–1691, 1989.

[35] T. N. Do, H. Kwon, M. Park, C. Kim, Y. T. Kim, and J. Kim. Carbon-neutral hydrogen production from natural gas via electrified steam reforming: Techno-economic-environmental perspective. *Energy Conversion and Management*, 279:116758, 2023.

[36] M. R. Dobbelaere, P. P. Plehiers, R. Van de Vijver, C. V. Stevens, and K. M. Van Geem. Machine learning in chemical engineering: Strengths, weaknesses, opportunities, and threats. *Engineering*, 7:1201–1211, 2021.

[37] F. J. Doyle, C. A. Harrison, and T. J. Crowley. Hybrid model-based approach to batch-to-batch control of particle size distribution in emulsion polymerization. *Computers & Chemical Engineering*, 27:1153–1163, 2003.

[38] T. F. Edgar and E. N. Pistikopoulos. Smart manufacturing and energy systems. *Computers & Chemical Engineering*, 114:130–144, 2018.

[39] P. H. Eilers and H. F. Boelens. Baseline correction with asymmetric least squares smoothing. *Leiden University Medical Centre Report*, 1:1–24, 2005.

[40] K. Elsayed and C. Lacor. Modeling and pareto optimization of gas cyclone separator performance using RBF type artificial neural networks and genetic algorithms. *Powder Technology*, 217:84–99, 2012.

[41] A. Engelbrecht, C. Uhlig, O. Stark, M. Hämmerle, G. Schmid, E. Magori, K. Wiesner-Fleischer, M. Fleischer, and R. Moos. On the electrochemical CO_2 reduction at copper sheet electrodes with enhanced long-term stability by pulsed electrolysis. *Journal of The Electrochemical Society*, 165:J3059–J3068, 2018.

[42] L. Fan, Y. Zhao, L. Chen, J. Chen, J. Chen, H. Yang, Y. Xiao, T. Zhang, J. Chen, and L. Wang. Selective production of ethylene glycol at high rate via cascade catalysis. *Nature Catalysis*, 2023.

[43] Q. Fan, M. Zhang, M. Jia, S. Liu, J. Qiu, and Z. Sun. Electrochemical CO_2 reduction to C_{2+} species: Heterogeneous electrocatalysts, reaction pathways, and optimization strategies. *Materials today energy*, 10:280–301, 2018.

[44] M. Feurer and F. Hutter. Hyperparameter optimization. In *Automated machine learning: Methods, systems, challenges*, pages 3–33. Springer, 2019.

[45] H. S. Fogler. *Elements of Chemical Reaction Engineering (4th Edition)*. Prentice Hall, 4 edition, 2005.

[46] T. N. From, B. Partoon, M. Rautenbach, M. Østberg, A. Bentien, K. Aasberg-Petersen, and P. M. Mortensen. Electrified steam methane reforming of biogas for sustainable syngas manufacturing and next-generation of plant design: A pilot plant study. *Chemical Engineering Journal*, 479:147205, 2024.

[47] F. P. García de Arquer, C.-T. Dinh, A. Ozden, J. Wicks, C. McCallum, A. R. Kirmani, D.-H. Nam, C. Gabardo, A. Seifitokaldani, X. Wang, et al. CO_2 electrolysis to multicarbon products at activities greater than 1 A cm^{-2}. *Science*, 367:661–666, 2020.

[48] J. M. Ginsburg, J. Piña, T. El Solh, and H. I. De Lasa. Coke formation over a nickel catalyst under methane dry reforming conditions: Thermodynamic and kinetic models. *Industrial & Engineering Chemistry Research*, 44:4846–4854, 2005.

[49] F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7:219–269, 1995.

[50] J. C. B. Gonzaga, L. A. C. Meleiro, C. Kiang, and R. Maciel Filho. ANN-based soft-sensor for real-time process monitoring and control of an industrial polymerization process. *Computers & Chemical Engineering*, 33:43–49, 2009.

[51] C. A. Grambow, L. Pattanaik, and W. H. Green. Deep learning of activation energies. *The Journal of Physical Chemistry Letters*, 11:2992–2997, 2020.

[52] L. Green Jr. An ammonia energy vector for the hydrogen economy. *International Journal of Hydrogen Energy*, 7:355–359, 1982.

[53] L. Guelman. Gradient boosting trees for auto insurance loss cost modeling and prediction. *Expert Systems with Applications*, 39:3659–3667, 2012.

[54] O. J. Guerra, J. Eichman, J. Kurtz, and B.-M. Hodge. Cost competitiveness of electrolytic hydrogen. *Joule*, 3:2425–2443, 2019.

[55] T. Haas, C. Schubert, M. Eickhoff, and H. Pfeifer. Bubcnn: Bubble detection using faster rcnn and shape regression network. *Chemical Engineering Science*, 216:115467, 2020.

[56] N. M. Haegel, H. Atwater Jr, T. Barnes, C. Breyer, A. Burrell, Y.-M. Chiang, S. De Wolf, B. Dimmler, D. Feldman, S. Glunz, et al. Terawatt-scale photovoltaics: Transform global energy. *Science*, 364:836–838, 2019.

[57] N. M. Haegel, R. Margolis, T. Buonassisi, D. Feldman, A. Froitzheim, R. Garabedian, M. Green, S. Glunz, H.-M. Henning, B. Holder, et al. Terawatt-scale photovoltaics: Trajectories and challenges. *Science*, 356:141–143, 2017.

[58] N. Hambali, A. Masngut, A. A. Ishak, and Z. Janin. Process controllability for flow control system using Ziegler-Nichols (ZN), Cohen-Coon (CC) and Chien-Hrones-Reswick (CHR) tuning methods. In *Proceedings of the International Conference on Smart Instrumentation, Measurement and Applications*, Kuala Lumpur, Malaysia, 2014.

[59] H.-G. Han, L. Zhang, Y. Hou, and J.-F. Qiao. Nonlinear model predictive control based on a self-organizing recurrent neural network. *IEEE Transactions on Neural Networks and Learning Systems*, 27:402–415, 2015.

[60] O. Hartig and J. Pérez. Semantics and complexity of graphql. In *Proceedings of the World Wide Web Conference*, WWW '18, page 1155–1164, Lyon, France, 2018. International World Wide Web Conferences Steering Committee.

[61] O. Hartig and J. Pérez. Semantics and complexity of GraphQL. In *Proceedings of the 2018 World Wide Web Conference*, pages 1155–1164, 2018.

[62] L. He, Y. Fan, L. Luo, J. Bellettre, and J. Yue. Preparation of $Pt/\gamma - Al_2O_3$ catalyst coating in microreactors for catalytic methane combustion. *Chemical Engineering Journal*, 380:122424, 2020.

[63] J. Heaton. An empirical analysis of feature engineering for predictive modeling. In *Proceedings of SoutheastCon 2016*, pages 1–6, Norfolk, VA, USA, 2016.

[64] L. Hewing, J. Kabzan, and M. N. Zeilinger. Cautious model predictive control using Gaussian process regression. *IEEE Transactions on Control Systems Technology*, 28:2736–2743, 2019.

[65] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 1997.

[66] K. Holkar and L. M. Waghmare. An overview of model predictive control. *International Journal of Control and Automation*, 3:47–63, 2010.

[67] Y. Hori, H. Konishi, T. Futamura, A. Murata, O. Koga, H. Sakurai, and K. Oguma. Deactivation of copper electrode in electrochemical reduction of CO_2. *Electrochimica Acta*, 50:5354–5369, 2005.

[68] Y. Hori, R. Takahashi, Y. Yoshinami, and A. Murata. Electrochemical reduction of co at a copper electrode. *Journal of Physical Chemistry B*, 101:7075–7081, 1997.

[69] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2:359–366, 1989.

[70] M. A. Hussain. Review of the applications of neural networks in chemical process control—simulation and online implementation. *Artificial Intelligence in Engineering*, 13:55–68, 1999.

[71] IEA. Tracking clean energy progress 2023. IEA Paris, France, 2023.

[72] P. Jain, I. Rahman, and B. D. Kulkarni. Development of a soft sensor for a batch distillation column using support vector regression techniques. *Chemical Engineering Research and Design*, 85:283–287, 2007.

[73] J. Jang, M. Rüscher, M. Winzely, and C. G. Morales-Guio. Gastight rotating cylinder electrode: Towards decoupling mass transport and intrinsic kinetics in electrocatalysis. *AIChE Journal*, 65:e17605, 2022.

[74] E. Jeng, Z. Qi, A. R. Kashi, S. Hunegnaw, Z. Huo, J. S. Miller, L. B. Bayu Aji, B. H. Ko, H. Shin, S. Ma, et al. Scalable gas diffusion electrode fabrication for electrochemical co2 reduction using physical vapor deposition methods. *ACS Applied Materials & Interfaces*, 14:7731–7740, 2022.

[75] S. Jin, Z. Hao, K. Zhang, Z. Yan, and J. Chen. Advances and challenges for the electrochemical reduction of CO_2 to CO: From fundamentals to industrialization. *Angewandte Chemie International Edition*, 60:20627–20648, 2021.

[76] S. Johari, M. Yaghoobi, and H. R. Kobravi. Nonlinear model predictive control based on hyper chaotic diagonal recurrent neural network. *Journal of Central South University*, 29:197–208, 2022.

[77] M. Jouny, W. Luc, and F. Jiao. General techno-economic analysis of CO_2 electrolysis systems. *Industrial & Engineering Chemistry Research*, 57:2165–2177, 2018.

[78] R. E. Kalman. A new approach to linear filtering and prediction problems. 1960.

[79] H. S. Kang, J. Y. Lee, S. Choi, H. Kim, J. H. Park, J. Y. Son, B. H. Kim, and S. D. Noh. Smart manufacturing: Past research, present findings, and future directions. *International journal of precision engineering and manufacturing-green technology*, 3:111–128, 2016.

[80] R. Kas, R. Kortlever, H. Yılmaz, M. T. M. Koper, and G. Mul. Manipulating the hydrocarbon selectivity of copper nanoparticles in CO_2 electroreduction by process conditions. *ChemElectroChem*, 2:354–358, 2015.

[81] J. Ke, C. Gao, A. A. Folgueiras-Amador, K. E. Jolley, O. de Frutos, C. Mateos, J. A. Rincón, R. C. Brown, M. Poliakoff, and M. W. George. Self-optimization of continuous flow electrochemical synthesis using Fourier transform infrared spectroscopy and gas chromatography. *Applied Spectroscopy*, 76:38–50, 2022.

[82] M. Khalid, S. Omatu, and R. Yusof. MIMO furnace control with neural networks. *IEEE Transactions on Control Systems Technology*, 1:238–245, 1993.

[83] B. Kim, Y. C. Tan, Y. Ryu, K. Jang, H. G. Abbas, T. Kang, H. Choi, K.-S. Lee, S. Park, W. Kim, et al. Trace-level cobalt dopants enhance CO_2 electroreduction and ethylene formation on copper. *ACS Energy Letters*, 8:3356–3364, 2023.

[84] Y.-G. Kim, J. H. Baricuatro, A. Javier, J. M. Gregoire, and M. P. Soriaga. The evolution of the polycrystalline copper surface, first to Cu (111) and then to Cu (100), at a fixed CO_2RR potential: A study by operando EC-STM. *Langmuir*, 30:15053–15056, 2014.

[85] Y.-G. Kim, J. H. Baricuatro, and M. P. Soriaga. 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, 2018.

[86] E. Komp and S. Valleau. Machine learning quantum reaction rate constants. *The Journal of Physical Chemistry A*, 124:8607–8613, 2020.

[87] B. O. Koopman. Hamiltonian systems and transformation in Hilbert space. *Proceedings of the National Academy of Sciences*, 17:315–318, 1931.

[88] B. O. Koopman and J. V. Neumann. Dynamical systems of continuous spectra. *Proceedings of the National Academy of Sciences*, 18:255–263, 1932.

[89] P. Korambath, J. Wang, A. Kumar, J. Davis, R. Graybill, B. Schott, and M. Baldea. A smart manufacturing use case: Furnace temperature balancing in steam methane reforming process via Kepler workflows. *Procedia Computer Science*, 80:680–689, 2016.

[90] M. Korda and I. Mezić. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica*, 93:149–160, 2018.

[91] R. Kortlever, J. Shen, K. J. P. Schouten, F. Calle-Vallejo, and M. T. Koper. Catalysts and reaction pathways for the electrochemical reduction of carbon dioxide. *The Journal of Physical Chemistry Letters*, 6:4073–4082, 2015.

[92] A. Kramer and F. Morgado-Dias. Applications of artificial neural networks in process control applications: A review. In *Proceedings of International Conference on Biomedical Engineering and Applications*, Funchal, Portugal, 2018.

[93] A. Kumar, M. Baldea, and T. F. Edgar. A physics-based model for industrial steam-methane reformer optimization with non-uniform temperature field. *Computers & Chemical Engineering*, 105:224–236, 2017.

[94] L. Lao, A. Aguirre, A. Tran, Z. Wu, H. Durand, and P. D. Christofides. CFD modeling and

control of a steam methane reforming reactor. *Chemical Engineering Science*, 148:78–92, 2016.

[95] D. A. Latham, K. B. McAuley, B. A. Peppley, and T. M. Raybold. Mathematical modeling of an industrial steam-methane reformer for on-line deployment. *Fuel Processing Technology*, 92:1574–1586, 2011.

[96] M. Ławryńczuk. *Computationally efficient model predictive control algorithms: A Neural Network Approach*, volume 3. Studies in Systems, Decision and Control Series, Springer, 2014.

[97] J. H. Lee. Model predictive control: Review of the three decades of development. *International Journal of Control, Automation and Systems*, 9:415–424, 2011.

[98] F. Li, A. Thevenon, A. Rosas-Hernández, Z. Wang, Y. Li, C. M. Gabardo, A. Ozden, C. T. Dinh, J. Li, Y. Wang, et al. Molecular tuning of CO_2-to-ethylene conversion. *Nature*, 577:509–513, 2020.

[99] M. Li, H. Wang, W. Luo, P. C. Sherrell, J. Chen, and J. Yang. Heterogeneous single-atom catalysts for electrochemical CO_2 reduction reaction. *Advanced Materials*, 32:2001848, 2020.

[100] X. Li, S. Wang, L. Li, Y. Sun, and Y. Xie. Progress and perspective for in situ studies of CO_2 reduction. *Journal of the American Chemical Society*, 142:9567–9581, 2020.

[101] Z. Li, L. Wang, T. Wang, L. Sun, and W. Yang. Steering the dynamics of reaction interme-

diates and catalyst surface during electrochemical pulsed CO_2 reduction for enhanced C_{2+} selectivity. *Journal of the American Chemical Society*, 145:20655–20664, 2023.

[102] M.-H. Lin, J. G. Carlsson, D. Ge, J. Shi, and J.-F. Tsai. A review of piecewise linearization methods. *Mathematical Problems in Engineering*, 2013:101376, 2013.

[103] W. Liu, H. Zuo, J. Wang, Q. Xue, B. Ren, and F. Yang. The production and application of hydrogen in steel industry. *International Journal of Hydrogen Energy*, 46:10548–10569, 2021.

[104] D. G. Luenberger. Observing the state of a linear system. *IEEE Transactions on Military Electronics*, 8:74–80, 1964.

[105] J. Luo, V. Canuso, J. B. Jang, Z. Wu, C. G. Morales-Guio, and P. D. Christofides. Machine learning-based operational modeling of an electrochemical reactor: Handling data variability and improving empirical models. *Industrial & Engineering Chemistry Research*, 61:8399–8410, 2022.

[106] Z. Lv, X. Yan, and Q. Jiang. Batch process monitoring based on just-in-time learning and multiple-subspace principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 137:128–139, 2014.

[107] B. Mahesh. Machine learning algorithms - a review. *International Journal of Science and Research*, 9:381–386, 2020.

[108] A. Malek, Q. Wang, S. Baumann, O. Guillon, M. Eikerling, and K. Malek. A data-driven

framework for the accelerated discovery of CO_2 reduction electrocatalysts. *Frontiers in Energy Research*, 9:52, 2021.

[109] H. Malerød-Fjeld, D. Clark, I. Yuste-Tirados, R. Zanón, D. Catalán-Martinez, D. Beeaff, S. H. Morejudo, P. K. Vestre, T. Norby, R. Haugsrud, et al. Thermo-electrochemical production of compressed hydrogen from methane with near-zero energy loss. *Nature Energy*, 2:923–931, 2017.

[110] R. Maya-Yescas, J. Lara-Romero, and R. Aguilar-López. Controllability of lumped parameter chemical reactors. *chemical Papers-Slovak Academy OF Sciences*, 59:487, 2005.

[111] E. Meloni, M. Martino, and V. Palma. A short review on Ni based catalysts and related engineering issues for methane steam reforming. *Catalysts*, 10:352, 2020.

[112] P. Mendis, C. Wickramasinghe, M. Narayana, and C. Bayer. Adaptive model predictive control with successive linearization for distillate composition control in batch distillation. In *2019 Moratuwa Engineering Research Conference (MERCon)*, pages 366–369, 2019.

[113] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. A method for PID controller tuning using nonlinear control techniques. In *Proceedings of the 2004 American Control Conference*, volume 4, pages 2925–2930, Boston, Massachusetts, 2004.

[114] A. Mistry, A. A. Franco, S. J. Cooper, S. A. Roberts, and V. Viswanathan. How machine learning will revolutionize electrochemical sciences. *ACS Energy Letters*, 6:1422–1431, 2021.

[115] S. Mohanty. Artificial neural network based system identification and model predictive control of a flotation column. *Journal of Process Control*, 19:991–999, 2009.

[116] E. M. Mokheimer, M. Ibrar Hussain, S. Ahmed, M. A. Habib, and A. A. Al-Qutub. On the modeling of steam methane reforming. *Journal of Energy Resources Technology*, 137:012001, 2015.

[117] A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, and S. D. Brown. An introduction to decision tree modeling. *Journal of Chemometrics*, 18:275–285, 2004.

[118] M. Nesselberger, S. J. Ashton, G. K. Wiberg, and M. Arenz. Design, development, and demonstration of a fully LabVIEW controlled in situ electrochemical Fourier transform infrared setup combined with a wall-jet electrode to investigate the electrochemical interface of nanoparticulate electrocatalysts under reaction conditions. *Review of Scientific Instruments*, 84, 2013.

[119] P. Nikolaidis and A. Poullikkas. A comparative overview of hydrogen production processes. *Renewable and Sustainable Energy Reviews*, 67:597–611, 2017.

[120] S. Nitopi, E. Bertheussen, S. B. Scott, X. Liu, A. K. Engstfeld, S. Horch, B. Seger, I. E. Stephens, K. Chan, C. Hahn, et al. Progress and perspectives of electrochemical CO_2 reduction on copper in aqueous electrolyte. *Chemical Reviews*, 119:7610–7672, 2019.

[121] S. Nitopi, E. Bertheussen, S. B. Scott, X. Liu, A. K. Engstfeld, S. Horch, B. Seger, I. E. L. Stephens, K. Chan, C. Hahn, J. K. Nørskov, T. F. Jaramillo, and I. Chorkendorff. Progress

and perspectives of electrochemical CO_2 reduction on copper in aqueous electrolyte. *Chemical Reviews*, 119:7610–7672, 2019.

[122] L. Niu, L. An, X. Wang, and Z. Sun. Effect on electrochemical reduction of nitrogen to ammonia under ambient conditions: Challenges and opportunities for chemical fuels. *Journal of Energy Chemistry*, 61:304–318, 2021.

[123] F. Núñez, S. Langarica, P. Díaz, M. Torres, and J. C. Salas. Neural network-based model predictive control of a paste thickener over an industrial internet platform. *IEEE Transactions on Industrial Informatics*, 16:2859–2867, 2019.

[124] Ç. Odabaşı, P. Dologlu, F. Gülmez, G. Kuşoğlu, and Ö. Çağlar. Investigation of the factors affecting reverse osmosis membrane performance using machine-learning techniques. *Computers & Chemical Engineering*, 159:107669, 2022.

[125] A. Ozden, Y. Wang, F. Li, M. Luo, J. Sisler, A. Thevenon, A. Rosas-Hernández, T. Burdyny, Y. Lum, H. Yadegari, T. Agapie, J. C. Peters, E. H. Sargent, and D. Sinton. Cascade CO_2 electroreduction enables efficient carbonate-free production of ethylene. *Joule*, 5:706–719, 2021.

[126] T. Park and G. Casella. The Bayesian Lasso. *Journal of the American Statistical Association*, 103:681–686, 2008.

[127] X. Pascual, H. Gu, A. R. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, P. D. Christofides, and Y. Cohen. Data-driven models of steady state and transient operations of spiral-wound RO plant. *Desalination*, 316:154–161, 2013.

[128] R. M. Perkin. Electrically generated heat. *Ullmann's Encyclopedia of Industrial Chemistry*, 2000.

[129] S. Phuyal, D. Bista, and R. Bista. Challenges, opportunities and future directions of smart manufacturing: A state of art review. *Sustainable Futures*, 2:100023, 2020.

[130] E. N. Pistikopoulos, N. A. Diangelakis, R. Oberdieck, M. M. Papathanasiou, I. Nascu, and M. Sun. PAROC—An integrated framework and software platform for the optimisation and advanced model-based control of process systems. *Chemical Engineering Science*, 136:115–138, 2015.

[131] S. Popović, M. Smiljanić, P. Jovanovič, J. Vavra, R. Buonsanti, and N. Hodnik. Stability and degradation mechanisms of copper-based catalysts for electrochemical CO_2 reduction. *Angewandte Chemie*, 132:14844–14854, 2020.

[132] A. Pozzi, M. Zambelli, A. Ferrara, and D. M. Raimondo. Balancing-aware charging strategy for series-connected lithium-ion cells: A nonlinear model predictive control approach. *IEEE Transactions on Control Systems Technology*, 28:1862–1877, 2020.

[133] J. L. Proctor, S. L. Brunton, and J. N. Kutz. Dynamic mode decomposition with control. *SIAM Journal on Applied Dynamical Systems*, 15:142–161, 2016.

[134] J. L. Proctor, S. L. Brunton, and J. N. Kutz. Generalizing Koopman theory to allow for inputs and control. *SIAM Journal on Applied Dynamical Systems*, 17:909–930, 2018.

[135] S. J. Qin and T. A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11:733–764, 2003.

[136] B. B. Rad, H. J. Bhatti, and M. Ahmadi. An introduction to docker and analysis of its performance. *International Journal of Computer Science and Network Security (IJCSNS)*, 17:228, 2017.

[137] R. Ramachandran and R. K. Menon. An overview of industrial uses of hydrogen. *International Journal of Hydrogen Energy*, 23:593–598, 1998.

[138] M. Ramdin, B. De Mot, A. R. Morrison, T. Breugelmans, L. J. Van Den Broeke, J. P. Trusler, R. Kortlever, W. De Jong, O. A. Moultos, P. Xiao, P. A. Webley, and T. J. Vlugt. Electroreduction of CO_2/CO to C_2Products: Process Modeling, Downstream Separation, System Integration, and Economic Analysis. *Ind. Eng. Chem. Res.*, 60:17862–17880, 2021.

[139] M. Ramdin, B. De Mot, A. R. T. Morrison, T. Breugelmans, L. J. P. van den Broeke, J. P. M. Trusler, R. Kortlever, W. de Jong, O. A. Moultos, P. Xiao, P. A. Webley, and T. J. H. Vlugt. Electroreduction of CO_2/CO to C_2 products: Process modeling, downstream separation, system integration, and economic analysis. *Industrial & Engineering Chemistry Research*, 60:17862–17880, 2021.

[140] M. Ramdin, O. A. Moultos, L. J. van den Broeke, P. Gonugunta, P. Taheri, and T. J. Vlugt. Carbonation in Low-Temperature CO_2 Electrolyzers: Causes, Consequences, and Solutions. *Ind. Eng. Chem. Res.*, 62:6843–6864, 2023.

[141] S. Ray. A quick review of machine learning algorithms. In *Proceedings of 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing*, pages 35–39, Faridabad, India, 2019.

[142] P. Reiser, M. Neubert, A. Eberhard, L. Torresi, C. Zhou, C. Shao, H. Metni, C. van Hoesel, H. Schopmans, T. Sommer, et al. Graph neural networks for materials science and chemistry. *Communications Materials*, 3:93, 2022.

[143] Y. M. Ren, M. S. Alhajeri, J. Luo, S. Chen, F. Abdullah, Z. Wu, and P. D. Christofides. A tutorial review of neural network modeling approaches for model predictive control. *Computers & Chemical Engineering*, 165:107956, 2022.

[144] Y. M. Ren, Y. Ding, Y. Zhang, and P. D. Christofides. A three-level hierachical framework for additive manufacturing. *Digital Chemical Engineering*, 1:100001, 2021.

[145] J. Richalet. Industrial applications of model based predictive control. *Automatica*, 29:1251–1274, 1993.

[146] D. Richard, J. Jang, B. Çıtmacı, J. Luo, V. Canuso, P. Korambath, O. Morales-Leslie, J. F. Davis, H. Malkani, P. D. Christofides, and C. G. Morales-Guio. Smart manufacturing inspired approach to research, development, and scale-up of electrified chemical manufacturing systems. *iScience*, 26:106966, 2023.

[147] D. Richard, M. Tom, J. Jang, S. Yun, P. D. Christofides, and C. G. Morales-Guio. Quantifying transport and electrocatalytic reaction processes in a gastight rotating cylinder electrode reactor via integration of computational fluid dynamics modeling and experiments. *Electrochimica Acta*, 440:141698, 2023.

[148] D. M. Richard. Development and testing of two lab-scale reactors for electrified steam methane reforming. Master's thesis, UCLA, 2021.

[149] F. S. Roberts, K. P. Kuhl, and A. Nilsson. High selectivity for ethylene from carbon dioxide reduction over copper nanocube electrocatalysts. *Angewandte Chemie*, 127:5268–5271, 2015.

[150] D. Roizard. *Antoine Equation*, pages 97–99. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

[151] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson. Spectral analysis of nonlinear flows. *Journal of Fluid Mechanics*, 641:115–127, 2009.

[152] F. Safari and I. Dincer. A review and comparative evaluation of thermochemical water splitting cycles for hydrogen production. *Energy Conversion and Management*, 205:112182, 2020.

[153] M. Saudagar, M. Ye, S. Al-Otaibi, and K. Al-Jarba. Smart manufacturing: Hope or hype. *Chemical Engineering Progress*, 115:43–48, 2019.

[154] A. Savitzky and M. J. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36:1627–1639, 1964.

[155] F. Scarselli and A. C. Tsoi. Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results. *Neural Networks*, 11:15–37, 1998.

[156] T. S. Schei. A finite-difference method for linearization in nonlinear estimation algorithms. *Automatica*, 33:2053–2058, 1997.

[157] P. J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.

[158] A. M. Schweidtmann, E. Esche, A. Fischer, M. Kloft, J.-U. Repke, S. Sager, and A. Mitsos. Machine learning in chemical engineering: A perspective. *Chemie Ingenieur Technik*, 93:2029–2039, 2021.

[159] Scipy. Scipy peak prominences api. `https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.peak_prominences.html`, 2022. Accessed: 2022-07-12.

[160] S. Semeniuta, A. Severyn, and E. Barth. Recurrent dropout without memory loss. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, Osaka, Japan, 2016.

[161] Y. Shin, R. Smith, and S. Hwang. Development of model predictive control system using an artificial neural network: A case study with a distillation column. *Journal of Cleaner Production*, 277:124124, 2020.

[162] R. Shiratsuchi, Y. Aikoh, and G. Nogami. Pulsed electroreduction of CO_2 on copper electrodes. *Journal of The Electrochemical Society*, 140:3479–3482, 1993.

[163] A. Shokry, F. Audino, P. Vicente, G. Escudero, M. P. Moya, M. Graells, and A. Espuña. Modeling and simulation of complex nonlinear dynamic processes using data based models: Application to Photo-Fenton process. *Computer Aided Chemical Engineering*, 37:191–196, 2015.

[164] C. H. Shomate. A method for evaluating and correlating thermodynamic data. *The Journal of Physical Chemistry*, 58:368–372, 1954.

[165] A. Simeone, A. Caggiano, L. Boun, and B. Deng. Intelligent cloud manufacturing platform for efficient resource sharing in smart manufacturing networks. *Procedia Cirp*, 79:233–238, 2019.

[166] B. Singh, P. Sihag, and K. Singh. Modelling of impact of water quality on infiltration rate of soil by random forest regression. *Modeling Earth Systems and Environment*, 3:999–1004, 2017.

[167] S. S. Soomro, D. Cozzula, W. Leitner, H. Vogt, and T. E. Müller. The microstructure and melt properties of CO-ethylene copolymers with remarkably low CO content. *Polym. Chem.*, 5:3831–3837, 2014.

[168] D. Spagnolo, L. Cornett, and K. Chuang. Direct electro-steam reforming: A novel catalytic approach. *International Journal of Hydrogen Energy*, 17:839–846, 1992.

[169] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15:1929–1958, 2014.

[170] J. Street, F. Yu, J. Warnock, J. Wooten, E. Columbus, and M. White. Design and testing of a Labview-controlled catalytic packed-bed reactor system for production of hydrocarbon fuels. *Transactions of the ASABE*, 55:1047–1055, 2012.

[171] I. Sullivan, A. Goryachev, I. A. Digdaya, X. Li, H. A. Atwater, D. A. Vermaas, and C. Xiang. Coupling electrochemical CO_2 conversion with CO_2 capture. *Nature Catalysis*, 4:952–958, 2021.

[172] P. Tan, C. Zhang, J. Xia, Q. Fang, and G. Chen. NO_X emission model for coal-fired boilers using principle component analysis and support vector regression. *Journal of Chemical Engineering of Japan*, 49:211–216, 2018.

[173] B. Tanç, H. T. Arat, E. Baltacıoğlu, and K. Aydın. Overview of the next quarter century vision of hydrogen fuel cell electric vehicles. *International Journal of Hydrogen Energy*, 44:10120–10128, 2019.

[174] C. Tarhan and M. A. Çil. A study on hydrogen, the clean energy of the future: Hydrogen storage methods. *Journal of Energy Storage*, 40:102676, 2021.

[175] Y. Tian, J. Zhang, and J. Morris. Modeling and optimal control of a batch polymerization reactor using a hybrid stacked recurrent neural network model. *Industrial & Engineering Chemistry Research*, 40:4525–4535, 2001.

[176] J. Timoshenko, H. S. Jeon, I. Sinev, F. T. Haase, A. Herzog, and B. Roldan Cuenya. Linking the evolution of catalytic properties and structural changes in copper–zinc nanocatalysts using operando EXAFS and neural-networks. *Chem. Sci.*, 11:3727–3736, 2020.

[177] T. L. Ting, N. W. Hasanah, F. S. Rohman, and D. Muhammad. Modeling and control of steam methane reforming process using model predictive control. *ASEAN Journal of Process Control*, 2:1–13, 2023.

[178] A. A. Topalov, I. Katsounaros, J. C. Meier, S. O. Klemm, and K. J. Mayrhofer. Development and integration of a LabVIEW-based modular architecture for automated execution of electrochemical catalyst testing. *Review of Scientific Instruments*, 82, 2011.

[179] A. Tran, A. Aguirre, H. Durand, M. Crose, and P. D. Christofides. CFD modeling of a industrial-scale steam methane reforming furnace. *Chemical Engineering Science*, 171:576–598, 2017.

[180] J. H. Tu. *Dynamic mode decomposition: Theory and applications*. PhD thesis, Princeton University, 2013.

[181] M. N. Uddin, V. V. Nageshkar, and R. Asmatulu. Improving water-splitting efficiency of water electrolysis process via highly conductive nanomaterials at lower voltages. *Energy, Ecology and Environment*, 5:108–117, 2020.

[182] UNFCCC. Paris agreement. Paris, France, 12 2015. United Nations.

[183] A. Ursua, L. M. Gandia, and P. Sanchis. Hydrogen production from water electrolysis: Current status and future trends. *Proceedings of the IEEE*, 100:410–426, 2011.

[184] V. Vapnik, S. E. Golowich, and A. Smola. Support vector method for function approximation, regression estimation and signal processing. *Advances in Neural Information Processing Systems*, 9:281–287, 1996.

[185] A. Wächter. Short tutorial: Getting started with IPOPT in 90 minutes. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2009.

[186] A. Wächter and L. T. Biegler. Line search filter methods for nonlinear programming: Local convergence. Technical report, Technical Report RC23033 (W0312-090), TJ Watson Research Center, Yorktown, 2003.

[187] A. Wächter and L. T. Biegler. Line search filter methods for nonlinear programming: Motivation and global convergence. *SIAM Journal on Optimization*, 16:1–31, 2005.

[188] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106:25–57, 2006.

[189] L. Wan, M. Zeiler, S. Zhang, Y. Le Cun, and R. Fergus. Regularization of neural networks using dropconnect. In *International Conference on Machine Learning*, pages 1058–1066. PMLR, 2013.

[190] K. Wang and P. Dave. Implementing an open-source sensor data ingestion, fusion, and analysis capabilities for smart manufacturing. *Manufacturing letters*, 2022.

[191] X. Wang, P. Ou, A. Ozden, S.-F. Hung, J. Tam, C. M. Gabardo, J. Y. Howe, J. Sisler, K. Bertens, F. P. García de Arquer, et al. Efficient electrosynthesis of n-propanol from carbon monoxide using a Ag–Ru–Cu catalyst. *Nature Energy*, 7:170–176, 2022.

[192] J. Wei and E. Iglesia. Isotopic and kinetic assessment of the mechanism of methane reforming and decomposition reactions on supported iridium catalysts. *Physical Chemistry Chemical Physics*, 6:3754–3759, 2004.

[193] J. Wei and E. Iglesia. Isotopic and kinetic assessment of the mechanism of reactions of ch$_4$ with co$_2$ or h$_2$o to form synthesis gas and carbon on nickel catalysts. *Journal of Catalysis*, 224:370–383, 2004.

[194] J. Wei and E. Iglesia. Mechanism and site requirements for activation and chemical conversion of methane on supported Pt clusters and turnover rate comparisons among noble metals. *The Journal of Physical Chemistry B*, 108:4094–4103, 2004.

[195] J. Wei and E. Iglesia. Reaction pathways and site requirements for the activation and chemical conversion of methane on Ru-based catalysts. *The Journal of Physical Chemistry B*, 108:7253–7262, 2004.

[196] J. Wei and E. Iglesia. Structural and mechanistic requirements for methane activation and chemical conversion on supported iridium clusters. *Angewandte Chemie International Edition*, 43:3685–3688, 2004.

[197] J. Wei and E. Iglesia. Structural requirements and reaction pathways in methane activation and chemical conversion catalyzed by rhodium. *Journal of Catalysis*, 225:116–127, 2004.

[198] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley. A data–driven approximation of the Koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25:1307–1346, 2015.

[199] R. H. Wiser and M. Bolinger. 2017 wind technologies market report. 2018.

[200] S. T. Wismann, J. S. Engbæk, S. B. Vendelbo, F. B. Bendixen, W. L. Eriksen, K. Aasberg-Petersen, C. Frandsen, I. Chorkendorff, and P. M. Mortensen. Electrified methane reform-

ing: A compact approach to greener industrial hydrogen production. *Science*, 364:756–759, 2019.

[201] S. T. Wismann, J. S. Engbaek, S. B. Vendelbo, W. L. Eriksen, C. Frandsen, P. M. Mortensen, and I. Chorkendorff. Electrified methane reforming: Elucidating transient phenomena. *Chemical Engineering Journal*, 425:131509, 2021.

[202] D. Wu, J. Zhang, M.-J. Cheng, Q. Lu, and H. Zhang. Machine learning investigation of supplementary adsorbate influence on copper for enhanced electrochemical CO_2 reduction performance. *The Journal of Physical Chemistry C*, 125:15363–15372, 2021.

[203] G. Wu, W. T. G. Yion, K. L. N. Q. Dang, and Z. Wu. Physics-informed machine learning for MPC: Application to a batch crystallization process. *Chemical Engineering Research and Design*, 192:556–569, 2023.

[204] Z. Wu, A. Aguirre, A. Tran, H. Durand, D. Ni, and P. D. Christofides. Model predictive control of a steam methane reforming reactor described by a computational fluid dynamics model. *Industrial & Engineering Chemistry Research*, 56:6002–6011, 2017.

[205] Z. Wu, J. Luo, D. Rincon, and P. D. Christofides. Machine learning-based predictive control using noisy data: Evaluating performance and robustness via a large-scale process simulator. *Chemical Engineering Research and Design*, 168:275–287, 2021.

[206] Z. Wu, D. Rincon, and P. D. Christofides. Process structure-based recurrent neural network modeling for model predictive control of nonlinear processes. *Journal of Process Control*, 89:74–84, 2020.

353

[207] Z. Wu, D. Rincon, J. Luo, and P. D. Christofides. Machine learning modeling and predictive control of nonlinear processes using noisy data. *AIChE Journal*, 67:e17164, 2021.

[208] Z. Wu, A. Tran, D. Rincon, and P. D. Christofides. Machine learning-based predictive control of nonlinear processes Part I: Theory. *AIChE Journal*, 65:e16729, 2019.

[209] X.-C. Xi, A.-N. Poo, and S.-K. Chou. Support vector regression model predictive control on a hvac plant. *Control Engineering Practice*, 15:897–908, 2007.

[210] R. Xia, S. Overa, and F. Jiao. Emerging electrochemical processes to decarbonize the chemical industry. *JACS Au*, 2:1054–1070, 2022.

[211] M. Xiao, C. Hu, and Z. Wu. Modeling and predictive control of nonlinear processes using transfer learning method. *AIChE Journal*, 69:e18076, 2023.

[212] J.-F. Xie, Y.-X. Huang, W.-W. Li, X.-N. Song, L. Xiong, and H.-Q. Yu. Efficient electrochemical CO_2 reduction on a unique chrysanthemum-like Cu nanoflower electrode and direct observation of carbon deposite. *Electrochimica Acta*, 139:137–144, 2014.

[213] K. Xie, A. Ozden, R. K. Miao, Y. Li, D. Sinton, and E. H. Sargent. Eliminating the need for anodic gas separation in CO_2 electroreduction systems via liquid-to-liquid anodic upgrading. *Nat. Commun.*, 13:1–9, 2022.

[214] S. Xie and J. Ren. Linearization of recurrent-neural-network-based models for predictive control of nano-positioning systems using data-driven Koopman operators. *IEEE Access*, 8:147077–147088, 2020.

[215] J. Xu and G. F. Froment. Methane steam reforming, methanation and water-gas shift: I. Intrinsic kinetics. *AIChE Journal*, 35:88–96, 1989.

[216] S. Yang, P. Navarathna, S. Ghosh, and B. W. Bequette. Hybrid modeling in the era of smart manufacturing. *Computers & Chemical Engineering*, 140:106874, 2020.

[217] Y. Yu and L. Jiang. Dynamic modeling on the MIMO system with linear programming support vector regression and its application. In *Proceedings of 2016 6th International Workshop on Computer Science and Engineering*, pages 710–714, Tokyo, 2016.

[218] Y. Yu, X. Si, C. Hu, and J. Zhang. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation*, 31:1235–1270, 2019.

[219] N. Zecevic and N. Bolf. Advanced operation of the steam methane reformer by using gain-scheduled model predictive control. *Industrial & Engineering Chemistry Research*, 59:3458–3474, 2020.

[220] H. Zhang, Z. Sun, and Y. H. Hu. Steam reforming of methane: Current states of catalyst design and process upgrading. *Renewable and Sustainable Energy Reviews*, 149:111330, 2021.

[221] T. Zhang, S. Li, and Y. Zheng. Implementable stability guaranteed Lyapunov-based data-driven model predictive control with evolving Gaussian process. *Industrial & Engineering Chemistry Research*, 61:14681–14690, 2022.

[222] Z. Zhang, E. W. Lees, S. Ren, B. A. Mowbray, A. Huang, and C. P. Berlinguette. Conversion

of Reactive Carbon Solutions into CO at Low Voltage and High Carbon Efficiency. *ACS Cent. Sci.*, 8:749–755, 2022.

[223] L. Zheng, M. Ambrosetti, F. Zaio, A. Beretta, G. Groppi, and E. Tronconi. Direct electrification of Rh/Al_2O_3 washcoated SiSiC foams for methane steam reforming: An experimental and modelling study. *International Journal of Hydrogen Energy*, 48:14681–14696, 2023.

[224] L.-T. Zhu, X.-Z. Chen, B. Ouyang, W.-C. Yan, H. Lei, Z. Chen, and Z.-H. Luo. Review of machine learning for hydrodynamics, transport, and reactions in multiphase flows and reactors. *Industrial & Engineering Chemistry Research*, 61:9901–9949, 2022.