2018

Dynamic Modeling, Predictive Control and Optimization of a Rapid Pressure Swing Adsorption System

Matthew Urich
Lehigh University

DYNAMIC MODELING, PREDICTIVE CONTROL AND OPTIMIZATION OF A RAPID PRESSURE SWING ADSORPTION SYSTEM

by

Matthew D. Urich

A Dissertation Presented to the Graduate and Research Committee
of Lehigh University
in Candidacy for the Degree of
Doctor of Philosophy

in
Chemical Engineering

Lehigh University
Bethlehem, PA

May 2018

Approved and recommended for acceptance as a dissertation in partial fulfillment of requirements for the degree of Doctor of Philosophy.

Date

Mayuresh V. Kothare
(Dissertation Advisor)

Accepted Date

Committee members:

Hugo Caram

William Luyben

Nader Motee

Mark Snyder

Acknowledgments

I would first like to give my sincere thanks to my advisor, Dr. Mayuresh V. Kothare, for his guidance, motivation, advice and support throughout my study and research while at Lehigh University. His thoughts and advice during my research helped me to grow as an independent researcher. I thoroughly enjoyed working with him during my doctoral training.

I am thankful for the continued support and encouragement from my family and friends during my studies. Completion of my degree would not have been possible without their support.

I would like to thank Theodore Bowen from the Dept. of Electrical Engineering at Lehigh University for his help in teaching a chemical engineer electricity, and his help in Raspberry Pi. I gratefully acknowledge financial support from the Department of Chemical and Biomolecular Engineering, Pennsylvania Life Sciences Greenhouse, and the National Science Foundation.

Contents

Acknowledgments **iii**

Abstract **1**

1 Introduction **3**

1.1 Rapid Pressure Swing Adsorption 3

1.2 COPD and Medical Oxygen Concentrators 4

1.3 PSA Modeling and Control in the Literature 7

1.3.1 Model Predictive Control 10

1.4 Dissertation Overview 12

2 Operation and Modeling of the RPSA System **13**

2.1 Motivation 13

2.2 Novel, Single Bed RPSA Device 14

2.3 RPSA Plant Model 15

2.4 Simulation Strategies for the RPSA System 20

2.5 Open Loop Dynamic Response 22

2.6 Closing Remarks 25

3 Model Predictive Control for RPSA MOCs 26

3.1 Motivation 26

3.2 Control Challenges and Feedback Strategy 27

3.3 Sub-Space System Identification 29

3.3.1 Identification of Baseline Operation 29

3.3.2 PRBS Signal Design and Optimization 30

3.3.3 Identification Results 34

3.4 Multivariable Model Predictive Controller 36

3.5 Controller Evaluation 40

3.5.1 Disturbance Rejection 40

3.5.2 Comparison to PID Control 47

3.5.3 Set Point Tracking 49

3.6 Stability of the MPC Algorithm 49

3.6.1 Demonstrating CSS Behavior in the RPSA Model 52

3.7 Closing Remarks 53

4 Piece-wise Linear MPC for other RPSA Applications 55

4.1 Motivation 55

4.2 Multi-Model Predictive Control Strategy for Nonlinear Systems . . . 56

4.3 Model Identification for Operating Space 58

4.3.1 Identification of Operating Points 59

4.3.2 Sub-Space Identification using PRBS Signals 60

4.3.3 Piece-Wise Linear Model for the RPSA System 62

4.4 Multi-Model Predictive Controller 64

4.4.1 Controller Formulation 64

4.4.2 Switching Logic and Boundary Considerations 65

4.5 M-MPC Performance and Evaluation 67

4.5.1 Set Point Tracking 67

4.5.2 Disturbance Rejection 70

4.6 Extensions of Piece-Wise Linear MPC 70

4.7 Closing Remarks 72

5 Implementation of an Embedded MPC for a MOC Device 73

5.1 Motivation 73

5.2 Design and Operation of the Lab-Scale MOC Device 74

5.2.1 Hardware Requirements of the Device 77

5.3 Model Predictive Control Algorithm 77

5.3.1 Multivariable MPC Formulation 79

5.3.2 Modeling and System Identification 81

5.4 Embedded Feedback Controller 84

5.4.1 Implementation of the MPC Algorithm 84

5.4.2 Synchronization of Controller Tasks 87

5.5 Performance and Evaluation of the MOC System 89

5.5.1 Disturbance Rejection 89

5.5.2 Set Point Tracking 91

5.6 Implementation of the M-MPC Algorithm 91

5.6.1 Operating Regions and Linearization Points 93

5.6.2 Comparison of Linear Identified Models 94

5.6.3 Implemented M-MPC Algorithm 95

5.6.4 Evaluation of the M-MPC 98

5.7 Closing Remarks 98

6 Steady State Optimization of MOC Devices 101

6.1 Motivation 101

6.2 Steady State Optimization of the RPSA Cycle 102

6.2.1 Formulating Model-Free Optimization Problems 105

6.2.2 Real-Time Optimization using Embedded Hardware 107

6.3 MOC Optimization Case Studies 109

6.3.1 Case 1: Baseline Identification for MPC Design 109

6.3.2 Case 2: Product Flow Rate Maximization 111

6.3.3 Case 3: Power Consumption Minimization 112

6.4 Future Directions in MOC Optimization 113

6.5 Advances in RPSA Modeling for Optimization 114

6.6 Closing Remarks 117

7 Conclusions and Future Directions 119

7.1 Model Predictive Control for the RPSA System 119

7.2 Future Directions 121

List of Publications 123

Bibliography 126

Biographcial Information 134

List of Tables

1.1 Summary of RPSA Modeling, Control and Optimization Literature Review 9

2.1 Boundary and Initial Conditions for the RPSA Model 18

3.1 Summary of the Chosen Baseline Operating Point for the RPSA MOC 30

3.2 Summary of PRBS-type Simulations for System Identification 33

4.1 Linear Model Operating Points and Identification Results 60

4.2 Summary of PRBS Signals Used to Identify the Piece-Wise Linear Model for the RPSA System 61

5.1 Summary of Identified Model Linearization Point 83

5.2 Summary of Operating Region Linearization Points and Boundaries . 93

6.1 Optimization of RPSA Cycle Timing for Various O_2 Compositions . . 111

List of Figures

1.1 Lungs of a COPD Patient 5

1.2 Various Sized MOC Devices: An ultra-small portable unit (left), a larger stationary unit (middle), a very large MOC which can also fill gas cylinders (right). [1] 6

1.3 A Representation of a MPC Control Problem 10

2.1 Description of the 4-Step RPSA Cycle 15

2.2 Simulation Strategy for the RPSA Model 21

2.3 Open Loop Dynamics in the RPSA System 23

2.4 Open Loop Block Diagram for the RPSA System 24

3.1 Effect of Adsorption Time on Storage Tank Composition and Pressure 28

3.2 PRBS-type Simulations for Sub-Space Model Identification 32

3.3 Sub-Space Identification using PRBS Signals 35

3.4 Block Diagram of the MPC Algorithm 36

3.5 Effect of Integral Action on MPC Performance 39

3.6 Changes in Feed Gas Temperature 41

3.7 Simultaneous Feed and Product Flow Rate Changes 43

3.8 Fluctuations in Product Flow Rate 45

3.9 Decrease in Adsorber Bed Capacity 46

3.10 A Comparison between MPC and PID Control 48

3.11 Set Point Tracking using MPC 50

4.1 RPSA Operating Regions and Points 59

4.2 PRBS Simulation for Operating Region 4 62

4.3 Comparison of the Piece-Wise Linear Model using in M-MPC 64

4.4 M-MPC Set Point Tracking Scenario 1 68

4.5 M-MPC Set Point Tracking Scenario 2 69

4.6 M-MPC Disturbance Rejection Scenario 71

5.1 Lab-Scale MOC Device with Embedded Controller 75

5.2 Embedded Controller Hardware Schematic 76

5.3 Components of the Embedded Controller 78

5.4 Open-Loop Block Diagram of the MOC Device 79

5.5 Sub-Space Identification using PRBS Input Signals 85

5.6 Embedded Controller Operation Flow Chart 88

5.7 Disturbance Rejection using the Embedded MPC 90

5.8 Embedded MPC Set Point Tracking Case Study 92

5.9 A Comparison of Identified Models 95

5.10 Embedded M-MPC Algorithm Flowchart 97

5.11 Evaluation of the M-MPC using Set Point Tracking Scenarios 99

6.1 RPSA Pressure Profiles at CSS 103

6.2 Steady State Optimization with Embedded Hardware 108

Abstract

Rapid Pressure Swing Adsorption (RPSA) is a gas separation technology with an important commercial application for Medical Oxygen Concentrators (MOCs). MOCs use RPSA technology to produce high purity oxygen (O_2) from ambient air, and provide medical oxygen therapy to Chronic Obstructive Pulmonary Disease (COPD) patients. COPD is a lung disease which prevents O_2 from entering a patient's blood, and reduces the blood oxygen level. The standard therapy for COPD is to provide the patient with high purity ($\sim 90\%$) O_2. MOCs have become more popular than traditional O_2 gas cylinders due to their improved safety, and smaller device size and weight. The MOC market is growing rapidly and was expected to grow from $358 million in 2011 to $1.8 billion in 2017. Recently, a novel, single-bed MOC design was developed and tested to further reduce the size and weight of the device, and provide a continuous supply of O_2 to the patient. This single-bed design uses a complex RPSA cyclic process with many nonlinear effects. Flow reversals, discrete valve switching, nonlinear adsorption effects, and complex fluid dynamics all make operating the RPSA system very challenging. Feedback control is necessary in a final commercial product to ensure the device operates reliably, but feedback control of PSA systems is not well studied in the current literature.

In this work, a study of dynamic modeling, predictive control and optimization

of this single-bed RPSA device is presented. A detailed, nonlinear plant model of the RPSA device is used to study the dynamics of the system as well as design a Model Predictive Controller (MPC) for the RPSA system. The plant model is a fully coupled, nonlinear set of Partial and Ordinary Differential Equations (PDEs and ODEs) which act as a representation of reality when design and evaluating the MPC. A sub-space model identification technique using Pseudo-Random Binary Sequence (PRBS) input signals generate a linear model which reduces the computational cost of MPC, and allows the algorithm to be implemented as an embedded controller for the RPSA device. The multivariable MPC *independently* manipulates the RPSA cycle step durations to control both the product composition and pressure. This MPC strategy was designed and tested in simulation before being implemented on a lab-scale device.

The MPC is implemented onto a lab-scale MOC prototype using Raspberry Pi hardware, and evaluated using several MOC-relevant disturbance scenarios. The MPC is also expanded using piece-wise linear modeling to improve the performance of an RPSA device for other concentrated O_2 applications. The embedded MPC features a convex quadratic optimization problem which is solved in real time using online output measurements. Additional hardware in the embedded controller operates the RPSA cycle and implements control actions supplied by the MPC.

Design and optimization of RPSA systems remains an active area of research, and many PSA models have been used to optimize RPSA cycles in simulation. In this work, a model-free steady state optimization approach using the embedded hardware is presented which does not require a detailed process model, and uses experimental data and a nonlinear solver to optimize the RPSA operation given various objectives.

Chapter 1

Introduction

1.1 Rapid Pressure Swing Adsorption

Pressure swing adsorption (PSA) is a common industrial and commercial gas separation technology which uses an adsorbent material and a cyclic pressure swing to purify gas mixtures. Many industrial applications exist, including hydrogen production, water removal, CO_2 capture, and air separation. PSA relies on phenomena known as gas adsorption and desorption. Adsorption is the adhesion of a gaseous species onto a solid surface due to an attractive force between the gas and solid, and desorption is the removal of the gas species from the solid surface. PSA is a cyclic process, where multiple adsorbent beds are synchronized to continuously separate a feed gas mixture by increasing and decreasing the gas pressure in the adsorbent column. There are many different types of PSA systems, which depend, in general, on the adsorbent material, adsorption and desorption pressures, feed gas mixtures and others. A subset of PSA systems is called Rapid Pressure Swing Adsorption (RPSA) which is generally categorized as a PSA system with a total cycle time of $<$10 seconds. RPSA has a significant commercial application in Medical Oxygen Concentrators (MOCs) which have become very popular medical devices in recent years.

1.2 COPD and Medical Oxygen Concentrators

Chronic Obstructive Pulmonary Disease (COPD) is a lung disease which prevent O_2 from transferring to the blood stream inside a patient's lungs. COPD causes a film to develop in a patient's lung which prevent the mass transfer from occuring, and reduces the patient's blood oxygen level. A common therapy for many COPD patients is long-term oxygen therapy, where the patient is continuously supplied medical-grade high purity oxygen (typically $\sim 90\%$ O_2). Fig. 1.1 compares the lungs of a healthy patient to one with COPD. In the past, patients would utilize high pressure O_2 cylinder to deliver this therapy, but there are several disadvantages to using gas cylinders from both safety and quality of life perspectives. A gas cylinder must contain a very high pressure to supply enough gas for an extended period, which could be several hours depending on the patient's acitivity level. These cylinders have the potential to rupture and cause severe injury to anyone standing close to them. The cylinders are also heavy, bulky items which are difficult to carry and transport. An elderly COPD patient is typically physically limited, and they cannot carry even moderately heavy loads. MOCs have become a popular alternative to gas cylinders due to their reduced size, weight and improved safety considerations.

MOCs produce high purity O_2 from ambient air using RPSA technology. There are many commerical devices available in the current market, and demand for MOCs is expected to grow [1]. MOCs range in size depending on the patient's required flow rate of high purity O_2, and their activity needs. Stationary units can produce

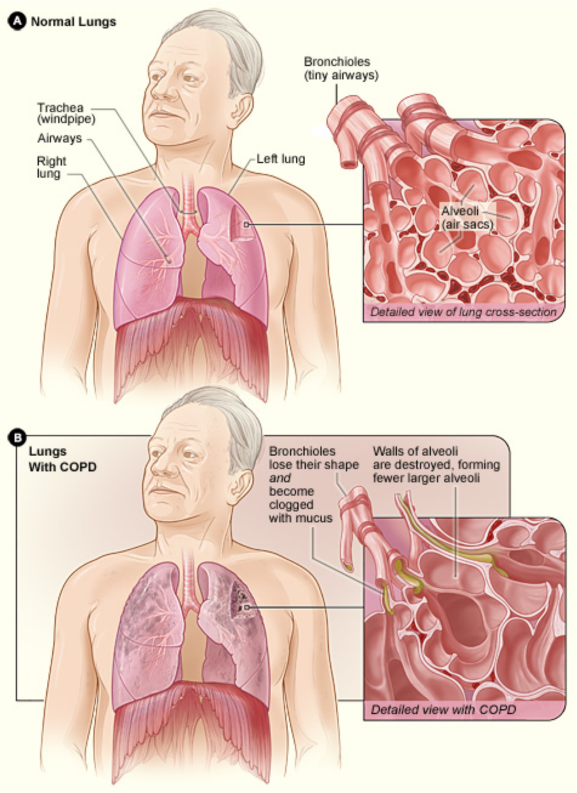

A Normal Lungs
Bronchioles (tiny airways)
Trachea (windpipe)
Airways
Right lung
Left lung
Alveoli (air sacs)
Detailed view of lung cross-section
B Lungs With COPD
Bronchioles lose their shape *and* become clogged with mucus
Walls of alveoli are destroyed, forming fewer larger alveoli
Detailed view with COPD

Figure 1.1: Lungs of a COPD Patient: COPD causes a deterioration of a patient's lungs, and prevents O_2 from transferring into the bloodstream.[2]

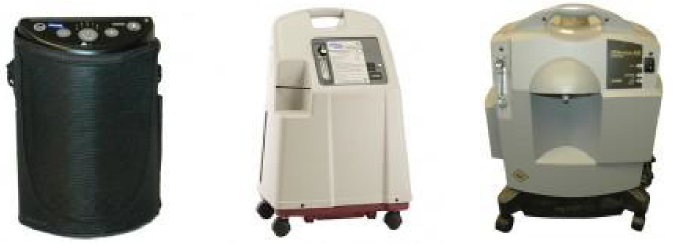


Figure 1.2: Various Sized MOC Devices: An ultra-small portable unit (left), a larger stationary unit (middle), a very large MOC which can also fill gas cylinders (right). [1]

1-5+ liters per min (SLPM) of O_2 and are large, immobile units patients use in their homes and while sleeping. Ambulatory units are smaller, produce approximately 1-3 SLPM O_2, and can typically be used on a cart or carried as a backpack. The newest and most popular designs are ultra-small portable units which produce $\sim$ 1 SLPM and can be carried with a shoulder strap. Fig. 1.2 shows examples of different sized MOCs. The market for MOC devices is expected to grow significantly, especially in foreign markets, due to increased diagnosis of COPD in developing countries. The MOC market was expected to grow from $358 million in 2011 to $1.8 billion by 2017 [1].

Current MOCs use a two-column RPSA design to produce high purity O_2. Recently, a novel, single bed MOC prototype was developed to reduce the overall size and weight of MOC devices [3]. These devices are complex systems with unsteady, cyclic operation and many moving parts which must operate in a precise manner to produce the required O_2 product. Feedback control is required to operate and monitor MOC devices, but control of PSA systems is not well represented in the current literature.

1.3 PSA Modeling and Control in the Literature

PSA systems are diverse in their applications, design and operation, and many different groups study PSA systems. Recent literature includes models of varying detail. Table 1.1 gives a summary of recent publications in PSA modeling, control and optimization. In most studies, the PSA model is used to optimize the system based on some given objective: cycle timing, product purity, energy consumption, etc. These optimization studies are performed off-line as part of the design process in simulation. The required detail of the PSA model varies depending on this optimization objective, and many nonlinear physical effects such as gas-solid heat transfer are neglected because they increase the computational cost of the model. In this work, a process model which makes as few assumptions as possible is used to capture the dynamic response of the RPSA system in response to several possible disturbances. Feedback control is not well documented in the literature, but plays an essential role in PSA system operation. Only, two studies have considered feedback control. Sun, et al presented a SISO PID controller implemented into a three-column vacuum PSA model which was then optimized for various factors. Khajuria, et al developed a SISO MPC for an industrial sized, four column PSA system. This MPC controls the product purity by manipulating only the adsorption step duration. In both of these studies, only a single cycle step duration is manipulated to control the PSA system. This control strategy is not ideal because the controller cannot take advantage of the coupled relationship all the cycle step durations have on the PSA dyanmics.

In this work, a multivariable MPC is developed for a novel RPSA design. The unique design requires a control strategy not found in other industrial sized PSA systems. Enhanced performance and online optimization of cycle step timing is achieved by the MPC independently manipulating all four cycle step durations.

Table 1.1: Summary of RPSA Modeling, Control and Optimization Literature Review

Reference	PSA Type, Beds	Isotherm, Adsorbent	Mass Balance	Energy Balance	Momentum Balance	Controller Type	Control Variable	Manipulated Variable	Optimization Objective
[4]	VPSA	Langmuir, Coal	LDF, ADPF	Equilibrium	Ergun	–	–	–	–
[5]	VPSA, 3	Langmuir, Carbon	LDF, ADPF	Equilibrium	Ergun	PID	Product Purity	Adsorption Time	Power Consumption, Product Purity, Recovery, Productivity
[6]	PSA, 2	Langmuir, Zeolite	LDF, IPF	–	Ergun	–	–	–	–
[7]	VPSA,4	Langmuir	LDF, ADPF	Equilibrium	Ergun	–	–	–	–
[8]	PTSA	Dubinin-Astakhov, Zeolite	LDF, ADPF	–	Ergun	–	–	–	–
[9]	PSA, 2	Langmuir, Carbon	LDF, IPF	Equilibrium	Ergun	–	–	–	–
[10]	PSA, 4	Langmuir, Carbon	LDF, ADPF	Equilibrium	Ergun	–	–	–	Product Purity
[11]	PSA, 2	Zeolite	–	–	–	–	–	–	–
[12]	V/PSA	Langmuir, Zeolite	LDF, ADPF	Equilibrium	Ergun	–	–	–	Product Costs
[13]	PSA, 2	Langmuir, Multiple	LDF, ADPF	Equilibrium	Ergun	–	–	–	Energy Consumption
[14, 15]	PSA, 4	Langmuir, Carbon	LDF, ADPF	Equilibrium	Ergun	MPC	Product Purity	Adsorption Time	–
[16]	PSA	Langmuir, Charcoal	LDF, IPF	Equilibrium	–	–	–	–	Optimal Cycle
[17]	PSA, 2	Langmuir, Zeolite	LDF, ADPF	Equilibrium	Ergun	–	–	–	Recovery, Profit
[18]	PSA	Langmuir, Zeolite	LDF, IPF	Equilibrium	–	–	–	–	Optimal Cycle
[19]	VPSA	Zeolite	LDF, ADPF	Equilibrium	–	–	–	–	Recovery
[20]	VPSA	Langmuir, Zeolite	LDF, ADPF	Equilibrium	Ergun	–	–	–	Optimal Cycle
[21]	PSA	Langmuir	LDF, ADPF	Equilibrium	Ergun	–	–	–	Optimal Cycle
[22]	VPSA	Langmuir	LDF, ADPF	Equilibrium	Ergun	–	–	–	Optimal Cycle
[23]	PSA	Langmuir, Zeolite	LDF, ADPF	Equilibrium	Darcy	–	–	–	Optimal Cycle
[24]	PSA	Langmuir	LDF, ADPF	Isothermal	–	–	–	–	Product Purity, Cost
[25]	RPSA, 1	Langmuir, Zeolite	LDF, ADPF	Non-isothermal	Ergun	MPC	Product Purity, Pressure	Cycle Step Durations	–

LDF-Linear Driving Force Model ADPF-Axially Dispersed Plug Flow IPF-Ideal Plug Flow

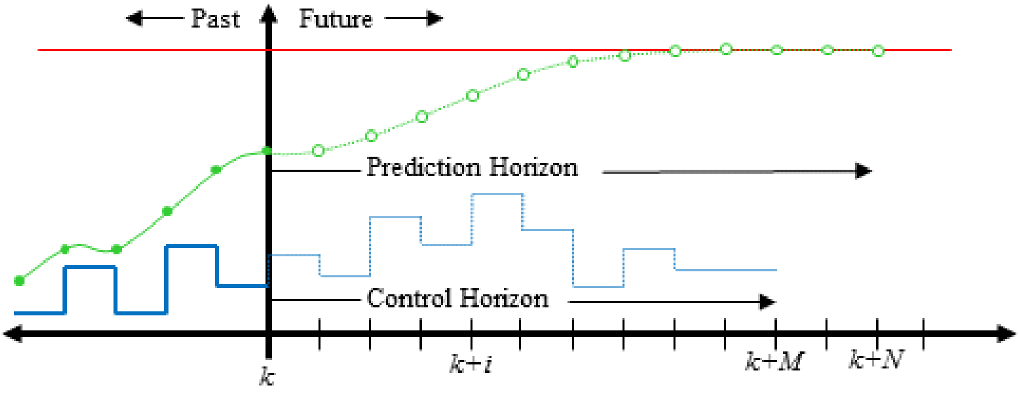


Figure 1.3: A Representation of a MPC Control Problem: At every sample point k, the MPC uses past measurements to estimate the current state of the system. A process model predicts the system dynamics over a prediction horizon in response to a chosen set of control actions. The MPC iteratively chooses the control movements over a set control horizon until a give objective function is minimized. The first chosen control movement is implemented, and the calculation is repeated at the next sample point.

1.3.1 Model Predictive Control

Model Predictive Control (MPC) is a type of optimal control which uses a process model and a given optimization problem to make ”optimal” control decisions [26]. A summary of MPC is shown in Fig. 1.3. At each sample point, k, the MPC uses past measurements to estimate the current system states. A process model is used to predict the system dynamic response over a set prediction horizon to chosen manipulated input changes. The MPC chooses these input changes, or control actions, over a given control horizon to yield an ”optimal” system response according to a given optimization objective. The MPC implements the first control action, and the calculation is repeated at the next sample point, $k + 1$.

MPC has many advantages which have made it popular in many control applications. Utilizing a process model directly in control decisions means the MPC can anticipate the effect its control actions will have on the system. Depending on the complexity of the process model, the MPC can handle multi-input, multi-output (MIMO) control problems with which traditional PID controllers would struggle. Objective functions can be tailored to whatever control objective is required by the particular system, and constraints to control decisions can be explicitly stated and enforced in all calculations. In the RPSA system, constraints are important to ensuring the RPSA cycle always operates properly.

While MPC has many advantages for optimal control, there are disadvantages to using it in real applications. Compared to PID control, MPC has a much higher computational cost which is related to the complexity of the process model and the type of optimization problem used. A major theme of this work is the trade-off between model complexity and computational cost, because the computational cost and solving time can preclude implementing MPC into embedded controllers. In general, MPC also requires more development time compared to PID control, and a MPC for one system is not generally transferable to another.

In this work, MPC was chosen as the best choice for using feedback control in the RPSA system. MPC has the ability to account for multivariable interactions between the control variables and manipulated inputs prevalent in the RPSA system, and constraints on the inputs can be explicitly enforced. The disadvantages of MPC can be mitigated in the process model design and optimization formulations. These issues are discussed in detail later.

1.4 Dissertation Overview

The dissertation is broken into three main objectives: design and test a Model Predictive Controller (MPC) for the MOC device in simulation using a process model, implement the designed MPC onto the lab-scale MOC prototype using embedded control hardware, and investigate a model-free steady state optimization technigue for designing the RPSA process using embedded hardware.

In Chapter 2, a detailed, nonlinear RPSA process model is presented to be used to design and evaluate the MPC. A unique modeling strategy is discussed which allows sequential RPSA cycles to be simulated, and different RPSA-specific dynamic behavior is demonstrated. In Chapter 3, the MPC algorithm, control strategy, design procedure and final evaluation are presented using the detailed process model as a representation of the real RPSA device. In Chapter 4, an extension of the presented MPC using a piece-wise linear model which allows the RPSA system to produce a wide range of product compositions. Chapters 2-4 detail the design and testing of the MPC algorithm in simulation.

In Chapter 5, the MPC is implemented as an embedded feedback controller onto a lab-scale RPSA device. Implementation challenges and solutions are discussed, as well as modifications to the MPC not seen in simulation. The extension of MPC using piece-wise linear models is also implemented and experimentally demonstrated. Finally, in Chapter 6, the same embedded hardware is used to demonstrate a model-free steady state optimization technique to aid in RPSA process design.

Chapter 2

Operation and Modeling of the RPSA System

2.1 Motivation

This chapter, the operation, open-loop dynamics and dynamic model for the single-bed RPSA system is presented. The operation of the RPSA system, which features a single adsorber column and product storage tank, helps to identify the control-relevant variables as well as potential manipulated inputs for the feedback controller. The plant model for the RPSA system is the most detailed model in the current literature, and its complexity is required to capture dynamic responses as accurately as possible. The plant model will be used in future chapters to design and evaluate a feedback controller for the RPSA system. Because of its cyclic nature, the RPSA system is extremely difficult to solve numerically, and a simulation strategy is discussed which integrates the plant model with feedback control calculations to complete the closed-loop system. A discussion of the dynamic behavior in the RPSA system finalizes the multivariable control problem, and motivates the choice of model predictive control for the RPSA system.

2.2 Novel, Single Bed RPSA Device

A novel, single bed RPSA device was developed previously for MOC applications [3, 27]. The system features a patented single adsorber column concentrically inserted inside a product storage tank [28]. A LiLSX zeolite material selectively adsorbed nitrogen from ambient air. The device uses a 4-step Skarstrom-like [29] RPSA cycle to continuously produce ∼90% oxygen. A schematic of the RPSA device and 4-step cycle is shown in Fig. 2.1. During the pressurization step, the column is pressurized to a super atmospheric pressure (P_a) with feed air. In the adsorption step, feed air is continuously supplied to the column, and high purity oxygen product leaves the column and enters the storage tank until the adsorbent bed is saturated with nitrogen. During blow down, nitrogen is desorbed from the adsorbent bed by reducing the column pressure to atmosphere. The nitrogen gas is discarded. In the purge step, the column is back purged using some of the high purity oxygen in the storage tank. High purity product is continuously supplied to the patient during all four steps. After the purge step, fresh compressed feed air is supplied and the RPSA repeats. The duration of each RPSA cycle step can be manipulated, and has a coupled, nonlinear effect on the RPSA system dynamics.

A process model for the RPSA system to develop and the test potential feedback control strategies. The process model must include a detailed model of both the adsorber column and storage tank to best simulate the dynamic response of the RPSA system to changes in cycle step durations and process disturbances. The

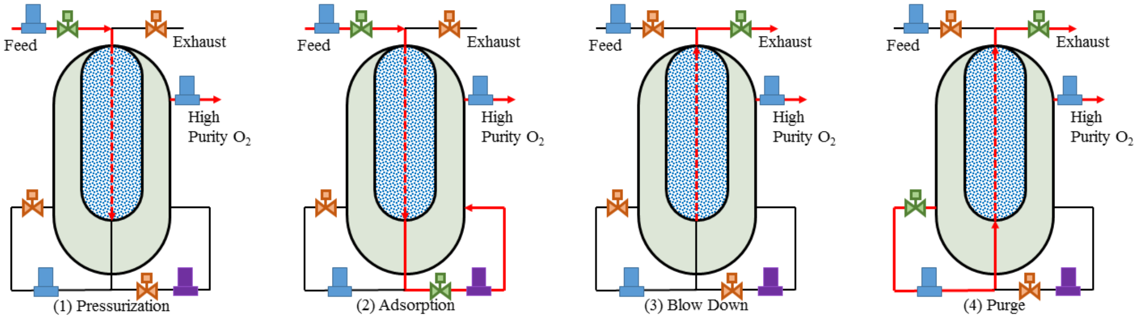


Figure 2.1: Description of the 4-Step RPSA Cycle: (1) Pressurization: The adsorber column (blue) is pressurized with feed air to super-atmospheric conditions. (2) Adsorption: Feed air is continuously added to the column and high purity O_2 enters the storage tank (green) from the product end. (3) Blow Down: The nitrogen in the adsorber column is exhausted to atmosphere and discarded. (4) Some of the high purity O_2 from the storage tank is used to back purge column and clean nitrogen from the voids. High purity oxygen is delivered to the patient continuously from the product line in all four steps.

process model used in this work is a set of coupled nonlinear partial and ordindary differential equations (PDEs and ODEs).

2.3 RPSA Plant Model

The RPSA plant model has two components: a adsorber column model and a storage tank model. The column model makes several assumptions:

- The ideal gas law holds
- Radial effects are negligible
- The column is non-isothermal and adiabatic
- Pressure drop follows the Ergun Equation

- Mass transfer between the gas and solid phases follows the Linear Driving Force (LDF) model

- Langmuir isotherms describe the adsorption equilibrium

- Heat transfer occurs between the gas and solid phases

- Feed air is a 79% N_2 and 21% O_2 binary mixture

A total and component mass balance in the adsorber column, gives

$$\frac{\partial \rho_g}{\partial t} = \frac{\bar{\epsilon}}{\epsilon} D_L \frac{\partial^2 \rho_g}{\partial z^2} - \frac{1}{\epsilon}\frac{\partial Q}{\partial z} - \frac{\rho_b}{\epsilon} \sum_{i \epsilon \{O_2, N_2\}} \frac{\partial n_i}{\partial t} \tag{2.1}$$

$$\frac{\partial(\chi_i \rho_g)}{\partial t} = \frac{\bar{\epsilon}}{\epsilon} D_L \frac{\partial^2(\chi_i \rho_g)}{\partial z^2} - \frac{1}{\epsilon}\frac{\partial(\chi_i Q)}{\partial z} - \frac{\rho_b}{\epsilon}\frac{\partial n_i}{\partial t} \tag{2.2}$$

where ρ_g is the total gas density, Q is the molar flux through the column, n_i is the fraction of species i adsorbed in the solid phase, and χ_i is the mole fraction of species i in the gas phase. ϵ and $\bar{\epsilon}$ are the helium and bed void fractions respectively, and D_L and ρ_b are effective axial mass dispersion coefficient and the bulk density, respectively. z is the axial distance through the adsorber column. Mass transfer between the gas and solid phases is modeled by the LDF model using experimentally determined Langmuir isotherms:

$$\frac{\partial n_i}{\partial t} = k_i(n_i^\infty - n_i) \tag{2.3}$$

where k_i and n_i^∞ are the LDF mass transfer coefficient of species i and the equilibrium fraction of species i adsorbed, respectively. Pressure drop across the column is modeled by Ergun's Equation:

$$\frac{\partial P}{\partial z} = -\frac{150\mu}{\rho_g d_p}\frac{(1-\bar{\epsilon})^2}{\bar{\epsilon}^3}Q - \frac{1.75M_g}{\rho_g d_p^2}\frac{1-\bar{\epsilon}}{\bar{\epsilon}^3}Q^2 \tag{2.4}$$

where μ, d_p and M_g are the viscosity, particle diameter and gas mixture molecular weight, respectively, and P is the column pressure. The energy balances for the gas phase (2.5) and solid phase (2.6) are

$$\begin{gathered}\epsilon C_{p_g}\frac{\partial(\rho_g\theta_g)}{\partial t} = \bar{\epsilon}C_{p_g}D_g\frac{\partial^2(\rho_g\theta_g)}{\partial z^2} - C_{p_g}\frac{\partial(Q\theta_g)}{\partial z} + h_{gs}a(\theta_s - \theta_g),\\ \theta_g = T_g - T_f\end{gathered} \tag{2.5}$$

$$\begin{gathered}\rho_b C_{p_s}\frac{\partial\theta_s}{\partial t} = \rho_b\sum_{i\epsilon\{O_2,N_2\}} q_i\frac{\partial n_i}{\partial t} - h_{gs}a(\theta_s - \theta_g),\\ \theta_s = T_s - T_f\end{gathered} \tag{2.6}$$

where θ_g, θ_s, T_g, T_s and T_f are the differential gas temperature, differential solid temperature, gas temperature, solid temperature and feed temperature, respectively. C_{p_g}, D_g, h_{gs} and a are the gas specific heat, the effective axial heat dispersion coefficient, the gas-solid heat transfer coefficient and the effective surface area, respectively. C_{p_s} and q_i are the solid phase heat capacity and the heat of adsorption for species i, respectively.

Each RPSA cycle step uses Eqns 2.1-2.6 to describe the column dynamics, but flow reversals are what define the RPSA cycle. Each cycle step has a unique set of boundary conditions which change to reflect each step's behavior. Table 2.1 gives a comprehensive list of all boundary conditions in the RPSA plant model. α_i, β_i are coefficients of a polynomial valve equation, and G is the purge to feed ratio.

The storage tank model is a pair of ODEs which make two further assumptions,

- The storage tank pressure is a function of time only

Table 2.1: Boundary and Initial Conditions for the RPSA Model

RPSA Cycle Step	$z = 0$	$z = L$
Pressurization	$-D_L \frac{\partial(\chi_i \rho_g)}{\partial z} = \frac{Q}{\bar{\epsilon}}(\chi_{i,f} - \chi_i)$	$\frac{\partial(\chi_i \rho_g)}{\partial t} = 0$
	$-D_g \frac{\partial(\rho_g \theta_g)}{\partial z} = \frac{Q}{\bar{\epsilon}}(\theta_f - \theta_g)$	$\frac{\partial(\rho_g \theta_g)}{\partial z} = 0$
	$Q = \sum_{i=1}^{6} \alpha_i (P_f - P)^{7-i}$	$Q = 0$
Adsorption	$-D_L \frac{\partial(\chi_i \rho_g)}{\partial z} = \frac{Q}{\bar{\epsilon}}(\chi_{i,f} - \chi_i)$	$\frac{\partial(\chi_i \rho_g)}{\partial t} = 0$
	$\frac{\partial P}{\partial z} = 0$	$P = P_a$
	$Q = Q_f$	$\frac{\partial Q}{\partial z} = 0$
	$-D_g \frac{\partial(\rho_g \theta_g)}{\partial z} = \frac{Q}{\bar{\epsilon}}(\theta_f - \theta_g)$	$\frac{\partial(\rho_g \theta_g)}{\partial z} = 0$
Blow Down	$\frac{\partial(\chi_i \rho_g)}{\partial t} = 0$	$\frac{\partial(\chi_i \rho_g)}{\partial t} = 0$
	$\frac{\partial(\rho_g \theta_g)}{\partial z} = 0$	$\frac{\partial(\rho_g \theta_g)}{\partial z} = 0$
	$Q = \sum_{i=1}^{6} \beta_i (P - P_{atm})^{7-i}$	$Q = 0$
Purge	$\frac{\partial(\chi_i \rho_g)}{\partial t} = 0$	$D_L \frac{\partial(\chi_i \rho_g)}{\partial z} = G \frac{Q}{\bar{\epsilon}}(\chi_{i,T} - \chi_i)$
	$P = P_{atm}$	$\frac{\partial P}{\partial z} = 0$
	$\frac{\partial Q}{\partial z} = 0$	$Q = -G Q_f$
	$\frac{\partial(\rho_g \theta_g)}{\partial z} = 0$	$-D_g \frac{\partial(\rho_g \theta_g)}{\partial z} = G \frac{Q}{\bar{\epsilon}}(\theta_f - \theta_g)$
Initial Conditions	$\chi_i(z,0) = \chi_{i,f};\ P(z,0) = P_T(z,0) = P_a;\ \theta_g(z,0) = \theta_s(z,0) = T_f - T_{atm}$	
	$Q(z,0) = 0;\ \chi_{O2,T}(0) = 0.90$	

- Storage tank temperature is constant at atmospheric conditions, and not modeled

The storage tank model includes a total and component mass balance.

$$\frac{dP_T}{dt} = \frac{F}{V_T}(\left. P\right|_{z=L,t} - P_T) - \frac{F_P}{V_T}(P_T - P_{atm}) \tag{2.7}$$

$$\frac{d(\chi_{i,T}P_T)}{dt} = \frac{F}{V_T}(\left.(\chi_i P)\right|_{z=L,t} - \chi_{i,T}P_T) - \frac{F_P}{V_T}\chi_{i,T}(P_T - P_{atm}) \tag{2.8}$$

Where P_T and $\chi_{i,T}$ are the total pressure and mole fraction of species i in the storage tank. V_T, F, F_P and P_{atm} are the storage tank volume, flow rate of gas between the column and tank, product flow rate and atmospheric pressure, respectively. Boundary and initial conditions for the model are summarized in Table 2.1. A similar model for a helium nitrogen mixture can be found in [30].

The challenge in modeling the RPSA system is sequentially solving the different cycle steps. All steps use the same model equations, but, in the column model, the boundary conditions change to reflect flow reversals. In the tank model, the parameter F changes to reflect different flows between the column and tank. During the pressurization and blow down steps, there is no flow between the column and tank, and $F = 0$ for these steps. In the adsorption step, gas flows from the column to tank according to the time-dependent solution of Eqns. 2.1-2.6. The flow between the column and tank is given as,

$$F = \left.\frac{QART_g}{P}\right|_{z=L,t} \tag{2.9}$$

During the purge step, the flow from the tank to the column is a constant fraction of the feed conditions, and the flow is given as,

$$F = -G\frac{Q_f ART_f}{P_f} \tag{2.10}$$

The RPSA model is difficult to solve, and a detailed simulation strategy is required. The simulation strategy in this work allows the development and testing the MPC by its ability to simulate different disturbance scenarios.

2.4 Simulation Strategies for the RPSA System

Solving the RPSA plant model is an enormous challenge because typical PSA models are very close to numerical instability due to steep fronts in spatially-dependent variables and discontinuities due to changing boundary conditions. In many other studies, models use simplifying assumptions to reduce the computational cost such as gas-solid thermal equilibrium, negligible pressure drop, ideal plug flow and others. A summary of some recent PSA modeling studies in summarized in Table 1.1. These physical effects are critical to accurately modeling the dynamics in RPSA systems, and cannot be sacrificed in this study. The RPSA plant model present here makes the fewest assumptions and includes the most physical phenomena in the current literature.

Modeling the RPSA system requires a special simulation technigue. Comsol Multiphysics® is a commercially available software package which specializes in solving nonlinear PDE systems, but does not have a convenient method of solving cyclic systems. The model in Eqns. 2.1-2.7 must be solved sequentially for each step, and

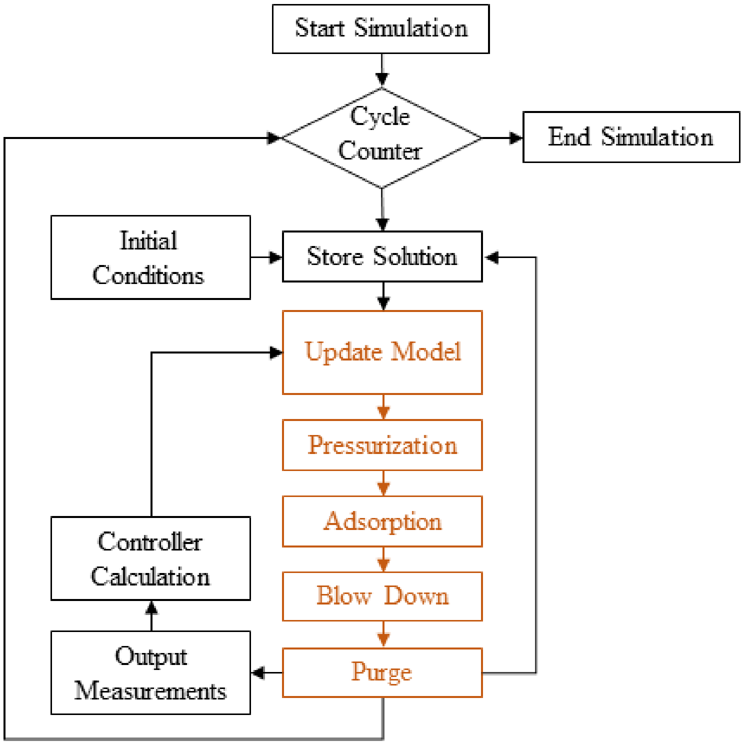


Figure 2.2: Simulation Strategy for the RPSA Model: The four RPSA cycle steps are solved sequentially in Comsol Multiphysics with the solution of the previous step used as the initial condition of the next. MATLAB is used to facilitate solving sequential cycles by saving previous solutions of the purge step and applying them as initial conditions of the next pressurization. Control calculations are also performed in MATLAB. Comsol Multiphysics tasks are highlighted in orange.

the solution of each step must be used as the initial condition for the next. At the beginning of each RPSA pressurization step, the solution to the purge step must be used as the initial condition. Comsol® is not able to accomplish the latter task directly, so MATLAB® is used to facilitate simulating sequential cycles. A summary of this simulation strategy is found in Fig. 2.2. MATLAB® is also used to make all feedback controller calculations and completes the simulated closed-loop system. Comsol Multiphysics® v5.2 and MATLAB® R2016B was used to simulate the plant model using a Dell® Precision Tower running dual core Xeon CPU (3.2 GHz) with 64GB RAM. On average, a RPSA cycle converges in approximately 27 sec.

2.5 Open Loop Dynamic Response

RPSA systems have two types of dynamics. "Intra-cycle" dynamics are the dynamics inside a single RPSA cycle such as changing pressures, temperatures and composition fronts through the adsorber column. An example of intra-cycle dynamics can be seen in Fig. 2.3. A single RPSA cycle profile of the adsorber column pressure is shown. The intra-cycle dynamics are essential to the operation of the RPSA cycle, and to producing the desired O_2 product. "Inter-cycle" dynamics occur when the RPSA cycle is disturbed in some way. Both intra- and inter-cycle dynamics are nonlinear, highly coupled functions of both the cycle step durations and process disturbances. Because of these dynamics and the cyclic nature of RPSA systems, a traditional steady state definition is not possible because the system states never reach a time-invariant state. Fortunately, the RPSA cycles can reach a "cyclic steady state" (CSS) where the cycle profiles of all system states reach a constant, time-invariant form. Fig. 2.3 also shows an example of inter-cycle dynamics in response to a process disturbance.

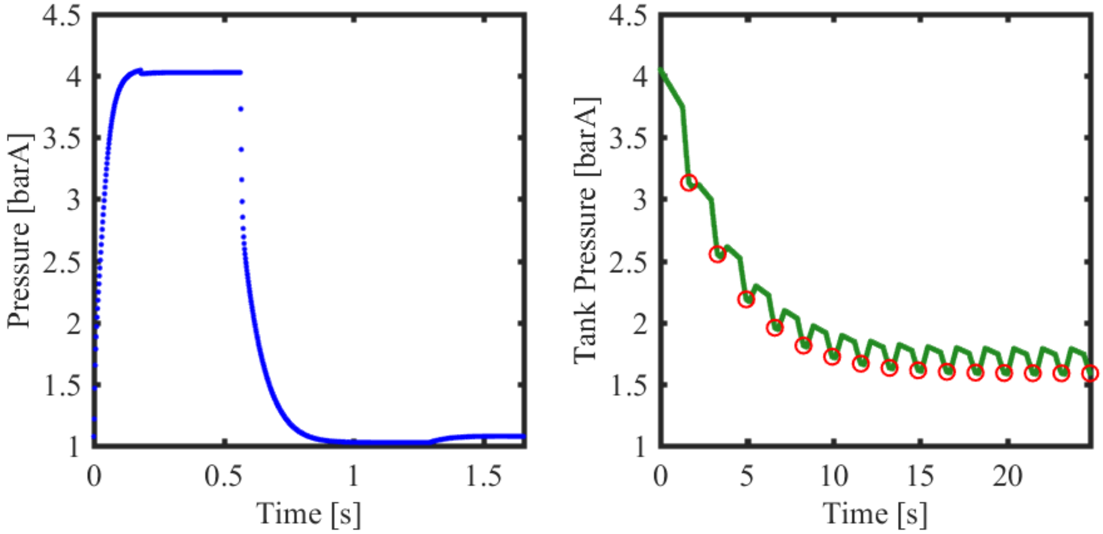


Figure 2.3: Open Loop Dynamics in the RPSA System: (Left) The intra-cycle dynamics of the adsorber column inlet pressure (blue) during a single RPSA cycle. (Right) The inter-cycle dynamics of the storage tank (green) in response to a process disturbance. The cycle reference point is highlighted in red circles.

The RPSA cycle profile returns to a constant form at CSS. This CSS property is useful for identifying the control problem for the RPSA system.

Feedback control is required to reduce or eliminate inter-cycle dynamics when process disturbances occur. Because of the rapid cyclic operation, controlling the entire cycle profile of the controlled variables is unnecessary and practically infeasible using MPC. An alternative definition of the CSS property helps mitigate this issue. Instead of considering the entire cycle profile of a system state, CSS can be defined as the point where all states have the same value at the beginning and end of the RPSA cycle. A cycle referencing strategy can then be used which captures all the necessary dynamic information without over-whelming the MPC with too much information. Choosing a cycle reference point reduces the measurement requirements in the system and makes controlling the RPSA cycle feasible, but choosing the correct reference

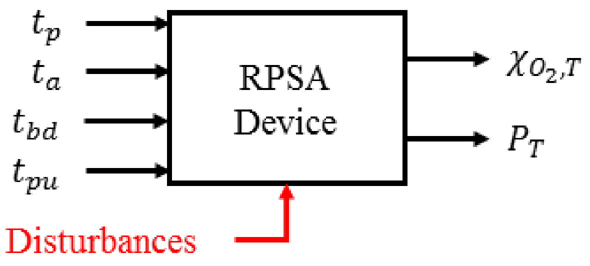


Figure 2.4: Open Loop Block Diagram for the RPSA System

point is essential for capturing all relevant information. The end of the purge step is an ideal choice for the cycle reference point for two reasons. The product composition in the tank only changes in the pressurization and adsorption steps, so waiting to measure composition until after purge reduces any sensor measurement delay effects which are common in composition sensors. Second, the tank pressure during the purge step must be high enough to ensure flow, and the pressure decreases during the purge step. If the controller maintains a high enough pressure after the purge step, the pressure also must be high enough during the entire purge.

The multivariable control problem can then be defined as shown in Fig. 2.4. The storage tank pressure (P_T) and O_2 composition ($\chi_{O2,T}$) at the end of the purge step are controlled by manipulating the four RPSA cycle step durations, (t_p,t_a,t_{bd},t_{pu}). The feedback controller will aim to reduce or eliminate the undesired inter-cycle dynamics caused by changes in several disturbance variables. These disturbance variables are kept general at this point, but will be specifically defined later. The feedback controller must be able to account for and utilize the coupled relationship between all four step durations and the controlled variables. Model Predictive Control (MPC) was chosen for this system because a process model and input constraints can be directly imposed in the controller and the MPC can independently manipulate the four cycle step durations.

2.6 Closing Remarks

The RPSA system uses a 4-step cycle to produce high purity O_2 using a novel, single bed design. This novel design presents a multivariable control problem where both the product composition and storage tank pressure. The cycle step durations are the only manipulated variables available to a feedback controller in a commercial MOC device, but they have a coupled, nonlinear relationship with the control variables. A nonlinear plant model for the RPSA system is required to design a feedback controller, test the overall control strategy and evaluate the closed-loop system performance. The plant model presented here makes the fewest simplifying assumptions, and is the most detailed in the current literature. This level of detail gives the best possible representation of reality to simulate the RPSA system and test the MPC. A combination of Comsol Multiphysics® and MATLAB® is used to solve this plant model and to make control calculations. A MPC can now be designed using this simulated RPSA system.

Chapter 3

Model Predictive Control for RPSA MOCs

3.1 Motivation

In this chapter, a Model Predictive Controller (MPC) for the RPSA system is presented which controls the product composition and storage tank pressure by independently manipulating all four cycle step durations. A multivariable control strategy for a cyclic system presents significant challenges, and several solutions are presented which make the MPC implementable in future experimental studies. Implementation onto embedded hardware is the eventual goal in designing the MPC, so each design decision is made with this overall goal in mind. A sub-space identification technique using Pseudo-Random Binary Sequence (PRBS) input signals is used to generate a linear model which reduces the computational cost of the nonlinear plant model. The closed-loop system is evaluated using several disturbance and set point scenarios. A comparison to traditional PID control is also presented. Most importantly, the work in this chapter was used to develop an MPC *design strategy* which can also be used in future implementation on a lab-scale RPSA prototype.

3.2 Control Challenges and Feedback Strategy

The RPSA system has a rapid, cyclic operation with typical total cycle times $\sim 5-6$ seconds. Because of the cyclic operation, the RPSA system never reaches a traditional steady state, so the feedback controller must consider a desired cyclic steady state as its objective. As seen in other studies [14], controlling the entire cycle profile is unnecessary and practically infeasible. A cycle reference point can be used to capture the relevant dynamics in the RPSA system, and gives the feedback controller enough information to make control decisions without too much added computational cost. Correctly choosing this cycle reference point in a way that does not sacrifice necessary dynamic information is very important. This can be accomplished by considering the storage tank equations in Eqns. 2.7 and 2.8. The end of the purge step is an ideal choice for the RPSA system, and is used in the MPC presented here. The choice of cycle reference point is not unique, and other acceptable choices exist.

The RPSA MOC design has two control objectives derived from both operational and performance requirements. Maintaining a product composition of $\sim 90\%$ is an obvious performance requirement, but controlling the storage tank pressure is also an important operational objective. In the single-bed design, oxygen from the storage tank is continuously delivered to the patient and used to back-purge the column. Neither would be possible if the tank pressure is too low. The two control variables, $\chi_{O2,T}$ and P_T, often compete when only one RPSA cycle step duration is manipulated. Fig. 3.1 shows an example of this behavior. A set of 10 open-loop simulations were

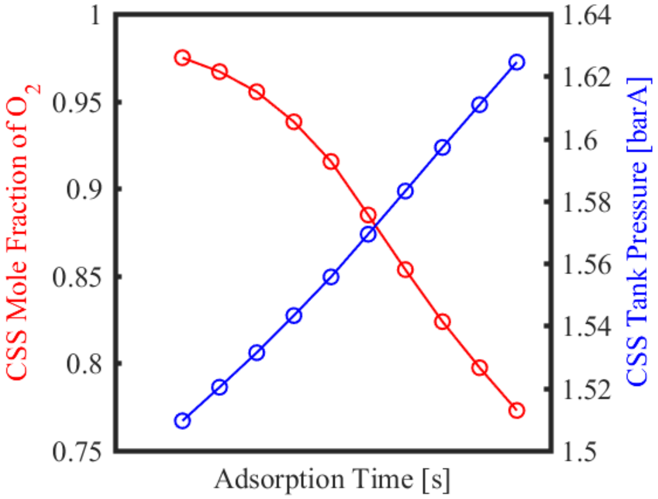


Figure 3.1: Effect of Adsorption Time on Storage Tank Composition and Pressure: The plot shows 10 open-loop simulations run with different adsorption times. The resulting CSS storage tank purities and pressures demonstrate how adsorption time has opposite effects on the controlled outputs.

run with only the adsorption time varied, and the controlled outputs, measured at the cycle reference point, experience opposite trends. When only a single RPSA cycle step duration is manipulated, a feedback controller would not be able to achieve both control variables. However, if the controller can independently manipulate all cycle step durations, the output competition effects can be significantly reduced or eliminated. A multivariable feedback controller is required to achieve both control objectives for the RPSA MOC system.

MPC was chosen for the RPSA system because it can use a process model to account for the multivariable interactions present in the RPSA system, but this benefit comes with an increased computational cost of solving an optimization problem. The computational cost of the MPC is directly related to the complexity of the process model, and as the model complexity increases, so does the computational cost. The

RPSA is highly complex nonlinear system which is modeled by a set of fully coupled PDEs. This process model would be impossible to implement into a MPC and solve in a short time, so a reduced order model is required for the RPSA system. Sub-space system identification is used to identify a linear model to be used in the MPC because the identification procedure is easy to implement in a real RPSA device in later work. The identified model relates the cycles step durations to the control variables and adequately predicts the RPSA dynamics in the narrow operation range for MOC applications. This linear model has a low computational cost and can be solved quickly and efficiently.

3.3 Sub-Space System Identification

Sub-space identification is a technique which uses specially-designed input signals and measured output responses to identify a process model. In this application, Pseudo-Random Binary Sequence (PRBS) input signals are chosen because they can continuously excite the RPSA system with relatively small amplitudes, and can be implemented easily into real systems [31]. A linear model, identified around a known baseline operating point, can accurately predict the RPSA dynamics in a narrow range in which the RPSA operates.

3.3.1 Identification of Baseline Operation

The RPSA MOC is required to produce $\sim 90\%$ O_2 product, but finding the RPSA cycle step durations to achieve this objective is difficult and time consuming. The four variables are manually manipulated without feedback control until the desired purity

Table 3.1: Summary of the Chosen Baseline Operating Point for the RPSA MOC

Parameter	Chosen Value	Measured Value
$\bar{t_p}$ [sec]	0.18	–
$\bar{t_a}$ [sec]	0.384	–
$\bar{t_{bd}}$ [sec]	0.73	–
$\bar{t_{pu}}$ [sec]	0.36	–
$\chi_{\bar{O2},T}$	–	0.90
$\bar{P_T}$ [barA]	–	1.5

is achieved. The tank pressure is also monitored during this process to ensure it is always super-atmospheric, so the purge and product flow rates can be maintained. The baseline value of tank pressure is the value when the composition is sufficient. The baseline operating point is summarized in Table 3.1. The chosen baseline input vector, $\bar{u} = [\bar{t_p}, \bar{t_a}, \bar{t_{bd}}, \bar{t_{pu}}]$ and measured output vector, $\bar{y} = \left[\chi_{\bar{O2},T}, \bar{P_T}\right]$ are used in designing the PRBS input signals. Manually determining the correct cycle step durations is difficult due to the nonlinear interactions the durations have with the product composition. An advantage to designing a MPC which manipulates these cycle step durations is they are changed online using the identified model and feedback output measurements.

3.3.2 PRBS Signal Design and Optimization

Once a baseline operating point is chosen, the RPSA plant model (Eqns. 2.1-2.8) must be perturbed around this operating point using specially designed PRBS input signals. Unfortunately, there is no well-defined procedure for designing PRBS signals, particularly for nonlinear systems, but several design criteria were developed for this application. The PRBS signals must designed and optimized experimentally in simulation to meet the following objectives:

- Each cycle step duration is perturbed using a PRBS-type signal with large enough perturbations to elicit a dynamic response, but not so large to move the RPSA far from its operating point

- The PRBS signals should be as short as possible to minimize the length of the simulation required to identify the model

The PRBS signal design can be reduced to a series of parameters to make optimization easier. A PRBS signal, $u(k)$, $k = [1, 2, ..., nT]$, is a series of $n \in \mathbb{Z}^+$ perturbations each with a duration of $T \in \mathbb{Z}^+$ RPSA cycles. The value of each perturbation, $u(\zeta T)$, $\zeta = [1, 2, ..., n]$, has the following constraints imposed:

- Upper and lower bounds, u_{min} are imposed according to,

  $$(1-\alpha)u_b \leq u(\zeta T) \leq (1+\alpha)\bar{u} \tag{3.1}$$

  where $0 \leq \alpha \leq 1$ is a PRBS design parameter.

- The magnitude of each perturbation, $|u(\zeta T) - u((\zeta-1)T)|$, must be large enough to elicit a dynamic response,

  $$|u(\zeta T) - u((\zeta-1)T)| \geq \beta\alpha\bar{u} \tag{3.2}$$

  where $0 \leq \beta \leq 1$ is a PRBS design parameter.

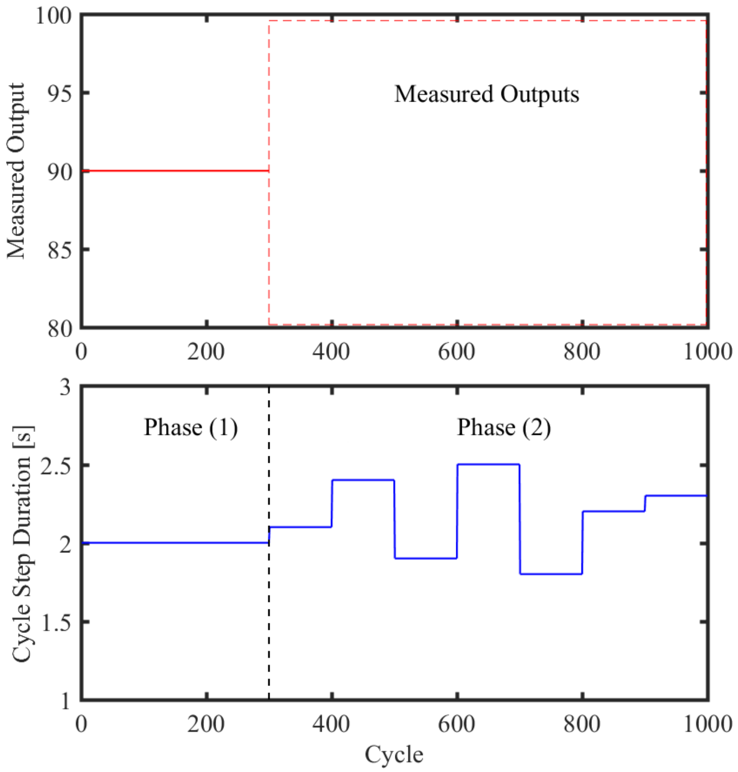


Figure 3.2: PRBS-type Simulations for Sub-Space Model Identification: A PRBS-type simulation has two phases. In phase (1), the RPSA reaches the known operating point, and all input signals are help constant. In phase (2), each cycle step duration input signal is PRBS-type which perturbs the RPSA system. The output variables are measured and collected during Phase (2). Sub-space model identification uses the input/output data from Phase (2).

- The magnitude of each perturbation must not be too large to keep the RPSA close to the operating point.

$$|u(\zeta T) - u((\zeta - 1)T)| \leq \gamma \alpha \bar{u} \tag{3.3}$$

where $0 \leq \beta < \gamma \leq 1$ is a PRBS design parameter.

Table 3.2: Summary of PRBS-type Simulations for System Identification

No.	n	T	α	β	γ	NRMS $\chi_{O2,T}$ [%]	NRMS P_T [%]
1	100	5	0.05	0.1	0.2	89.9	96.9
2	100	1	0.10	0.1	0.2	97.7	82.9
3	100	5	0.10	0.1	0.2	84.2	93.9
4	20	5	0.20	0.1	0.2	99.5	99.1
5	100	1	0.20	0.1	0.2	95.5	92.4
6	100	20	0.20	0.1	0.2	86.2	88.7

The five PRBS signal design parameters, n, T, α, β, γ, can be experimentally varied in simulation to determine the optimal signal design. Each PRBS-type simulation consists of two phases. First, the RPSA system is brought to CSS at the operating point $(\bar{u}, \bar{y})$ before perturbing the system. Fig. 3.2 summarizes a PRBS-type simulation used in the sub-space system identification procedure. Each cycle step duration signal is designed using the PRBS signal design parameters. These four step duration signals perturb the output variables around the baseline value which are measured during the simulation. The measured output signals and known input signals are then used to identify the linear model. The MATLAB® implementation of the n4sid algorithm [32] was used to identify the linear model of the form,

$$\begin{aligned} \hat{x}(k+1|k) &= A\hat{x}(k|k) + Bu(k|k) \\ \hat{y}(k|k) &= C\hat{x}(k|k) + Du(k|k) \end{aligned} \tag{3.4}$$

where A, B, C and D are the state space matrices resulting from the n4sid algorithm. The number of estimated system states in $\hat{x}$ are also determined by n4sid to best predict the measured output signals. The first 2/3 of the data collected in the PRBS-type simulations are used to in the n4sid algorithm to identify the model, and the last 1/3 of the PRBS data is used to validate the linear model predictions. Because the optimization of the PRBS signal design is experimentally-based, a metric is needed to gauge the identified model quality for different design parameters. In this work, the normalized root mean square (NRMS) is used to determine "goodness-of-fit". Many PRBS-type simulation were performed to optimize the PRBS signal design. Some of the PRBS design parameters can be found with a understanding of the RPSA dynamics. β and γ were chosen from knowledge and experience in running open-loop simulations, while n, T and α had to be determined experimentally. Simulation 4 in Table 3.2 was chosen as the optimal PRBS-type simulation for model identification due to its high NRMS percentages for both output variables.

3.3.3 Identification Results

The PRBS input/output data chosen for identifying the linear model is shown in Fig. 3.3. Region (a) shows the first 2/3 of the data used to identify the model and Region (b) is the portion of the data set used for model validation. The identified model has eight estimated states which were chosen by the n4sid algorithm to give the best prediction quality to the PRBS data set. Stability of the identified model was enforced, so $\lambda(A) \leq 1$. The model was further tested using various step tests by perturbing individual step durations. This identified model is detailed enough to predict the RPSA dynamics in the narrow operating range of $\sim 90\%$ O_2 product

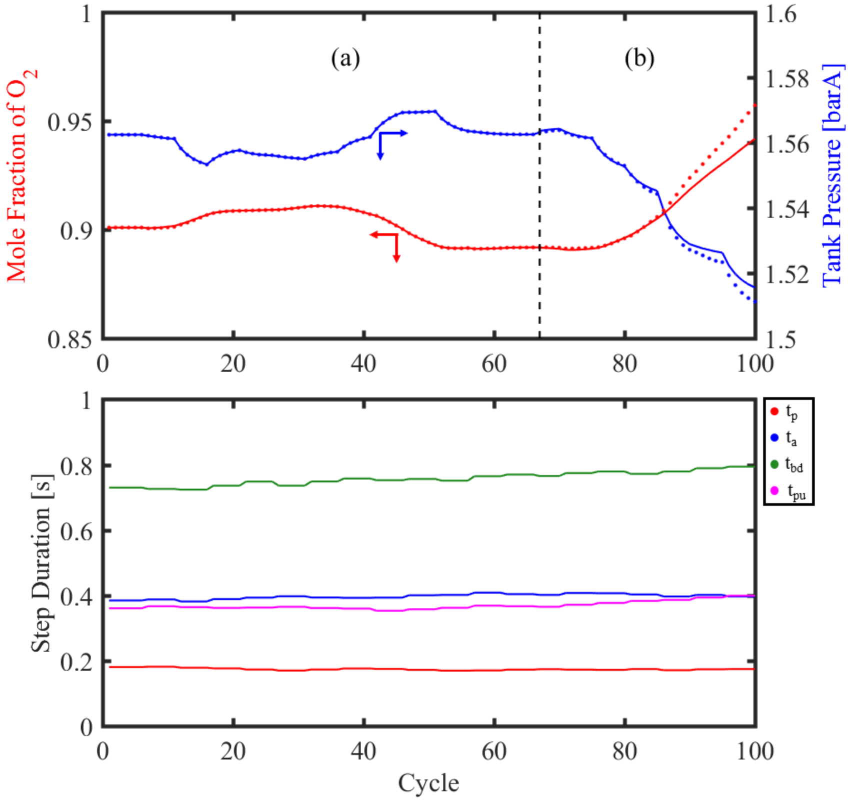


Figure 3.3: Sub-Space Identification using PRBS Signals: The first 2/3 of the PRBS data set, region (a), was used to identify a linear model, and the last 1/3, region (b), was used to validate the linear model. In the validation region, prediction horizons less than 30 cycles give very good approximations to the RPSA PDE model. Solid lines are the PRBS data, and the dots are the linear model predictions.

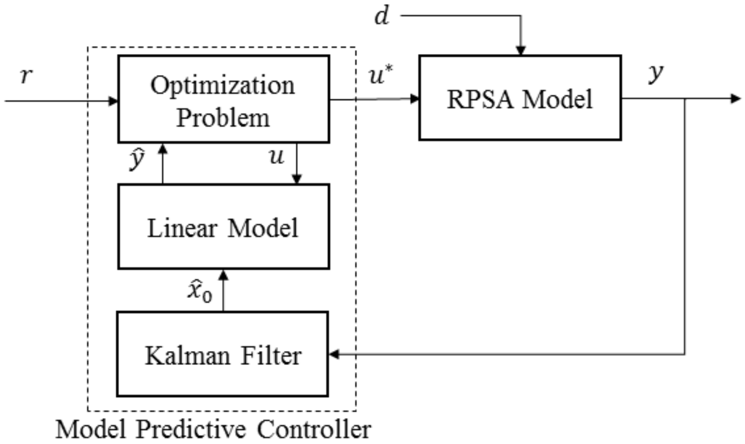


Figure 3.4: Block Diagram of the MPC Algorithm: The MPC uses a Kalman filter, identified linear model and an optimization problem to control the product composition and tank pressure by manipulating the cycle step durations.

composition without the prohibitive computational cost of the full PDE plant model. The identified model is used in the multivariable MPC optimization formulation.

3.4 Multivariable Model Predictive Controller

The multivariable MPC designed for the RPSA system manipulates the four cycle step durations to control product composition and storage tank pressure. The MPC uses a convex quadratic optimization program, the identified linear model described above and a discrete Kalman filter to provide the model initial condition. A block diagram of the closed-loop RPSA-MPC system is shown in Fig. 3.4. The discrete Kalman filter is shown in Eqn. 3.5.

$$\begin{aligned} \hat{x}(k+1|k) &= (A-LC)\hat{x}(k|k-1) + (B-LD)u(k|k) + Ly(k|k) \\ \hat{x}(k|k) &= \hat{x}(k|k-1) + M_I(y(k|k) - C\hat{x}(k|k-1) - Du(k|k)) \end{aligned} \tag{3.5}$$

where L is the Kalman observer gain and was found by solving a Ricatti Equation such that $\lambda(A-LC) \leq 1$. The innovation matrix, M_I, can also be found to improve the performance of the Kalman filter.

The MPC can be formulated similar to standard MPC formulations [26] according to,

$$\begin{aligned} \min_{\substack{u(k+i|k) \\ i=1,2,...,M}} & \sum_{i=1}^{N} [\hat{y}(k+i|k) - r(k+i|k)]^2_{\omega_1} + \sum_{i=2}^{M} [\Delta u(k+i|k)]^2_{\omega_2} \\ \text{s.t.} \;\; & \hat{x}(k+i+1|k) = A\hat{x}(k+i|k) + Bu(k+i|k) \\ & \hat{y}(k+i|k) = C\hat{x}(k+i|k) + Du(k+i|k) \\ & u_{min} \leq u(k+i|k) \leq u_{max}, \;\; i = 1,2,...,M \\ & \hat{y}_{min} \leq \hat{y}(k+i|k) \leq \hat{y}_{max}, \;\; i = 1,2,...,N \end{aligned} \tag{3.6}$$

where $\hat{y}$ is the estimated outputs, $r \in \mathbb{R}^{2\times 1}$ is the set point, $\Delta u \in \mathbb{R}^{4\times 1}$ is the change in $u(k+i|k)$, $\omega_1 \in \mathbb{R}^{2\times 1}$, ω_2, $\omega_3 \in \mathbb{R}^{4\times 1}$ are weighting vectors. $A \in \mathbb{R}^{8\times 8}$, $B \in \mathbb{R}^{8\times 4}$, $C \in \mathbb{R}^{2\times 8}$ and $D \in \mathbb{R}^{2\times 4}$ are the identified linear model state space matrices, and $\hat{x} \in \mathbb{R}^{8\times 1}$ is the estimated state vector. u_{min}, $u_{max} \in \mathbb{R}^{4\times 1}$ are lower and upper bounds on u and $\hat{y}_{min}$, $\hat{y}_{max} \in \mathbb{R}^{2\times 1}$ are the lower and upper bounds on $\hat{y}$. N is the prediction horizon and M is the control horizon.

The advantage of the MPC algorithm is apparent from the objective function.

Output deviations from set point are penalized explicitly, as are large manipulated input changes. The identified model is imposed as a constraint, and both inputs and predicted outputs are given bounds. Each term in the objective function can be tuned, and has an effect on the closed-loop performance. However, the coupled effects of the output variables cause output oscillations around a CSS. An example is shown in Case (a) of Fig. 3.5. Both output variables oscillate around their set points, and the MPC must oscillate the cycle step durations to correct this. Standard MPC fails to achieve the control objectives, and the addition of integral action must be added to improve controller performance.

Integral action is used in feedback control to eliminate steady state offset, and is added to enhance the standard MPC formulation. The updated MPC with integral action is shown as,

$$\begin{aligned}
\min_{\substack{u(k+i|k)\\ i=1,2,...,M}} \; & \sum_{i=1}^{N} \left[\hat{y}(k+i|k) - r(k+i|k)\right]^2_{\omega_1} + \sum_{i=2}^{M} \left[\Delta u(k+i|k)\right]^2_{\omega_2} \\
& + \sum_{i=1}^{M} \left[u(k+i|k) - \bar{u}(k+i|k)\right]^2_{\omega_3} \\
\text{s.t.} \; & \hat{x}(k+i+1|k) = A\hat{x}(k+i|k) + Bu(k+i|k) \\
& \hat{y}(k+i|k) = C\hat{x}(k+i|k) + Du(k+i|k) \\
& u_{min} \leq u(k+i|k) \leq u_{max}, \;\; i = 1, 2, ..., M \\
& \hat{y}_{min} \leq \hat{y}(k+i|k) \leq \hat{y}_{max}, \;\; i = 1, 2, ..., N
\end{aligned} \tag{3.7}$$

where $\omega_3 \in \mathbb{R}^{4\times 1}$ is a tuning parameter. This form of integral action penalizes deviations of the manipulated inputs from their baseline values to discourage extraneous

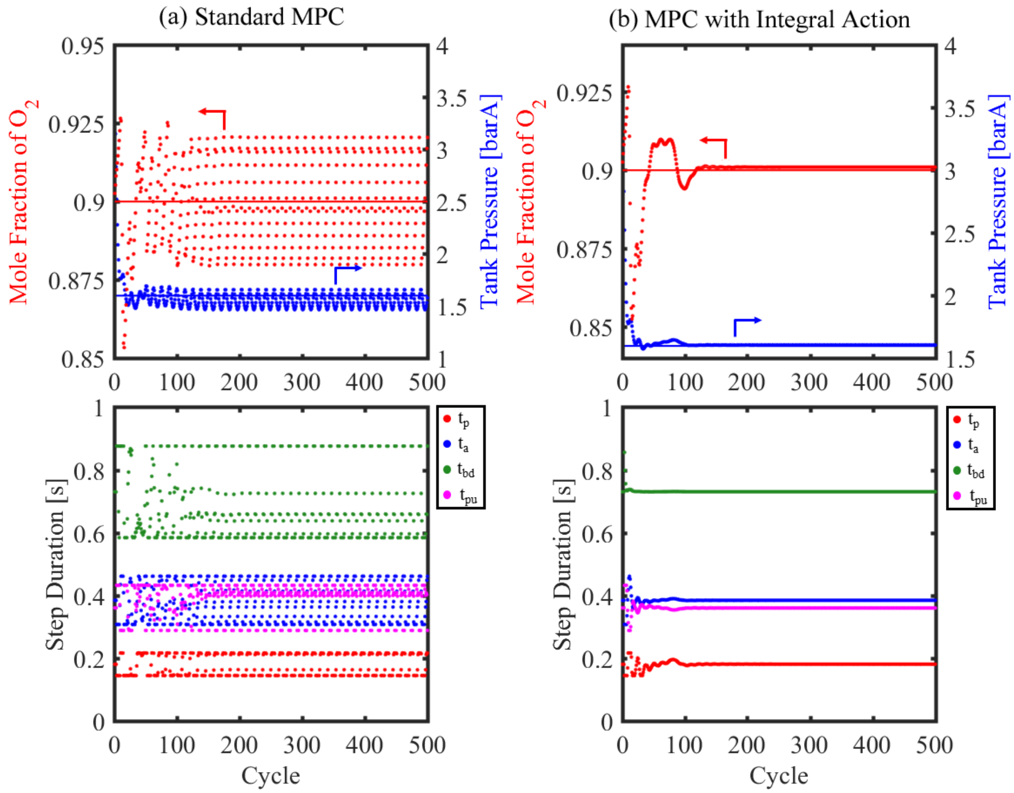


Figure 3.5: Effect of Integral Action on MPC Performance: (Left) The standard MPC causes large oscillations in both outputs and violates the control objectives at CSS. (Right) Using the same tuning parameters, the integral formulation reduces these oscillations to zero, and brings the RPSA system to CSS very quickly. When the outputs deviate from set point, the MPC adjusts the inputs accordingly, but the integral term encourages them back to baseline at CSS.

control actions. The integral action term can be tuned using ω_3, so set point tracking is still the most important objective. Case (b) in Fig. 3.5 shows integral action eliminating output oscillations and allows the cycle step durations to convert to optimal values. The MPC formulation with integral action achieved all control objectives using a control horizon of 23 cycles and a prediction horizon of 25 cycles. The MPC can then be evaluated using simulated MOC disturbances.

3.5 Controller Evaluation

The MPC developed here was intended to function in a MOC device which was simulated using the RPSA plant model. To evaluate the MPC performance, a set of realistic process disturbances were chosen to determine if the MPC could enhance the MOC device operation. To further test the MPC, set point tracking and comparison to traditional PID control are also shown.

3.5.1 Disturbance Rejection

Four case studies are given which demonstrate the ability of the MPC to reject realistic process disturbances which occur in MOC devices:

(1) A decrease in feed gas temperature

(2) A simultaneous pulse of feed and product flow rates

(3) A fluctuation in feed flow rate

(4) A decrease in adsorber bed capacity

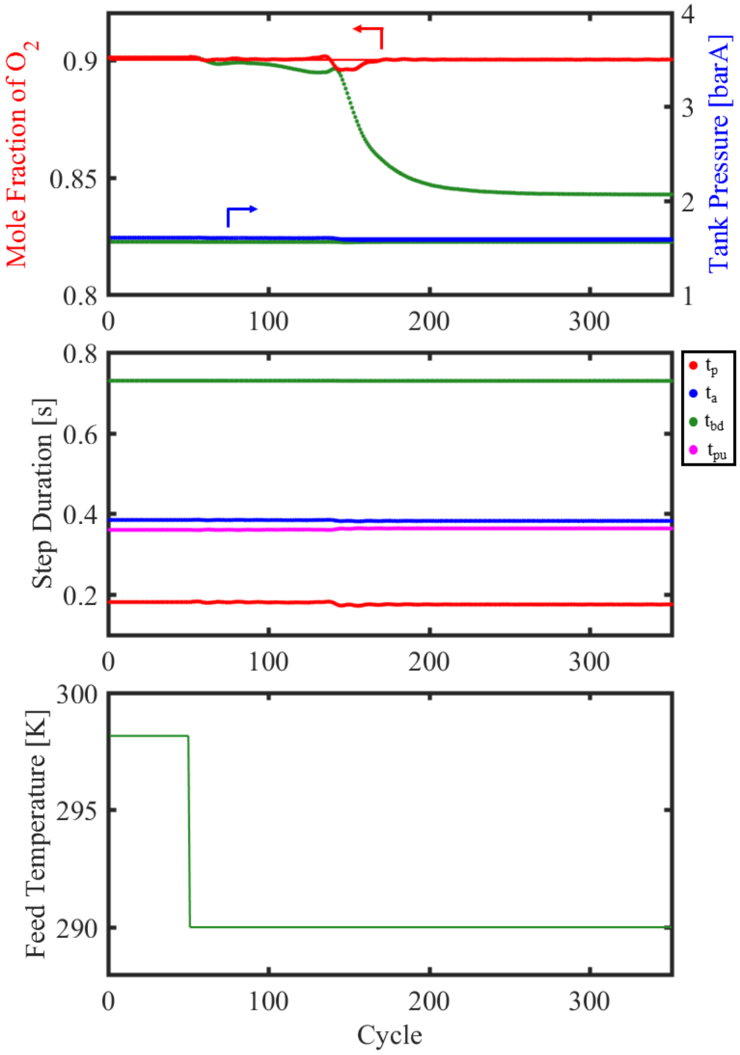


Figure 3.6: Changes in Feed Gas Temperature: A 9K step decrease in the feed temperature occurs at cycle 50. The MPC dramatically improves the closed-loop response, and purity remains at set point at CSS. The open loop output response (green) shows a large decrease in oxygen purity well below the required 90%.

Case (1), shown in Fig. 3.6, simulates a change in feed gas temperature. Some MOCs are considered portable, and patients typically travel while using these small devices. In winter months, the feed gas temperature to the MOC device can drop dramatically when a patients travels outdoors. When the feed gas temperature drops from 298 to 290K, the product composition drops from 90% to 84% in open-loop mode (green curve) when the MPC is turned off. The gas temperature affects the adsorption capacity of the LiLSX zeolite material. The open-loop response also demonstrates a significant delayed response between the product composition and feed temperature due to the heat transfer effects in the RPSA plant model. In closed-loop mode, the MPC makes small adjustments to four cycle step durations to keep the product composition at the required 90% (red curve). Note in this case, the MPC does not need to make large changes to the cycle step durations, and only makes small changes. The cycle step durations not only have a direct impact on overall RPSA performance, but they also have a coupled effect which is highly complex. Only by independently manipulating the cycle step durations can the MPC fully utilize these relationships.

In Disturbance Case (2), a simultaneous rectangular pulse in both product and feed flow rates is simulated. Stationary-type MOC devices are designed for use while the patient is at home. These types of MOCs are typically larger and also have a range of product delivery rates which the patient can change. A patient will typically use different product flow rates depending on their breathing habits (for example, breathing rate decreases while sleeping). When a patient changes the product flow rate, the MOC is programmed to adjust the feed flow rate via some kind of "look-up" table programmed into the device. Fig. 3.7 shows a change in the product flow rate 5 to 6 SLPM of O_2. In open-loop mode, the pulse causes a decrease in product

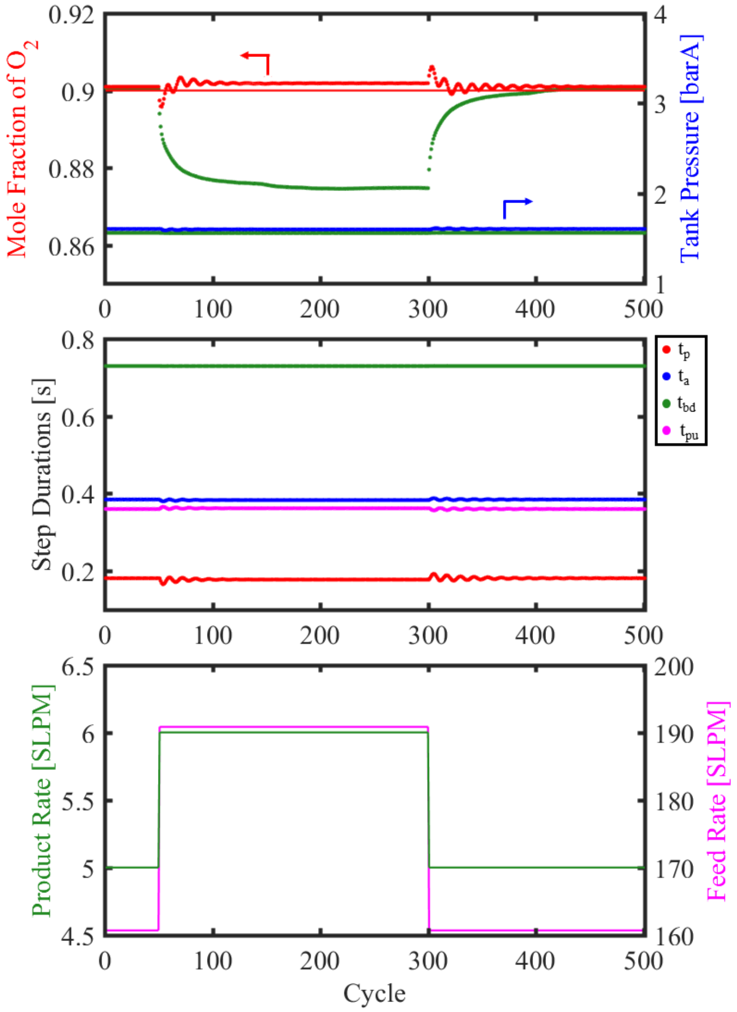


Figure 3.7: Simultaneous Feed and Product Flow Rate Changes: The product flow rate increases from 5LPM to 6LPM between cycles 250 and 300, and the feed flowrate is scaled accordingly. The MPC rejects the disturbance and maintains the oxygen purity at or above 90% at CSS. The open loop output response (green) drops to almost 87%, much lower than the required oxygen purity.

purity of approximately 3% below the required 90%. The MPC is able to make small adjustments to the cycle step durations which keep the product purity at or above the required level.

Disturbance Case (3) simulates fluctuations in the feed flow rate and is shown in Fig. 3.8. In MOC devices compressors are used to increase the pressure of the feed air. These compressors could experience dynamics from disturbances which could negatively impact the MPC performance. In this simulated disturbance, the feed flow rate is pulsed, and the product composition experiences some minor oscillations. The MPC responds by varying the cycle step durations, and the output oscillations finish before the disturbance ends. The MPC reduced the impact of this disturbance by shortening its effects, but it also sacrificed the tank pressure set point tracking to reject the disturbance to product composition. The pressure set point can be considered a soft constraint in the MPC optimization formulation. The MPC tuning parameters were chosen to encourage this choice, but constraints on the outputs prevent pressures too low to prevent proper RPSA operation.

In Case (4), contamination of the adsorber bed is simulated by decreasing the saturation capacity of the zeolite. In a typical MOC device, water contamination can occur over long periods of time, and will eventually prevent the device from producing the required O_2 purity. Without feedback, the device cannot respond to this type of disturbance, and the adsorber bed would have to be replaced by the MOC device manufacturer. With the MPC, the step durations can be manipulated as the performance deteriorates, and increase the bed lifespan which reduces maintenance costs for the patient.

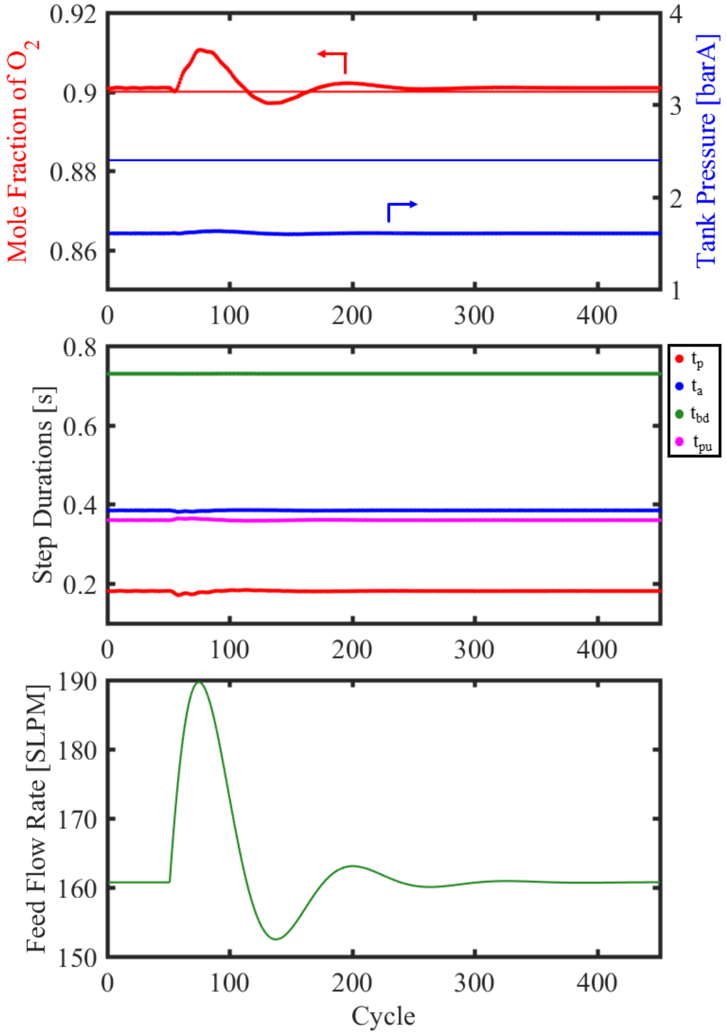


Figure 3.8: Fluctuations in Product Flow Rate: A fluctuation in feed flow rate simulates a disturbance due to compressor dynamics, and causes an oscillation in the product composition. The MPC makes small adjustments to the step durations to mitigate this disturbance.

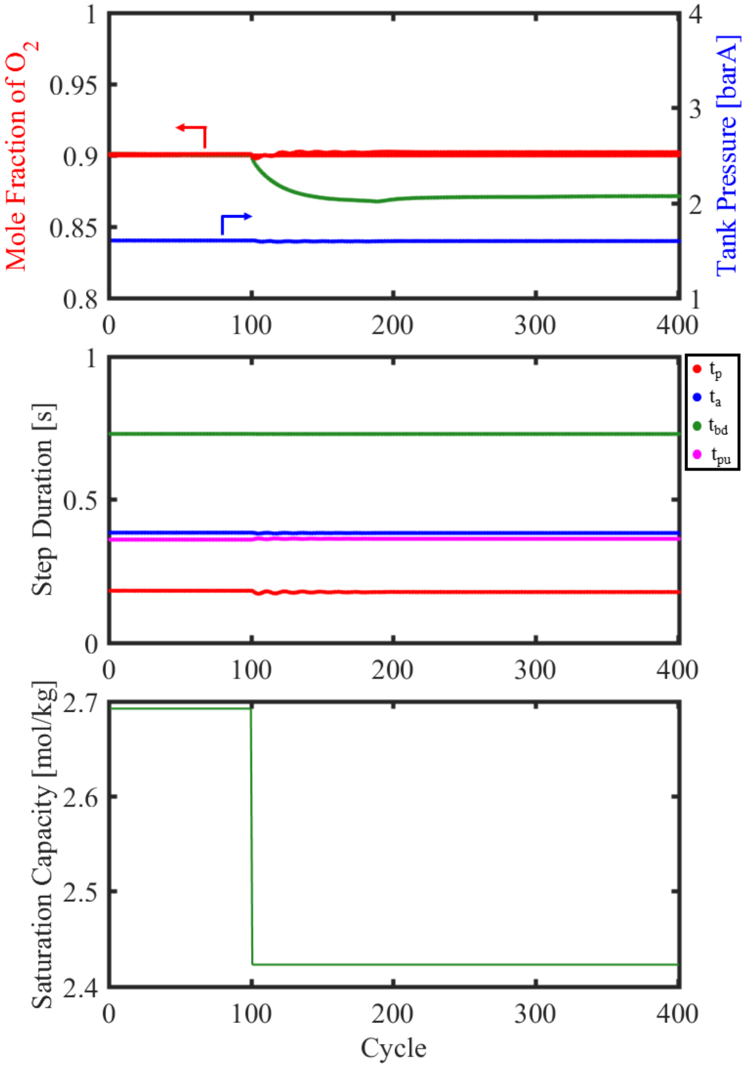


Figure 3.9: Decrease in Adsorber Bed Capacity: As a MOC device operates for long periods of time, the adsorber bed can be contaminated by moisture, which deactivates the zeolite adsorber capacity. (Top) In open-loop (green curve), the O_2 purity decreases below the required 90% when the saturation capacity of the bed decreases. In closed-loop, the MPC adjusts the step durations and maintains the required purity.

3.5.2 Comparison to PID Control

A comparison between the MPC and PID control is necessary to demonstrate the superior performance of the multivariable controller over the simpler PID controller. However, this comparison is not easy because it is difficult to design a PID controller which can control both output variables by manipulating all the cycle step durations. To make the comparison easier, a single-input, single-output (SISO) PID controller was used which controls only the product composition by manipulating only the adsorption time. The remaining cycle step durations are a constant multiple of the adsorption time. A similar PID strategy was used by others in the literature [5, 14]. A comparison between the MPC and the SISO PID controller is shown in Fig. 3.10. The same feed temperature change simulated in Disturbance Case (1) is used in this comparison. The MPC independently manipulates the four step durations, while the PID manipulates only adsorption time. The total cycle time is shown in Fig. 3.10 to compare the repsonses. Although both MPC and PID can control the product composition, the PID controller imposes significant output oscillations which is not present in the MPC response. Furthermore, the PID controller makes very large changes to the total cycle time while the MPC only makes minor adjustments. This comparison further reinforces the high degree of interactions between all cycle step durations which only a mulitvariable controller can utilize.

The SISO PID controller has severe limitations that preclude its use in the RPSA system. Unlike MPC, there is no direct imposition of input constraints, and saturation blocks must be in the PID controller to correct this. The main disadvantage to PID is its inability to utilize the coupled relationship of the cycle step durations. These interactions are vital to controlling the RPSA device and have a significant impact as

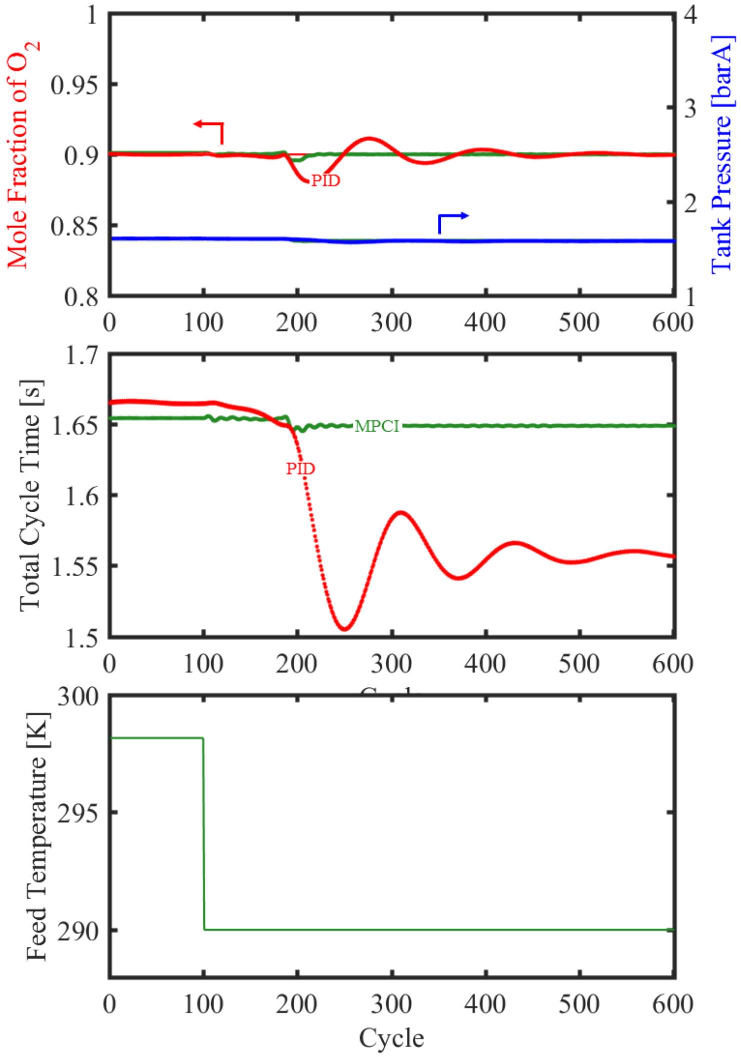


Figure 3.10: A Comparison between MPC and PID Control: A SISO PID controller (red) makes large changes to the total cycle time, and has a very long response time. The MPCI (green) changes the ratios between the step durations, and does not need to change the total cycle time dramatically to control the RPSA system.

seen in the various disturbance case studies. For these reasons, multivariable MPC is a superior choice over PID control for the RPSA system.

3.5.3 Set Point Tracking

Set point tracking is a difficult problem in complex nonlinear systems because the controller must move the system to a completely different operating point. This also challenges the linear model approximation around the original operating point made during model identification. Fig. 3.11 shows a set point change from 90% to 95%. The MPC is able to track the set point change accurately because the RPSA system remains close to original operating point. As the system moves further from the 90% point, the linear model quality will deteriorate quickly. MOC devices only operate at a single point, so the linear model approach works very well. If the RPSA system was used in a different concentrated oxygen application with different product purity requirements, this MPC would not perform well, and would have to be modified.

3.6 Stability of the MPC Algorithm

Proving stability of the MPC is challenging, but some tools exist which can aid demonstrating stability. The MPC presented here has a discrete time, constrained convex quadratic optimization objective function, and a linear model approximation of a nonlinear system. A stability proof would consist of two objectives: demonstrate a stability criterion for the objective function without any kind of terminal constraint and show the error between the identified and process models is finite. The combination of these two expressions could yield a sufficient stability condition. This is a

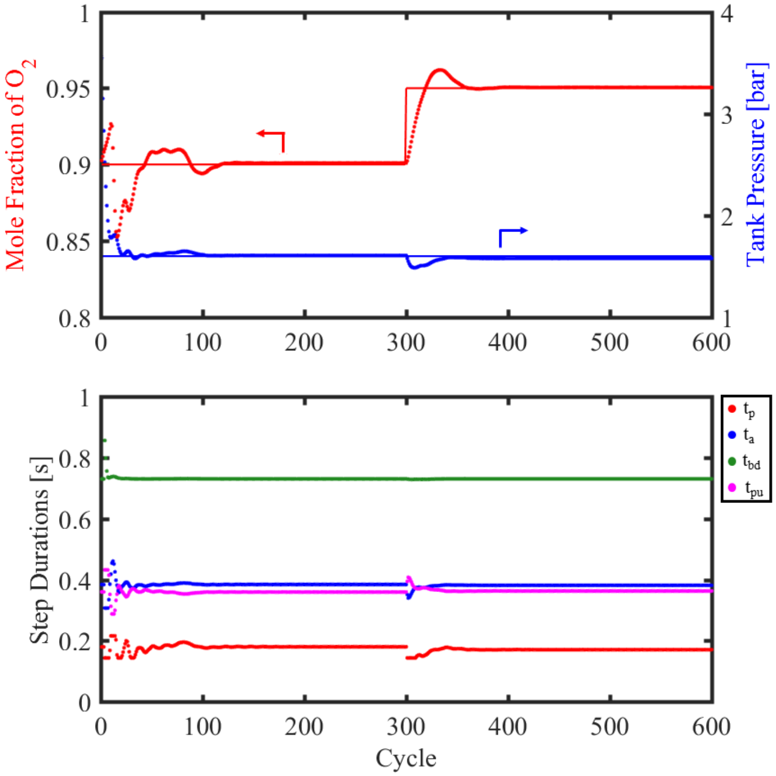


Figure 3.11: Set Point Tracking using MPC: The mole fraction set point increases to 95% at cycle 300, and the pressure set point remains constant. The MPC adjusts the cycles step durations to achieve this new set point with small offset.

complex problem, and the stability problem is presented here, but future work will be required to fully complete the proof.

A study by Primbs, et al [33] provides a strategy for proving stability of a MPC objective function without terminal constraints. In this strategy, if the objective function, $J(\hat{x})$, can be shown to be decreasing over the control horizon, stability can be inferred.

$$J(\hat{x}(k+i+1|k)) - J(\hat{x}(k+i|k)) \leq 0$$

Relating these expressions to the full objective function expression allows them to show upper and lower bounds for the objective. This same strategy can be used with the MPC presented here, with one difference: the RPSA process is nonlinear, and there exists some error between the identified linear model and the process model. The second component of a stability proof must include an expression of this modeling error.

Proving the stability of the RPSA plant model is significantly more difficult part of the stability problem because the model is highly nonlinear, fully coupled and cannot be solved analytically. To set up the problem, the process model in Eqns. 2.1-2.8 can be expressed as,

$$y = \mathcal{H}(x, u, \alpha)$$

where x are the true system states and α are the set of model constants. This model includes all nonlinear effects and is taken as a true representation of a real RPSA system. The model error between the process model and the identified model, $\hat{\epsilon}$ can then be expressed as,

$$\hat{\epsilon} = y - \hat{y}$$

The identified linear model is designed to be stable, so to prove $\hat{\epsilon}$ finite, $\mathcal{H}$ must be stable, or at least bounded. Because of flow reversals and discrete valve switching, several states in the process model are not smooth, continuous functions, so proving stability of the model in the continuous time domain may be impossible, but it might be possible to generate some type of stability criterion in the discrete time domain. If $\hat{\epsilon}$ can be proven finite, it could be integrated with the stability of the MPC objective function, and a general stability statement for the closed-loop system may be possible. However, future work is required to fully complete this stability analysis.

3.6.1 Demonstrating CSS Behavior in the RPSA Model

An interesting example of the CSS condition came out of preliminary work in the MPC stability analysis which may aid future attempts at a proof. In simulation, a uniqueness behavior was observed between the chosen u of cycle step durations and the corresponding CSS value of y. Regardless of the initial condition, a given u will achieve the same y at CSS. The initial condition will change the path to the CSS condition, but not the final values. This can be demonstrated using the storage tank equations in Eqns. 2.7-2.8 and the boundary conditions in Table 2.1. At CSS, the value of the outputs at the beginning and end of each cycle are constant. This condition can be expressed mathematically as,

$$y(t = kt_{cycle}) = y(t = (k-1)t_{cycle})$$

where $t_{cycle} = t_p + t_a + t_{bd} + t_{pu}$ and k is the cycle index. Because the manipulated inputs appear in the integration bounds of each RPSA cycle step, the solutions of the outputs at the cycle reference point are functions of only the inputs, model constants and initial condition of that cycle (contributions of the state variables are solved for in the integrand). At cycle k, the outputs can be expressed as,

$$y(t = kt_{cycle}) = f\left(u, \alpha, y(t = (k-1)t_{cycle})\right)$$

This relationship is true of any RPSA cycle, but at CSS, the given expression simplifies to,

$$\lim_{k \to \infty} y(t = kt_{cycle}) = f(u, \alpha)$$

which demonstrates the CSS value of the outputs are not a function of the initial condition. This kind of proof may aid future attempts at proving the stability of the plant model, and is at least an interesting result of using bounds of integration as manipulated variables.

3.7 Closing Remarks

The multivariable MPC presented here controls the RPSA device by manipulating the cycle step durations, and can reject MOC-relevant process disturbances well. The sub-space identification technique using PRBS input signals is a very efficient method for generating a reduce-cost model used in the MPC calculations, and the technique is easy to implement experimentally. Although the MPC can track set points in a narrow operating regions, the linear approximation around this operating point limits the controllers effectiveness if the RPSA operates in a wider range of product compositions. A logical extension using piece-wise linear modeling can improve the MPC performance while still minimizing the computational cost. Stability considerations were discussed, but a rigorous proof will require more development. In this work, stability of the MPC will be experimentally demonstrated on a lab-scale RPSA prototype.

Chapter 4

Piece-wise Linear MPC for other RPSA Applications

4.1 Motivation

In this chapter, an extension to the MPC using a piece-wise linear modeling approach is presented which enables the MPC to control the RPSA device at many operating points. Although MOCs typically produce a single product composition of $\sim 90\%$, there are other concentrated O_2 applications which have other product composition requirements. Other applications also operate in a more diverse set of conditions, various product flow rates, and different scales. Piece-wise linear MPC keeps the advantage of low computational cost, while improving on the linear approximation. The generation of a piece-wise linear model for the RPSA system is shown, as well as additions to the MPC algorithm to enable model switching.

4.2 Multi-Model Predictive Control Strategy for Nonlinear Systems

Feedback control of PSA systems is not well studied in the literature, because most studies focus only on modeling and optimizing the PSA cycle design. However, feedback control is an essential component of a RPSA device to both operate the PSA cycle and ensure the device produces the desired product composition. Sun, et al used a traditional PID controller which controlled the product composition by manipulating the total cycle time [5]. Khajuria, et al used a Model Predictive Controller (MPC) which also controlled the product composition by manipulating the total cycle time [14, 15]. By only manipulating the total cycle time in control actions, these methods do not fully utilize the coupled effect the individual cycle step times have on PSA systems. A multivariable MPC using a single linear model was developed for a novel RPSA Medical Oxygen Concentrator (MOC) prototype which produces 90% O_2 for COPD therapies [25]. This multivariable controller independently manipulates the RPSA cycle step durations to control the product O_2 purity and storage tank pressure in the RPSA device. MPC is an ideal choice for RPSA systems because constraints can be imposed in all control decisions, and a process model can be used to predict the coupled effect of the cycle step durations on the tank composition and pressure. We have also demonstrated that the multivariable MPC can improve on traditional PID control because the MPC can independently manipulate all cycle step durations [25]. The MPC was able to reject realistic process disturbances to the RPSA device and track a single set point of 90% O_2, but the single linear model failed when the RPSA device operated outside this narrow operating range. For the RPSA device to produce different product compositions for other concentrated O_2 applications, the MPC using a single linear process model is not adequate. A nonlinear MPC formulation is a better option for the RPSA device.

Nonlinear MPC is an active, ongoing area of research which has great potential for nonlinear processes such as the RPSA device, but the computational cost of nonlinear MPC is a significant disadvantage. To make useful, online control decisions, the MPC used in the RPSA device must be solved quickly, in less time than the RPSA cycle (typically less than 6 sec). Ideally, a MPC will use a process model which accurately predicts the nonlinear RPSA dynamics while still minimizing the computational cost. This is a difficult problem to solve, because the nonlinear adsorption effects, discrete cycle step switching, flow reversals, and heat effects make modelling PSA systems incredibly challenging. There is then a trade-off between model complexity and computational cost which influences which kind of MPC can be used in the RPSA device. An attractive compromise between nonlinear and linear MPC is Multi-Model Predictive Control (M-MPC). M-MPC is an extension of standard MPC which uses a collection of reduced-order, often linear, models to better predict nonlinear system behavior [34]. Various approaches to M-MPC can be found in many industries such as wind energy [35], aeronautics [36, 37], steam generation in nuclear power plants [38] and medicine [39]. One approach from the medical industry changes the model parameters based on a patients medical information. The model chosen here has a constant structure, but parameters are unique to different types of patients [39]. Other approaches use a collection of models to predict nonlinear behavior, and algorithms choose which model is appropriate at a given time. In these algorithms a preliminary optimization will compare all the model predictions to check which is the most accurate. This chosen model is then used in the M-MPC control decision. Examples of this approach can be found in [40, 41, 42, 43]. This approach is useful in highly complex systems where choosing an appropriate model is non-trivial. The disadvantage to this method is the increased computational cost of solving an additional optimization problem. A second approach is to choose a model based on online output measurements. Example of this approach can be found in[38, 41, 44]. This strategy works well when output measurements can be used to define operating regions, and models can be identified or linearized around operating points. The RPSA device is one such system because the nonlinear dynamics are caused mainly by nonlinear adsorption effects.

Here, a new application of M-MPC for RPSA devices which uses a piece-wise linear MPC to operate and control the RPSA device. The detailed, nonlinear process model simulates the RPSA device, and is used to generate the piece-wise linear model using sub-space system identification techniques. The RPSA device will use the M-MPC to produce a range of 35-95% concentrated O_2. The M-MPC performance will be evaluated using the nonlinear process model to simulate realistic RPSA process disturbance rejection and set point tracking case studies.

4.3 Model Identification for Operating Space

The piece-wise linear model consists of a collection of linear models each identified at a different operating point in the nonlinear RPSA operating space. Each of the linear models are identified using the same sub-space identification procedure and PRBS input signals, but designing the PRBS signals correctly becomes more important and requires more careful consideration.

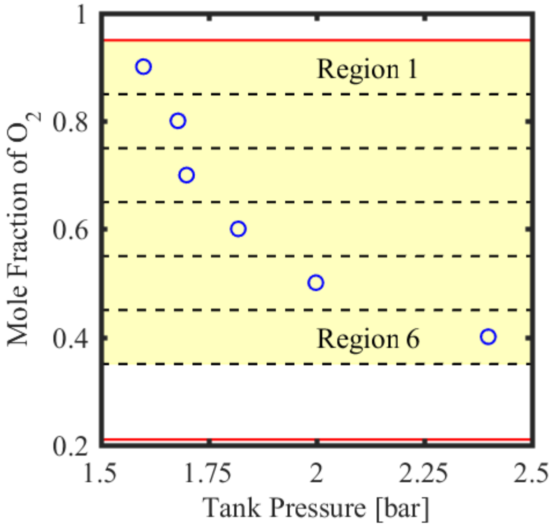


Figure 4.1: RPSA Operating Regions and Points: The nonlinear operating space (yellow) is split into 6 regions defined by operating points (blue cirles). The region boundaries are given as a ±5% O_2 range around the operating points.

4.3.1 Identification of Operating Points

The operating space is defined as the range of O_2 purities required in the RPSA device. The space is segmented into $j \in \{1, 2, .., 6\}$ operating regions where the RPSA system can be assumed locally linear, and each region is defined by an operating point. The size of each region is carefully chosen to be small enough for the linear approximation to be valid without being so small to have a cumbersome number of models. Each operating point is defined primarily by the desired value of $\chi_{O2,T}$. The corresponding value of P_T at the operating point is the measured CSS value found in simulation. Based on the set point tracking results in Chapter 3, each of the 6 regions covers a ±5% range in $\chi_{O2,T}$. A summary of the chosen operating regions and points is shown in Fig. 4.1. The yellow shaded region shows the operating space covered by the piece-wise linear model, and the blue circles highlighting each operating point. Region boundaries are shown as dashed lines. With the 6 regions, the piece-wise

Table 4.1: Linear Model Operating Points and Identification Results

Region	$\chi_{\bar{O_2},T}$	$\bar{P}_T$ [bar]	$\bar{t_p}$ [s]	$\bar{t_a}$ [s]	$\bar{t_{bd}}$ [s]	$\bar{t_{pu}}$ [s]
1	0.90	1.60	0.1800	0.3840	0.7300	0.3600
2	0.80	1.68	0.1986	0.3875	0.7296	0.3533
3	0.70	1.70	0.2308	0.3944	0.7300	0.3390
4	0.60	1.82	0.2086	0.4302	0.7288	0.3060
5	0.50	2.00	0.2307	0.4910	0.7321	0.2576
6	0.40	2.39	0.2999	0.7600	0.7075	0.1803

linear model can accurately model the RPSA system in the range of 35% to 95% O_2.

Finding these operating points is challenging because the four cycle step durations must be manually manipulated until the desired $\chi_{O2,T}$ is achieved at CSS. The found baseline inputs, $\bar{u_j} = [\bar{t_{p,j}}, \bar{t_{a,j}}, \bar{t_{bd,j}}, \bar{t_{pu,j}}]$ and the measured baseline outputs, $\bar{y_j} = \left[\chi_{O\bar{2},T,j}, \bar{P_{T,j}}\right]$ make up the operating point of region j. The main advantage of using the M-MPC is the controller will automatically find the desired $\chi_{O2,T}$ without manual manipulation of the step durations. Table 4.1 summarizes the found step duration values for each operating point. The nonlinearities in the RPSA system are due primarily from the adsorption effects in the adsorber column. The current measurement of $\chi_{O2,T}$ can be used to determine which model is most accurate as the M-MPC operates.

4.3.2 Sub-Space Identification using PRBS Signals

Each of the operating regions uses an identified linear model dedicated to that region, and each model will use a similar identification procedure to that used in Chapter

Table 4.2: Summary of PRBS Signals Used to Identify the Piece-Wise Linear Model for the RPSA System

Region	n	T	α	β	γ	NRMS $\chi_{O2,T}$ [%]	NRMS P_T [%]
1	20	5	0.20	0.1	0.2	99.5	99.1
2	20	5	0.20	0.1	0.2	98.9	98.8
3	20	5	0.20	0.1	0.2	99.7	99.4
4	100	5	0.20	0.1	0.2	92.8	96.7
5	100	5	0.20	0.1	0.2	93.8	97.3
6	100	5	0.20	0.1	0.2	96.8	95.9

3. However, in this application, more care needs to be taken during the model identification procedure to ensure the M-MPC will be able to switch between different region models. The five PRBS design parameters (n, T, α, β and γ) are chosen for each region so the linear model prediction is high. The PRBS signals for all regions are summarized in Table 4.2.

Each PRBS-type simulation was run, and the output data collected was used to both fit and validate the identified model generated by the n4sid algorithm. In the piece-wise model, each region model, in addition to being accurate inside the region boundaries, must also apply around the region boundaries, so the M-MPC can switch between models without sacrificing accuracy. In the PRBS signal design, the value of α is critical to ensure this behavior. A precise switching rule, described later, will make the choice of α more clear. In the PRBS-type simulations, the perturbations in the cycle step durations must cause the output to move inside and outside the region boundaries to ensure model accuracy in switching. An example of a PRBS response from Region 4 is shown in Fig. 4.2. Region 4 is resides in 55% to 65% $\chi_{O2,T}$, but the PRBS signals and α were chosen so $\chi_{O2,T}$ also moves outside this region. The Region

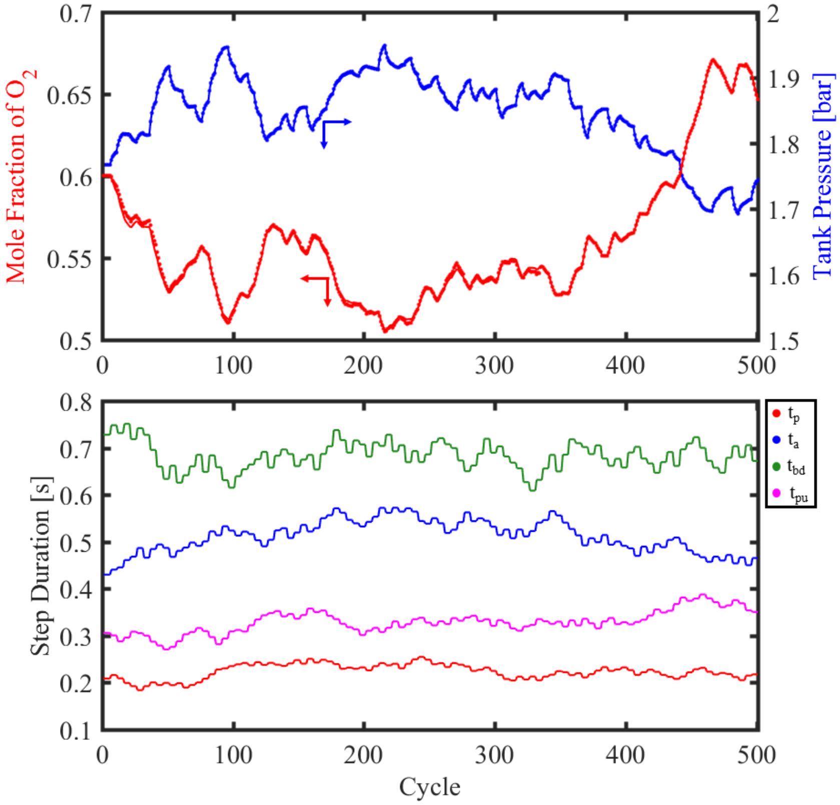


Figure 4.2: PRBS Simulation for Operating Region 4: In the bottom plot, each cycle step duration is a carefully designed PRBS-type signal. The PRBS signals perturb the RPSA PDE model, and the output response is shown in solid lines. Section (a) of the data is used to identify a linear model, and section (b) is used to validate the model. The linear model fit and prediction is shown as dotted lines.

4 model will then overlap with Region 3 and Region 5 models, and model switching is easy.

4.3.3 Piece-Wise Linear Model for the RPSA System

The piece-wise linear model then consists of the 6 identified linear models described here. Each linear model has the form,

$$\begin{aligned} \hat{x}(k+1|k) &= A_j\hat{x}(k|k) + B_j u(k|k) \\ \hat{y}(k|k) &= C_j\hat{x} + D_j u(k|k) \end{aligned} \tag{4.1}$$

where $(\cdot)_j$ corresponds a parameter in region j. Each model was identified to have the same number of estimated states, and all model have $Re[\lambda(A_j)] \leq 1$. Each region also has a Kalman filter of the form,

$$\begin{aligned} \hat{x}(k+1|k) &= (A_j - L_jC_j)\hat{x}(k|k-1) + (B_j - L_jD_j)u(k|k) + L_jy(k|k) \\ \hat{x}(k|k) &= \hat{x}(k|k-1) + M_{I,j}(y(k|k) - C_j\hat{x}(k|k-1) - D_ju(k|k)) \end{aligned} \tag{4.2}$$

where the observer gain L_j is obtained from the corresponding Ricatti equation and $Re[\lambda(A_j - L_jC_j)] \leq 1$.

Each region model is significantly different from the others because the nonlinear behavior of the RPSA changes in different regions. An example of these differences is shown in Fig. 4.3. The RPSA plant model from Eqns. 2.1-2.7 response (black curve) is compared three region models (from regions 4, 5 and 6). Region 5 model gives very accurate predictions of the plant model while the response resides in Region 5. The models in adjacent Regions 4 and 6 have significant error, up to 20%, which would severely damage the M-MPC decisions and lead to extremely poor closed-loop performance. Together, the piece-wise linear model provides reasonably accurate predictions of the RPSA plant model, and the M-MPC can use this model to control the RPSA device in a wide range of O_2 purities.

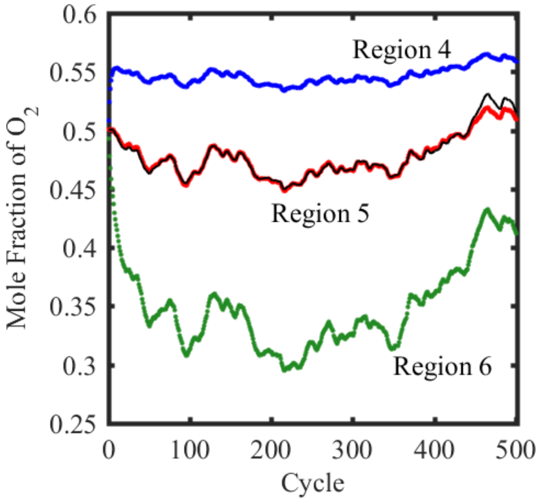


Figure 4.3: Comparison of the Piece-Wise Linear Model using in M-MPC: The RPSA PDE model (black) is approximated reasonably well by the linear model in Region 5 (red), but linear models from neighboring regions 4 (blue) and 6 (green) have up to 20% prediction error.

4.4 Multi-Model Predictive Controller

The M-MPC has a similar formulation to the MPC with integral action used in Chapter 3 except with region-specific and model and tuning parameters. Carefully chosen model switching logic and boundary considerations help the M-MPC to maintain stable output responses while tracking any $\chi_{O2,T}$ set point.

4.4.1 Controller Formulation

The M-MPC optimization problem shown below uses a quadratic objective function and linear constraints which are updated according to the region in which the RPSA resides.

$$
\begin{aligned}
\min_{\substack{u(k+i|k)\\ i=1,2,\ldots,M}} & \sum_{i=1}^{N}\left[\hat{y}(k+i|k)-r_j(k+i|k)\right]^2_{\omega_{1,j}}+\sum_{i=2}^{M}\left[\Delta u(k+i|k)\right]^2_{\omega_{2,j}} \\
& +\sum_{i=1}^{M}\left[u(k+i|k)-\bar{u_j}(k+i|k)\right]^2_{\omega_{3,j}} \\
\text{s.t. } & \hat{x}(k+i+1|k)=A_j\hat{x}(k+i|k)+B_ju(k+i|k) \qquad (4.3)\\
& \hat{y}(k+i|k)=C_j\hat{x}(k+i|k)+D_ju(k+i|k) \\
& u_{min,j}\leq u(k+i|k)\leq u_{max,j},\;\; i=1,2,...,M \\
& \hat{y}_{min}\leq \hat{y}(k+i|k)\leq \hat{y}_{max},\;\; i=1,2,...,N
\end{aligned}
$$

where $(\cdot)_j$ are region-specific parameters. The set point, $r_{j,1}(k)$ is set by the device user for the desired $\chi_{O_2,T}(k)$, and $r_{j,2}(k)$ is automatically adjusted based on $\bar{P}_{T,j}$. Updating the $r_{j,2}(k)$ keeps the local MPC closed to the operating point while the RPSA resides in region j. The tuning parameters, $\omega_{1,j}$, $\omega_{2,j}$, $\omega_{3,j}$ are region-specific to make local MPCs more or less aggressive depending on the nonlinear behavior of the RPSA plant model. The same $u_{min,j}$ and $u_{max,j}$ are used in the M-MPC as were used in the PRBS signal design procedure. Choosing these limits is essential for model switching in the M-MPC algorithm.

4.4.2 Switching Logic and Boundary Considerations

Switching between local MPCs does not require intensive computation because well-defined regions were found offline based on operating points. The online measurement $\chi_{O_2,T}(k)$ is used to determine the current region j because $\chi_{O_2,T}(k)$ is an excellent metric of the nonlinear adsorption effects in the RPSA device. After each cycle, the region j is updated based on the following switching logic,

$$j = \begin{cases} j+1, \ \chi_{O_2,T}(k) \geq \chi_j^+ + \delta \\ j, \ \chi_j^- - \delta < \chi_{O_2,T}(k) < \chi_j^+ + \delta \\ j-1, \ \chi_{O_2,T}(k) \leq \chi_j^- - \delta \end{cases} \tag{4.4}$$

where χ_j^- and χ_j^+ are the lower and upper bounds of region j as summarized in Fig. 4.1. A boundary tolerance, $\delta = 0.005$, is used to reduce extraneous switching when the device is close to a region boundary, and improves that stability of the piece-wise linear model. The choice in α from the PRBS signal design can be formalized using Eqn. 4.4. α is chosen so there exists a u_j^+, u_j^- such that,

$$\begin{aligned} u_{min,j} \leq u_j^+ \leq u_{max,j} \\ u_{min,j} \leq u_j^- \leq u_{max,j} \\ \chi_{O_2,T}\big|_{u=u_j^+}(k \to \infty) \geq \chi_j^+ + \delta \\ \chi_{O_2,T}\big|_{u=u_j^-}(k \to \infty) \leq \chi_j^- - \delta \end{aligned} \tag{4.5}$$

This means a local MPC can choose a u which causes the $\chi_{O_2,T}$ to move outside the current region at CSS. This ensures that each local controller can move the RPSA device between regions efficiently. There is a trade-off when choosing α. If α is too large, a linear model cannot accurately predict the nonlinear RPSA plant model, but if α is too small, switching cannot occur.

Proving the stability of the M-MPC algorithm is a difficult task. The RPSA plant model is a fully coupled nonlinear set of PDE equations that does not have an analytic solution, and a MPC with constraints does not have an analytic equation relating the current output measurements to the optimal solution. A stability proof for this algorithm needs to main components: proving stability of the discrete piecewise linear model and proving convergence of a finite horizon MPC. Both problems are not new, approaches to proving stability of discrete systems has been explored [45, 46, 47]. Finite horizon MPC has also been studied [33]. Applying these principles to the M-MPC algorithm can be part of future discussions.

4.5 M-MPC Performance and Evaluation

The M-MPC algorithm is evaluated using several disturbance rejection and set point tracking scenarios.

4.5.1 Set Point Tracking

Set points in the M-MPC algorithm occur in three possible locations: (a) operating points, (b) region boundaries, or (c) non-operating points. Fig. 4.4 shows examples of each case. In case (a), the set point occurs at the operating point for region 3. At CSS, $u(k = 100) = \bar{u_3}$, and the M-MPC converges with zero offset due to integral action. In cases (b) and (c), the set points are not a known operating point, and $u(k = 300)$ and $u(k = 500)$ were previously unknown cycle step durations the M-MPC found automatically. Without M-MPC the RPSA device could operate only at known operating points. Case (b) also demonstrates the effect of the boundary tolerance δ on closed-loop performance. $\delta = 0.005$ is large enough to minimize switching and prevents oscillatory dynamics at region boundaries. In all three cases, switching

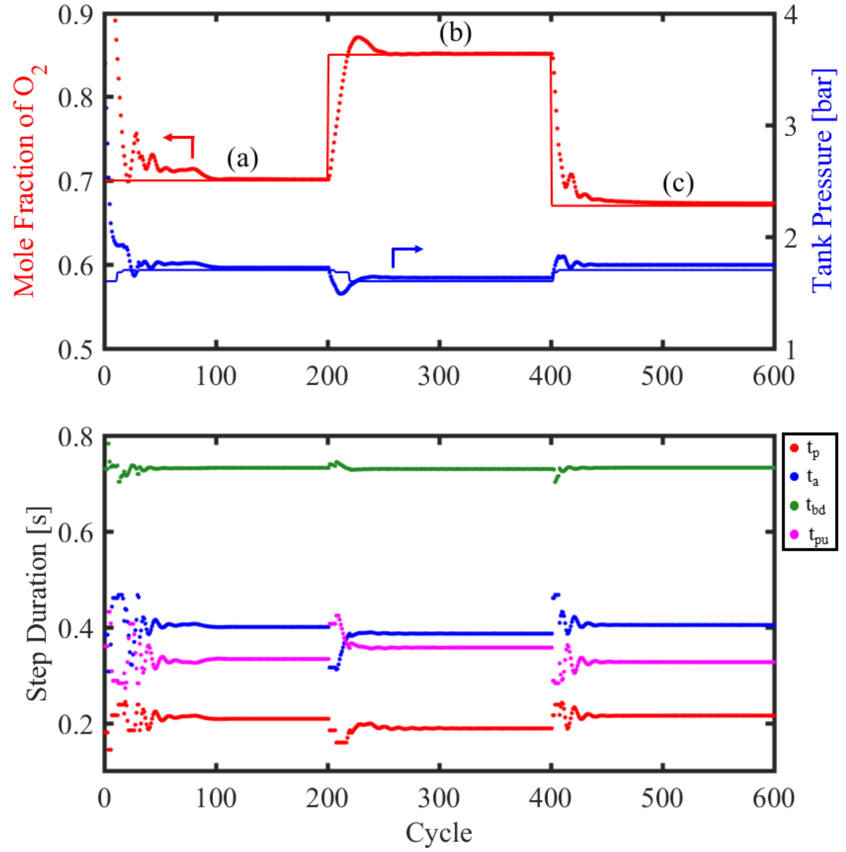


Figure 4.4: M-MPC Set Point Tracking Scenario 1: Set points at operating points (a), region boundaries (b) and non-operating points (c) are tracked by the M-MPC.

occurs automatically based on $\chi_{O_2,T}(k)$, but can be visualized in the example whenever the tank pressure set point changes. The stability of the piece-wise linear model is caused by proper PRBS signal design, and M-MPC tuning.

Further set point tracking scenarios demonstrate the ability of the M-MPC to make large set point changes across many operation regions. Fig. 4.5 shows a step change from 45% O_2 to 90%. The M-MPC switches between five regions before settling at the desired set point with a fast response time and little extraneous movement. Also in this scenario is an example of the trade-off between pressure and composition

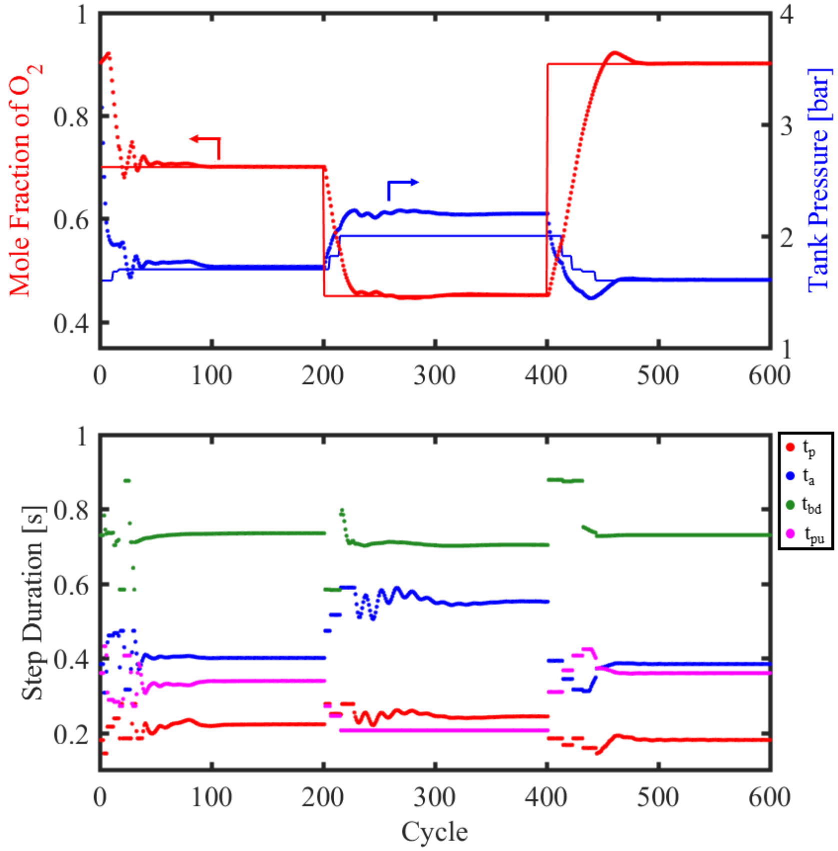


Figure 4.5: M-MPC Set Point Tracking Scenario 2: Large changes in set points require multiple model switching between several regions. The M-MPC easily handles these changes and tracks set points successfully.

set point tracking. The M-MPC is tuned to favor composition, and the pressure set point is treated as a soft constraint to prevent the system from deviating too far from the region operating point. As before, the M-MPC found a previously unknown $u(k = 375)$ which produces the desired 45% O_2 composition automatically. The M-MPC can use the piece-wise linear model to control the RPSA device in the entire nonlinear operating space between 35 and 95% O_2.

4.5.2 Disturbance Rejection

When the set point remains constant, the M-MPC behaves like a local MPC which can reject process disturbances similar to the scenarios shown in Chapter 3. A typical process disturbance to the RPSA device occurs when the product flow rate is changed by the user. In a commercial product, the feed flow rate is set according to the product flow rate, and when one changes, the other is automatically updated via a look-up table. This look-up table is typically based on a easy mass balance of the adsorber column. When this update occurs, unexpected behavior occurs due to nonlinearities in the RPSA device, and $\chi_{O_2,T}$ drops below set point. The M-MPC can automatically adjust u to correct for these small disturbances. An example of this is shown in Fig. 4.6, and the open loop output response without feedback control is shown in green for reference. The M-MPC small changes to u to maintain the required performance when the disturbance occurs. The device is extremely sensitive not only to the value of individual step times, but also the ratio of step times to each other. The M-MPC can independently change each step duration and alter the ratio between steps. This allows the RPSA device to not only produce a wide range of O_2 purities, but also a range of product flow rates.

4.6 Extensions of Piece-Wise Linear MPC

The M-MPC described here was intended for a RPSA device which produces a wide range of O_2 product compositions, and a piece-wise linear model based on this composition was necessary to approximate the nonlinear RPSA plant model. However, this modeling and control strategy can work for RPSA applications produce a single

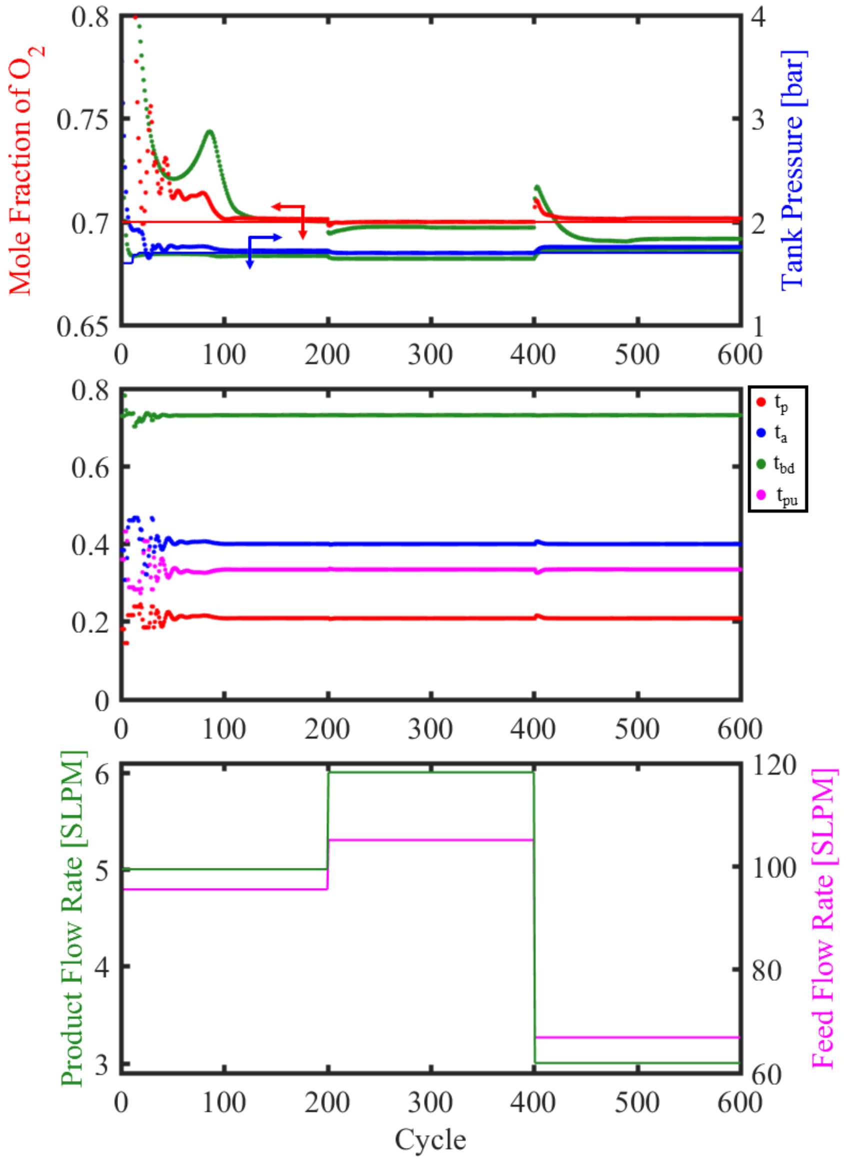


Figure 4.6: M-MPC Disturbance Rejection Scenario: The open loop output response is shown in green for comparison. The M-MPC adjusts u to maintain the O_2 purity.

product composition, but have variability in other process design variables. MOCs only produce a single O_2 product composition, but the product flow rate can vary dramatically based on the patient's medical requirements and the MOC device size. Product flow rate, like product composition, is a highly nonlinear function of the cycle step durations, system pressures and other flow rates, so a single linear model could not accurately predict all of the nonlinear dynamics. If product flow rate is the relevant design variable in designing a M-MPC algorithm, the piece-wise linear model could be designed around this flow rate instead of composition. The M-MPC structure and design procedure would not change significantly, and the M-MPC strategy could be applied just as easily. With this in mind, the M-MPC is a flexible option for different nonlinear aspects of the RPSA design, and can have a wider range of uses.

4.7 Closing Remarks

The M-MPC algorithm presented here works well for the RPSA system because operating points and regions are easily identifiable from online composition measurements. Switching between the piece-wise linear models is also possible using online measurements, and simple switching logic helps in stabilizing the M-MPC performance. Set point tracking is now possible in a wide range of product compositions, which was impossible using a single linear model. Disturbance rejection was also maintained because each local MPC behaves as before, and can reduce or eliminate unwanted dynamics. As before, the performance and stability of the M-MPC will be demonstrated experimentally on a lab-scale RPSA prototype.

Chapter 5

Implementation of an Embedded MPC for a MOC Device

5.1 Motivation

In this chapter, the MPC and M-MPC algorithms are implemented as embedded controllers on a lab-scale RPSA prototype. A Raspberry-Pi® 3 is used to implement the controller, and the algorithm is written using the Python coding language. There are several implementation challenges which are not seen in simulation, and these challenges and solutions are discussed. The MPC design procedure developed in simulation is repeated on the RPSA device, and system identification is performed using experimental data instead of simulation results. The embedded controller makes online control calculations by solving an optimization problem every cycle, and synchronizes these calculations with the RPSA cycle operation. The MPC algorithm is demonstrated experimentally, and evaluated using both set point tracking and disturbance rejection scenarios.

5.2 Design and Operation of the Lab-Scale MOC Device

A novel, single bed, MOC was developed previously to continuously deliver 90% O_2 using a 4-step RPSA cycle [3, 27, 48]. An image of the MOC unit is shown in Fig. 5.1. The device has a single adsorber column concentrically inserted inside a product storage tank. The adsorber column contains a LiX zeolite material which selectively adsorbs N_2 from air. Mass flow controllers are used to define the feed, product and purge flow rates, and a back pressure regulator controls the maximum pressure inside the adsorber column. Four solenoid valves are precisely timed to open and close during the four RPSA cycle steps and direct flow through the system. A Linux-based Raspberry-Pi 3 (R-Pi) system running Python code is used to communicate with the digital flow and pressure controllers, open and close solenoid valves, collect output measurements and make the MPC calculations. Honeywell pressure sensors are used to monitor various pressures, including the pressure of the storage tank, and a zirconium-based O_2 analyzer is used to measure the O_2 mole fraction inside the storage tank.

The MOC unit utilizes a Skarstrom-like [29], 4-step PSA cycle: pressurization, adsorption, blow down and purge. During pressurization, the adsorber column is pressurized to super-atmospheric conditions from the feed end. In the adsorption step, feed air is continuously supplied to the column from the feed end while high purity O_2 enters the storage tank from the product end. During blow down, the exhaust valve is opened, and the adsorber column is exhausted to atmospheric pressure. The enriched-N_2 gas in the column is discarded. In the purge step, some of the high purity

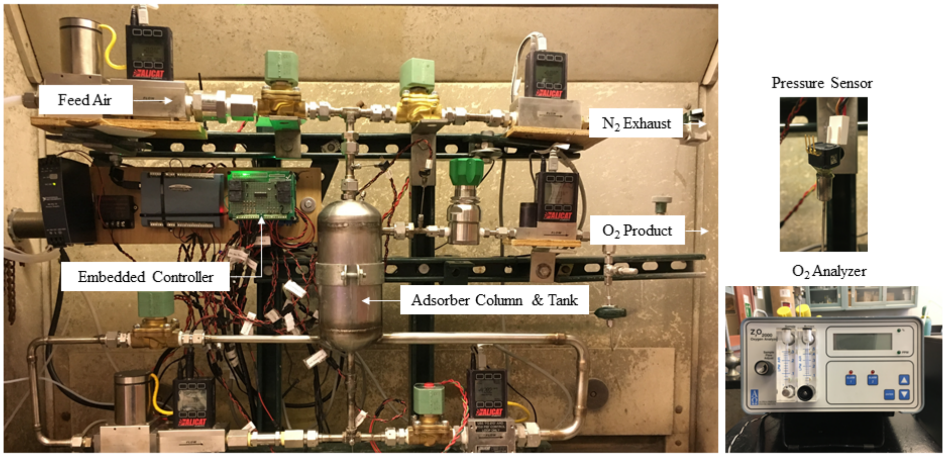


Figure 5.1: Lab-Scale MOC Device with Embedded Controller: The single-bed RPSA device uses a single adsorber column inserted concentrically into a product storage tank. Feed air is supplied from a compressed-air source, and high purity O_2 product exits the storage tank.

O_2 product is used to scrub the adsorber bed and clean the voids. The patient is continuously supplied with high purity O_2 during all cycle steps. Because the RPSA relies on pressure-driven flow, the tank pressure is a critical component of this single-bed design which is not found in traditional multi-column PSA systems. The tank pressure must be controlled by the MPC to ensure both purge flow and product flow at all times.

From a systems perspective, the RPSA cycle has eight variables which affect the dynamic reponse of the MOC device. The four cycle step durations (t_p, t_a, t_{bd}, t_{pu}) must be precisely controlled because they each contribute to the performance in a highly coupled, nonlinear manner. These step durations are manipulated by the MPC to control the MOC device, and considered the system input variables, $u = [t_p, t_a, t_{bd}, t_{pu}]^T$. In a commercial MOC product, the cycle step durations are the only variables available for control because using digital mass and pressure controllers

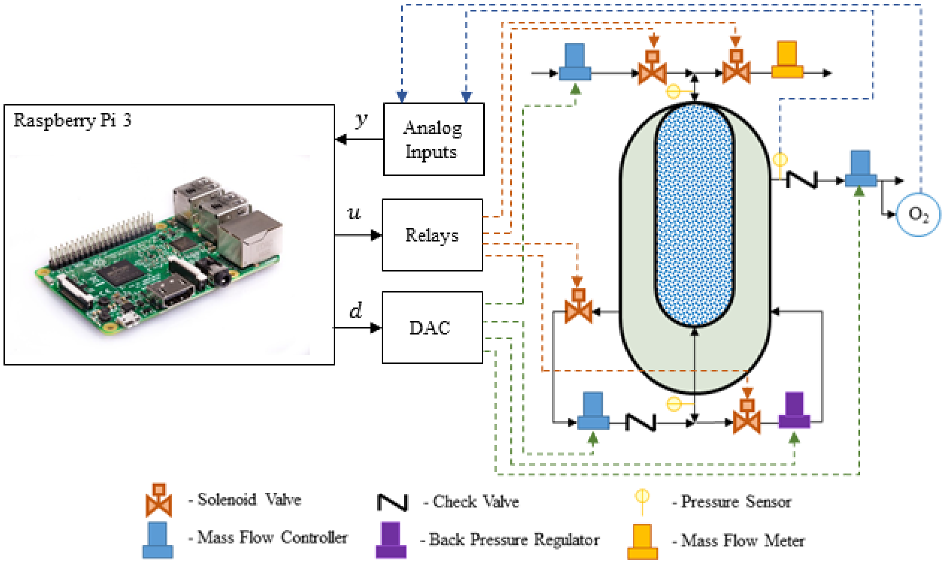


Figure 5.2: Embedded Controller Hardware Schematic: The embedded controller communicates with the RPSA device in three ways. Electrical relays impose the cycle step durations via the timing of the solenoid valves. Digital-to-Analog Converters (DAC) are used to transmit setpoints to the digital controllers, and composition and pressure measurements are sent to the controller via voltage signals. The Raspberry Pi coordinates this process as well as running the MPC algorithm in real-time.

is economically prohibitive. In the lab-scale MOC, these digital controllers are used to design and test the MPC and the MOC device. The remaining four variables are treated as disturbance variables. The purge flow rate (F_p) and adsorption pressure (P_a), feed flow rate (F_f) and product flow rate (F_{O2}) are used to simulate disturbances due to compressor fluctuations, patient needs and changing ambient conditions. The effect of the disturbance variables, $d = [F_f, F_{O2}, F_p, P_a]^T$, can be greatly reduced by the MPC. A variety of valves, digital controllers, sensors and a Raspberry Pi 3 is used to operate the RPSA cycle and run the MPC algorithm.

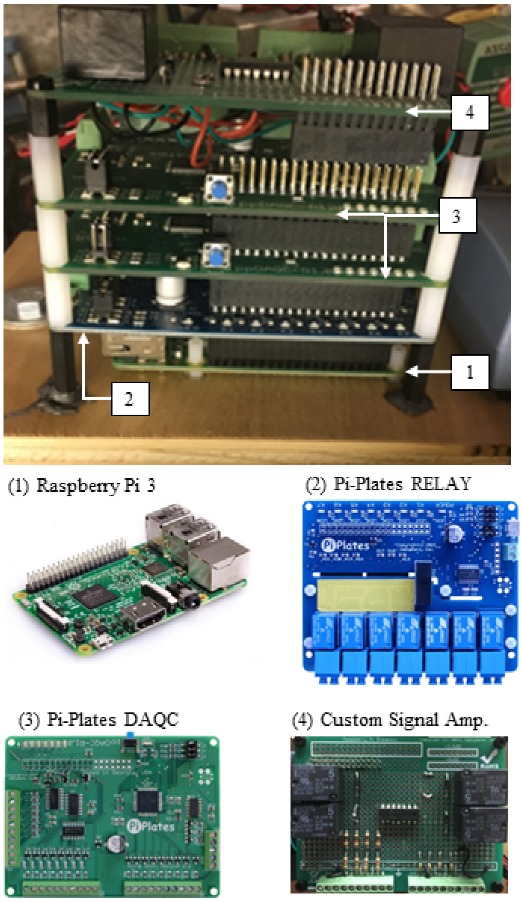
5.2.1 Hardware Requirements of the Device

A Raspberry-Pi® 3 (RPi) micro-computer is used in the RPSA device. It uses a set of electrical relays to precisely time the four 24VDC solenoid valves which direct gas flow during the RPSA cycle. Using digital-to-analog converters (DAC) and a custom-made signal amplification circuit, the RPi sends 0-10VDC set point signals to the four Alicat® digital controllers to maintain the proper flow rates and column pressure in the system. A Honeywell® pressure sensor measures the storage tank pressure, and sends a 0-10VDC signal to the RPi via an analog input channel. A zirconium-based O_2 sensor measures the product composition and communicates with the RPi using a 0-4VDC analog signal. A diagram of the MOC hardware is shown in Fig. **??**. The RPi uses a quad-core 1.2 GHz Broadcom processor with 1 GB of RAM to run Python code which both operates the RPSA cycle and runs the MPC algorithm. The synchronization of these two tasks is a key feature of this work, and vital to successful implementation of the MPC algorithm as an embedded controller.

The embedded controller consists of the RPi connected to several circuit board to expand the RPi Input/Output (I/O) pin capability. Pi-Plates® Relay and Data Acquisition and Control (DAQC) boards increase the I/O pin count and allow the RPi to operate the RPSA cycle and collect output measurements. An image of the embedded controller is shown in Fig. 5.3.

5.3 Model Predictive Control Algorithm

The RPSA system poses a multivariable control problem due to the single-bed device design. The MOC device must produce the required 90% O_2 product purity ($\chi_{O2,T}$)



(1) Raspberry Pi 3

(2) Pi-Plates RELAY

(3) Pi-Plates DAQC

(4) Custom Signal Amp.

Figure 5.3: Components of the Embedded Controller: The Raspberry Pi 3 is connected to additional circuitry to operate the RPSA cycle and run the MPC algorithm. A board with electrical relays is used to open and close solenoid valves. Two data acquisition and control (DAQC) boards are used to collect measured data and send set point signals to digital controllers. A custom circuit is used to amplify the set point signals.

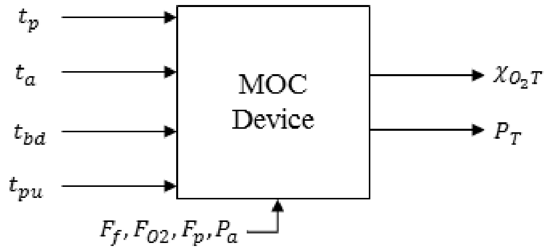


Figure 5.4: Open-Loop Block Diagram of the MOC Device: The four cycle step durations (t_p, t_a, t_{bd}, t_{pu}) are used by the MPC to control the product tank composition and pressure. Disturbances from feed, product and purge flow rates, and the adsorption pressure can disturb the MOC device.

to meet the medical therapy requirement. The total gas pressure in the storage tank (P_T) must also be maintained at super-atmospheric pressure to ensure both product and purge flow occurs in each RPSA cycle. Fig. 5.4 summarizes the multivariable control problem.

Because of its cyclic operation, the RPSA device never reaches a traditional steady state. Instead, a cyclic steady state (CSS) is achieved which occurs when the cycle profiles of all system states reach a constant, time-invariant form. When a disturbance occurs and the CSS condition is broken, undesired "inter-cycle" dynamics occur which breaks the CSS condition. The MPC adjusts the cycle step durations to control $y = [\chi_{O2,T}, P_T]^T$ and operate the RPSA device to the desired performance.

5.3.1 Multivariable MPC Formulation

The multivariable MPC algorithm was developed previously in simulation using a nonlinear plant model for the RPSA system [25]. The MPC uses a quadratic optimization problem and a linear process model identified using sub-space identification techniques around the desired operating point. The MPC optimization problem is given as,

$$
\begin{aligned}
\min_{\substack{u(k+i|k)\\ i=1,2,...,M}} & \sum_{i=1}^{N} [\hat{y}(k+i|k) - r(k+i|k)]^2_{\omega_1} + \sum_{i=2}^{M} [\Delta u(k+i|k)]^2_{\omega_2} \\
& + \sum_{i=1}^{M} [u(k+i|k) - \bar{u}(k+i|k)]^2_{\omega_3} \\
\text{s.t. } & \hat{x}(k+i+1|k) = A\hat{x}(k+i|k) + Bu(k+i|k) \\
& \hat{y}(k+i|k) = C\hat{x}(k+i|k) + Du(k+i|k) \\
& u_{min} \leq u(k+i|k) \leq u_{max}, \;\; i = 1,2,...,M \\
& \hat{y}_{min} \leq \hat{y}(k+i|k) \leq \hat{y}_{max}, \;\; i = 1,2,...,N
\end{aligned}
\tag{5.1}
$$

where $\hat{y}$ is the estimated outputs, $r \in \mathbb{R}^{2x1}$ is the set point, $\Delta u \in \mathbb{R}^{4\times 1}$ is the change in $u(k+i|k)$, $\omega_1 \in \mathbb{R}^{2\times 1}$, ω_2, $\omega_3 \in \mathbb{R}^{4\times 1}$ are weighting vectors. $A \in \mathbb{R}^{7\times 7}$, $B \in \mathbb{R}^{7\times 4}$, $C \in \mathbb{R}^{2\times 7}$ and $D \in \mathbb{R}^{2\times 4}$ are the linear model state space matrices, and $\hat{x} \in \mathbb{R}^{7\times 1}$ is the estimated state vector. u_{min}, $u_{max} \in \mathbb{R}^{4\times 1}$ are lower and upper bounds on u and $\hat{y}_{min}$, $\hat{y}_{max} \in \mathbb{R}^{2\times 1}$ are the lower and upper bounds on $\hat{y}$. The prediction horizon, $N = 50$ cycles, and the control horizon, $M = 20$ cycles, were chosen after extensive controller tuning experiments.

Linear MPC was chosen in this application because of its relatively low computational cost and feasibility to implement into an embedded controller. The RPSA device relies on a highly nonlinear, gas-adsorption based, cyclic process, and detailed process models are too computationally expensive to use in implemented MPC algorithms where the optimization problem is solved in real-time. Because the MOC only operates in a narrow range of O_2, a linear approximation around this point can be made. An experimental data-driven sub-space identification procedure is used to generate this linear process model to be used in the MPC.

5.3.2 Modeling and System Identification

The MPC requires a process model which relates the manipulated inputs, $u = [t_p, t_a, t_{bd}, t_{pu}]$, to the controlled outputs, $y = [\chi_{O2T}, P_T]$. In a commercial MOC, few sensors are included to minimize the manufacturing cost of the device, so the process model cannot also rely on measurable disturbance variables. System identification of nonlinear systems is an active area of research, and many possible solutions exist [49]. In this application, a sub-space identification procedure using Pseudo-Random Binary Sequence (PRBS) signals was used along with the algorithm N4SID [32]. PRBS signals were used because they are easy to implement, and they can continuously excite a system with relatively small amplitudes [31].

Designing proper PRBS signals is critical for system identification, but there is not a definitive design procedure, especially for nonlinear systems. The system identification is performed around a linearization point, $(\bar{u}, \bar{y})$. As described in previous work [25], there are four main guidelines used in PRBS signal design.

- Each signal is contains $n \in \mathbb{I}^+$ perturbations, $u(\zeta T)$, $\zeta \in (1, 2, ..., n)$, of length $T \in \mathbb{I}^+$ cycles.
- Upper and lower bounds are imposed on each signal,

$$u_{min} \leq u(\zeta T) \leq u_{max} \tag{5.2}$$

- The magnitude of each perturbation change must be large enough to excite the

MOC system

$$|u(\zeta T) - u((\zeta - 1)T)| \geq \beta \tag{5.3}$$

- The magnitude of each perturbation change must not be too large to move the MOC from its baseline point

$$|u(\zeta T) - u((\zeta - 1)T)| \leq \gamma \tag{5.4}$$

The four PRBS signal tuning parameters, n, T, β, and γ as well as the input bounds u_{min} and u_{max} can be tested experimentally to best excite the MOC system. This is a labor-intensive process of multiple experimental runs, but the resulting input-output data can then be used to identify a linear model using the sub-space identification algorithm N4SID [32].

For the MOC device, T was the most important parameter to optimize in order to capture the correct dynamics of RPSA process. If T is too small, process delay and overshoot does not allow the outputs to respond to perturbations in the cycle step durations, and if T is too large, the experimental data set does not yield a higher quality model. The optimized values for the PRBS signals are $n = 100$, $T = 5$, $\beta = 0.1$, and $\gamma = 0.2$.

A summary of the baseline, upper and lower bound values for the cycle step durations can be found in Table 5.1. The output baseline values were found to be $\chi_{O2T} = 89\%$ and $P_T = 3.12$ barA. The baseline value for χ_{O2T} does not meet the 90% purity requirement of the MOC device because it is difficult to manually adjust the cycle step durations with enough precision to meet this requirement. With MPC, this difficulty is eliminated with online adjustments to the step durations. Based on

Table 5.1: Summary of Identified Model Linearization Point

	t_p [s]	t_a [s]	t_{bd} [s]	t_{pu} [s]	$\chi_{O2,T}[\%]$	$P_T[barA]$
$\bar{u}$	0.5	2.4	2.5	1.5	–	–
u_{min}	0.5	2.0	2.3	1.0	–	–
u_{max}	0.7	2.6	2.7	1.7	–	–
$\bar{y}$	–	–	–	–	89	3.12
$\hat{y}_{min}$	–	–	–	–	0	1.5
$\hat{y}_{max}$	–	–	–	–	100	4.5

the measured output signals in the PRBS experiment, this baseline is close enough to the requirement for the linear model to make high quality predictions.

The algorithm N4SID was used to identify the linear model used in the MPC. The PRBS data used in the identification is shown in Fig. 5.5. The first 2/3 of the PRBS data set was used to identify the model, and the final 1/3 was used to validate the prediction accuracy. The N4SID algorithm was weighted to favor χ_{O2T} at the expense of P_T. The final identified linear model has the form,

$$\begin{aligned}\hat{x}(k+1|k) &= A\hat{x}(k|k) + Bu(k|k)\\ \hat{y}(k|k) &= C\hat{x} + Du(k|k)\end{aligned} \tag{5.5}$$

The number of estimated states were chosen as part of the N4SID algorithm for the most accurate model predictions to the PRBS data set. The MPC also requires the value of the estimated states for each control calculation. A discrete Kalman Filter was designed such that the observer was stable, $Re\left[\lambda(A - LC)\right] \leq 1$.

$$\begin{aligned}\hat{x}(k+1|k) =&(A - LC)\hat{x}(k|k-1) + (B - LD)u(k|k) + Ly(k|k)\\ \hat{x}(k|k) =&\hat{x}(k|k-1) + M_I(y(k|k) - C\hat{x}(k|k-1) - Du(k|k))\end{aligned} \tag{5.6}$$

Fig. 5.5 shows linear model fit to and prediction of the PRBS data set. Region (a) shows the high quality of the linear model fit to the PRBS data set. The fit of χ_{O2T} is better overall than the fit for P_T due to the heavier weight on χ_{O2T} during the identification procedure. Any residual model error is compensated for using feedback measurements in the MPC.

5.4 Embedded Feedback Controller

The embedded controller has two tasks for the MOC device: operate the RPSA cycle and run the MPC algorithm. The multivariable MPC algorithm must be reformulated so it can be solved in real-time using an efficient optimization solver. The linear process model and Kalman filter are used in each MPC calculation while the RPSA cycle runs continuously.

5.4.1 Implementation of the MPC Algorithm

The MPC formulation in Eqn. 5.1 is not in a usable form for standard solvers, but it can be reformulated as a convex quadratic optimization problem (QP) with linear inequality constraints. The reformulated QP is given as,

$$\begin{aligned} &\min_{U} \frac{1}{2} U^T H U + f(\hat{x}(k|k))^T U \\ &\text{s.t. } \; A_c U \leq b_c(\hat{x}(k|k)) \end{aligned} \tag{5.7}$$

where $u = [u(k+1|k)^T, u(k+2|k)^T, ..., u(k+M|k)^T]^T$ are the control moves manipulated by the MPC, H is the Hessian matrix, $f(\hat{x}(k|k))$ is the gradient vector, A_c is

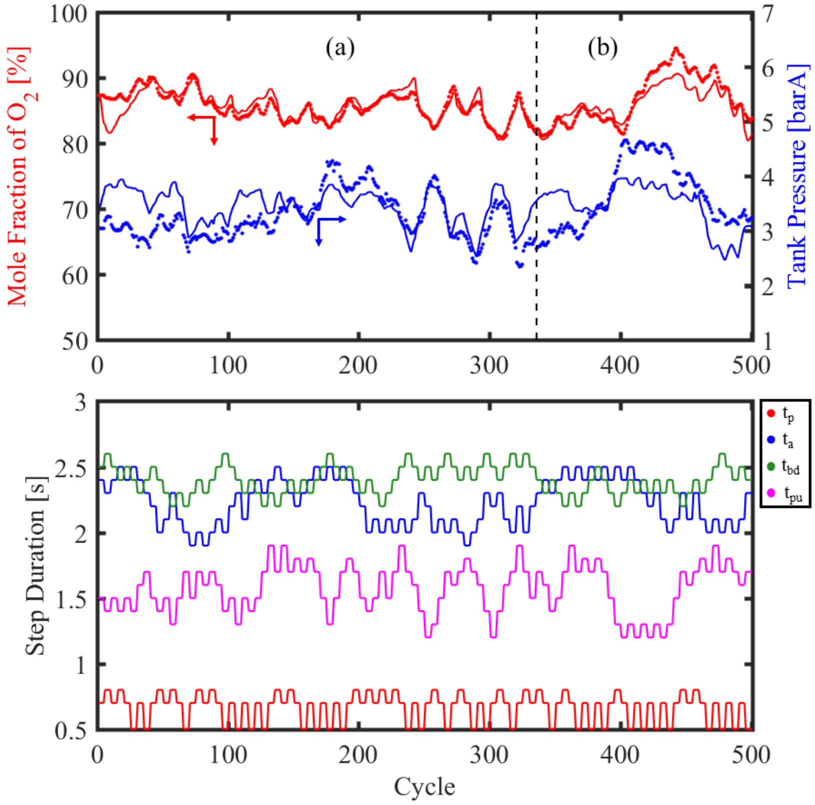


Figure 5.5: Sub-Space Identification using PRBS Input Signals: (Bottom) The four cycle step durations are simultaneously varied using PRBS input signals which excite the MOC controlled output variables. (Top) The controlled variables are measured (dots). Region (a) is used to identify a linear model (solid lines) which relates the four cycle step durations to the O_2 composition and tank pressure. Region (b) is used to validate the linear model prediction.

the constraint matrix, and $b_c(\hat{x}(k|k))$ is the constraint vector. f and b_c are written to highlight the rold of the Kalman filter in Eqn. 5.6 plays in the MPC calculation. Eqn. 5.7 can be solved using standard QP solvers, and the Python library CVXOPT [50] was used in this work.

It is critical to the performance of the RPSA device that the MPC calculations do not interrupt the RPSA cycle for any reason, so certain safe-guards need to be used when implementing the algorithm into the embedded controller. The computation time of the QP solver can be limited by setting a maximum number of solver iterations, and can ensures the MPC finishes before the RPSA cycle concludes. The QP solver may terminate with a non-optimal solution (either due to infeasibility or by reaching the maximum number of iterations). In this case, the algorithm rejects the current solution, and implements the previous cycle step durations.

Sensor dynamics and response times are other factors to consider when implementing the MPC. The zirconium O_2 sensor has a 2% error margin due to its flow-dependent thermal conductivity sensing mechanism which makes offset-free control impossible in the MOC unit. To account for the measurement error, a controller deadband of $d_b = \pm 0.5\%$ was imposed in the MPC algorithm, and a $\pm 1\%$ acceptable region is used all controller evaluation studies. Correspondingly, the composition set point used in the MPC evaluation was set to 91%, so the entire acceptable region remains above the required 90% O_2 composition. The careful choice of the cycle reference point also helps in accurately measuring composition. χ_{O2T} only changes in the first half of the RPSA cycle when gas is added to the tank. The time between the end of adsorption and the end of purge give the sensor more time to respond to the dynamics.

The modified MPC algorithm must be solved in real-time in concert with the continuously running RPSA cycle. This requires careful consideration and coding.

5.4.2 Synchronization of Controller Tasks

The MOC device with the embedded controller operation is summarized in Fig. 5.6. Each of the four RPSA cycle steps are a precisely timed loop where the corresponding valve positions and digital controller set points are sent via the RPi. The cycle step durations for that cycle are collected from the MPC at the beginning of the cycle, and output measurements, collected at the conclusion of the cycle, are communicated back to the MPC for the next calculation. The precise timing of the RPSA cycle cannot be interrupted for any reason, so it is coded as a separate process task on the RPi, with minimal communication during a single RPSA cycle.

In parallel to the RPSA cycle, the RPi makes a control calculation via the MPC algorithm. The previous cycle input and output data is collected, and the Kalman filter in Eqn. 5.6 is solved for $\hat{x}(k|k)$. The gradient and constraint vectors in Eqn. 5.7 are updated, and the QP is solved if the deadband conditions are met. If the solver terminates with an optimal solution, the MPC updates the cycle step durations. Otherwise, the previous durations are used in the next cycle. A key feature of the embedded controller is the real-time calculation of the control actions, which eliminate the need to solve the MPC offline and estimate the solution on hardware.

Because the QP is solved in real-time, there is an inherent 1 RPSA cycle delay in the MPC decisions which must be acknowledged. A typical MPC will implement the first calculated control movement, $u^*(k+1|k)$. To account for the delay, the second control movement, $u^*(k+2|k)$ is used from each MPC calculation.

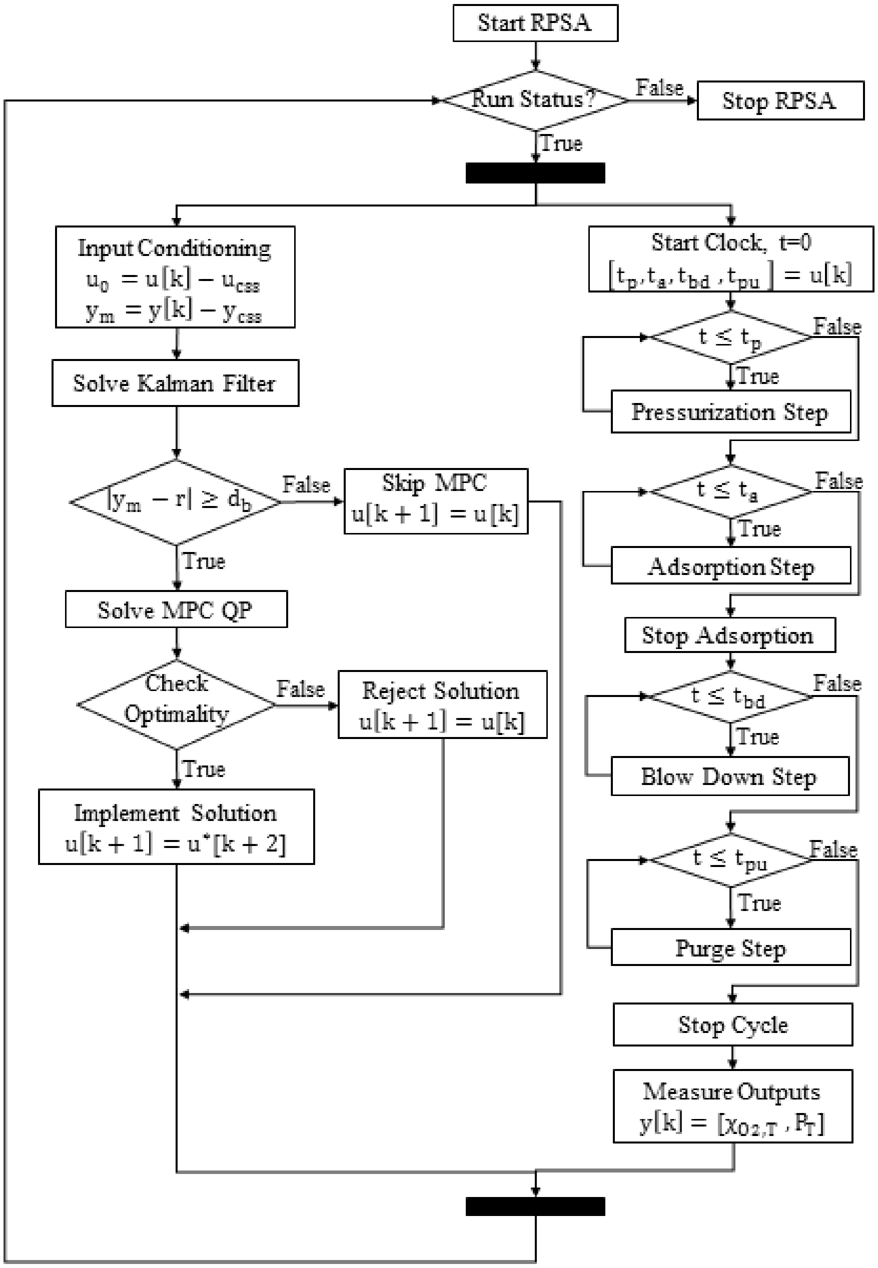


Figure 5.6: Embedded Controller Operation Flow Chart: Operating a single RPSA cycle requires two parallel processes. The MPC calculation is shown in the left half of the flow chart, and operating the RPSA cycle is shown in the right. These processes are dedicated to separate processor threads on the Raspberry Pi to ensure no interference occurs.

5.5 Performance and Evaluation of the MOC System

The embedded MPC was evaluated using both set point tracking and disturbance rejection cases. Examples of each case are described here.

5.5.1 Disturbance Rejection

As a commercial product, MOC devices use the minimum amount of hardware to reduce manufacturing costs and the final retail price of the device. The embedded MPC then has no ability to measure possible disturbances from flow rates, temperatures or other sources, and the MPC must be able to correct for unmeasured process disturbances. The lab-scale MOC does have the ability to manipulate some of these variables in order to simulate possible disturbances and evaluate the MPC performance. In the case shown in Fig. 5.7, F_{O2} is pulsed for approximately 40 cycles which causes a significant decrease in both χ_{O2T} and P_T. This case is shown in Figure 5.7. The pulse in F_{O2} causes a 13% decrease in χ_{O2T} which recovers in 150 cycles while the device is in open-loop mode. In closed-loop mode and the MPC active, χ_{O2T} drops by only 7% and recovers 75 cycles. The duration and magnitude of this disturbance was reduced by 50% with the MPC active. The input signals further validates the efficacy of the MPC. When the disturbance is most severe, the MPC brings the inputs to their imposed limits. Once χ_{O2T} returns to the acceptable region, the MPC adjusts the inputs back to a constant value.

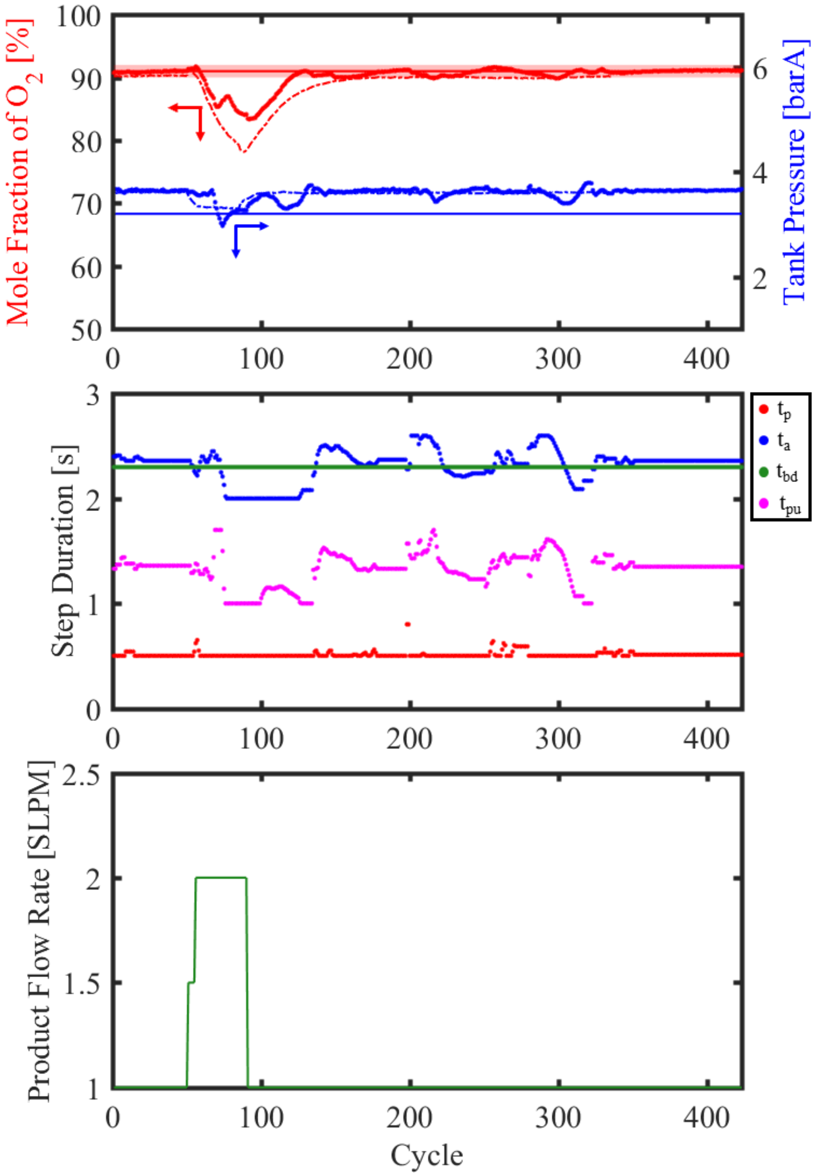


Figure 5.7: Disturbance Rejection using the Embedded MPC: The product flow rate is pulsed from 1 to 2 SLPM for approximately 40 cycles, which causes a significant decrease in O_2 purity without the MPC (red dashed). With the MPC active, the cycle step durations are adjusted, and the disturbance is dramatically improved (red solid).

5.5.2 Set Point Tracking

For the MOC device, the MPC must track a constant purity set point at 91%, but imposing small magnitude set point changes can evaluate the responsiveness of the MPC. Region (a) of Fig. 5.8 shows the MPC bringing χ_{O2T} into the acceptable region in less than 50 cycles, and settling to a constant 91% after 250 cycles. In region (b), three set point changes occur, and the MPC responds well in all cases. When set point changes occur, the MPC significantly changes the RPSA cycle step durations, and brings the MOC to a previously unknown operating point. Finding these new operating points with this degree of accuracy and in this short time would be impossible to manually without feedback control.

There is a limit to how well this MPC responds to set point changes because the linear model becomes less accurate the further the MOC is from its design point. In the last step change in region (b), there is a significantly longer response time before the MOC settles into the acceptable region. As the model becomes less accurate, and the closed-loop response deteriorates until either unacceptable offset occurs or the response time is too long. Based on the step tests shown here, the MPC can track set points above 87% O_2. Even with the longer response time, the MPC is still able to find the correct cycle step durations in a relatively short time.

5.6 Implementation of the M-MPC Algorithm

The M-MPC algorithm presented in Chapter 4 can also be implemented onto the RPSA system using the Raspberry-Pi® hardware. A similar design procedure used in simulation will be repeated using experimental data to generate the piece-wise

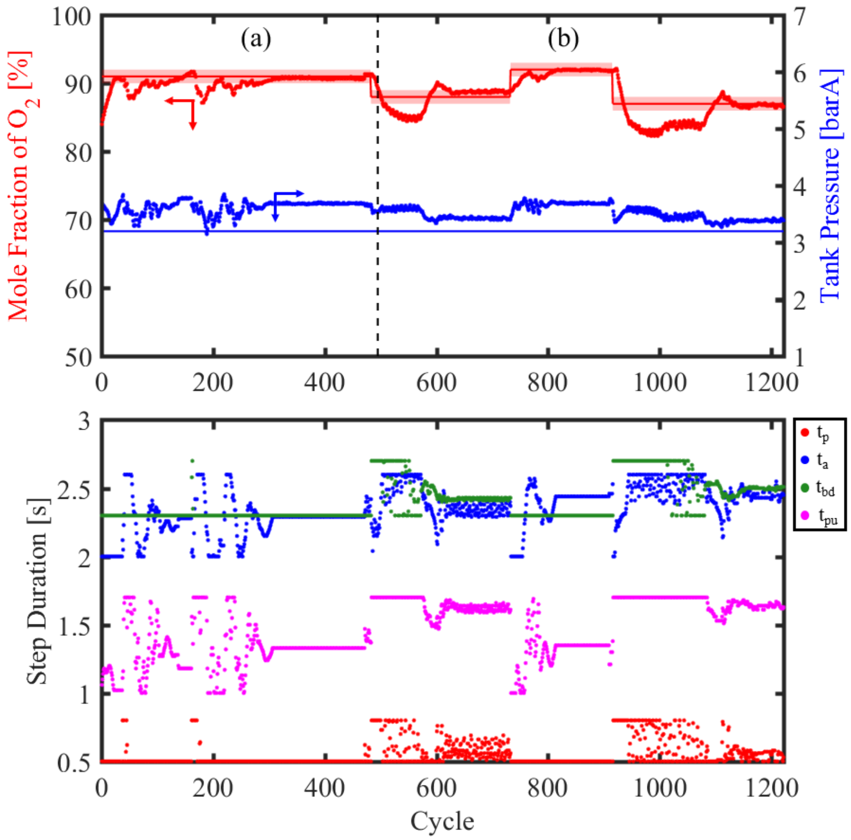


Figure 5.8: Embedded MPC Set Point Tracking Case Study: In region (a), the MPC sucessfully tracks the 91% O_2 set point by adjusting the cycle step durations. In region (b), three set point step changes occur, and the MPC locates the correct cycle step durations which achieve the desired O_2 purity.

Table 5.2: Summary of Operating Region Linearization Points and Boundaries

Region	$\bar{t_p}$ [s]	$\bar{t_a}$ [s]	$\bar{t_{bd}}$ [s]	$\bar{t_{pu}}$ [s]	$\chi_{\bar{O2},T}$ [%]	$\bar{P_T}$ [barA]	χ^+ [%]	χ^- [%]
1	0.50	2.90	2.50	1.70	91.5	3.15	92	86
2	0.80	3.50	2.50	2.10	84.0	3.40	86	80
3	1.00	3.00	2.60	2.30	77.1	3.20	80	72

linear model. The embedded M-MPC is evaluated using set point tracking scenarios.

5.6.1 Operating Regions and Linearization Points

The desired operating space is defined according to $\chi_{O2,T} \in [73\%, 92\%]$. This operating space is broken into $j = 3$ operating regions defined by the baseline input vector, $\bar{u_j}$, the baseline output vector, $\bar{y_j}$, the upper region boundary, χ_j^+, and the lower region boundary, χ_j^-. Each $\bar{u_j}$ are very difficult to determine and must be found manually by experimentally varying all four cycle step durations. The measured $\bar{y_j}$ is the CSS values of y, and each operating region is defined as a range of $\chi_{O2,T}$ in the operating space. A linear model is generated at each linearization point, $(\bar{u_j}, \bar{y_j})$, using sub-space identification techniques and specially designed Pseudo Random Binary Sequence (PRBS) input signals. A summary of the operating regions and linearization points is shown in Table 5.2. In the RPSA system, it is impossible to get very precise $\chi_{O2,T}$ values experimentally because of the highly sensitive, coupled relationship between the cycle step durations and the control variables. The main advantage of the M-MPC is its ability to find new steady states at desired O_2 purities.

5.6.2 Comparison of Linear Identified Models

Each linear model is identified using the PRBS input/output data and the sub-space identification algorithm n4sid [32]. The PRBS data is split into two parts. The first 2/3 of the data set is used to generate the linear model matrices (A_j, B_j, C_j, D_j), and the remaining 1/3 is used to validate the linear model predictions. Predictions of $\chi_{O2,T}$ are weighted more than predictions for P_T because the product composition requirement is more important. Each linear model has the form,

$$\begin{aligned} \hat{x}(k+1|k) &= A_j\hat{x}(k|k) + B_j u(k|k) \\ \hat{y}(k|k) &= C_j\hat{x}(k|k) + D_j u(k|k) \end{aligned} \tag{5.8}$$

where $\hat{x}$ are estimated states which are mathematical constructs and do not directly relate to any physical system states. The number of states was kept constant for all operating regions, and each model has $Re\left[\lambda(A_j)\right] \leq 1$ to be stable.

A discrete Kalman observer is used in the M-MPC algorithm to estimate $\hat{x}(k|k)$ when making control decisions. The observer has the form,

$$\begin{aligned} \hat{x}(k+1|k) &= (A_j - L_jC_j)\hat{x}(k|k-1) + (B_j - L_jD_j)u(k|k) + L_jy(k|k) \\ \hat{x}(k|k) &= \hat{x}(k|k-1) + M_{I,j}(y(k|k) - C_j\hat{x}(k|k-1) - D_ju(k|k)) \end{aligned} \tag{5.9}$$

where L_j is found by solving the corresponding Ricatti equation, and $M_{I,j}$ is used to improve the observer predictions. Each observer is also designed to have $Re\left[\lambda(A_j - L_jC_j)\right] \leq 1$ and be stable.

A single linear model is not adequate to describe the nonlinear dynamics of the

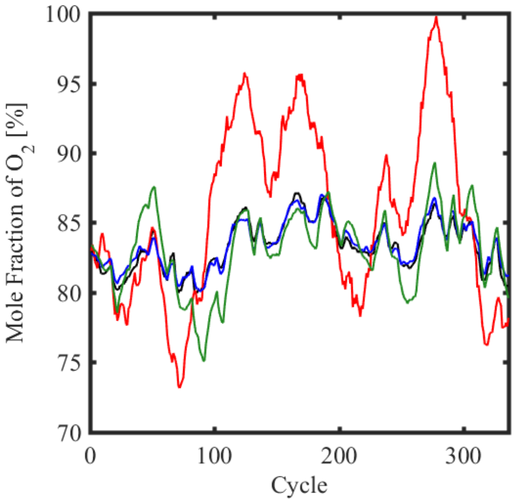


Figure 5.9: A Comparison of Identified Models: The measured $\chi_{O2,T}$ PRBS data from Region 2 (black) is compared to predictions using the identified models for all three regions. Region 2 model has the best prediction quality, while models from the neighboring regions have up to 20% error. The collection of linear models can better predict the nonlienar RPSA system than a single linear model.

RPSA system, and each identified model is only accurate in the region for which it was designed. A comparison between the identified models is shown in Fig. 5.9. The PRBS data from region 2 (shown in black) is used to compare the predictions of the identified models. The model identified for region 2 (blue) gives very accurate predictions, but the models from adjacent regions (red and green) have very poor predictions. Used together, the piece-wise linear model better predicts the nonlinear RPSA dynamics than a single linear model.

5.6.3 Implemented M-MPC Algorithm

The embedded M-MPC makes online control calculations every cycle, and these actions are synchronized in parallel with the RPSA cycle operation. Computational cost is an important factor in implementing the M-MPC quadratic program (QP) in

Chapter 4. As written, standard convex QP solvers can be used, and the QP must be reformulated into a standard form,

$$\begin{aligned} &\min_{U} \frac{1}{2} U^T H_j U + f_j(\hat{x}(k|k))^T U \\ &\text{s.t. } \; A_{c,j} U \leq b_{c,j}(\hat{x}(k|k)) \end{aligned} \tag{5.10}$$

where H and A_c are the hessian and constraint matrices, respectively. f is the gradient vector and b_c is the constraint vector. U is the decision variable vector. $f(\hat{x}(k|k))$ and $b_c(\hat{x}(k|k))$ are written to highlight the effect of the Kalman observer on the QP. The Python package CVXOPT [50] is used on the RP3 to solve Eqn. 5.10 as the RPSA cycle operates.

During each cycle, the M-MPC algorithm determines the current operating region via the $\chi_{O2,T}$ measurement. The local linear model, Kalman observer and MPC parameters are updated accordingly. The Kalman observer is solved to provide an initial condition for the MPC QP. The QP in Eqn. 5.10 is solved, and the solution is implemented if the solution is optimal. A summary of the implemented M-MPC algorithm is shown in Fig. 5.10. In this implementation, there is an inherent 1 RPSA cycle delay between output measurement and implementation of the control action. To account for this, the M-MPC implements the second optimal control action, $u^*(k+2|k)$ instead of the standard $u^*(k+1|k)$. The composition sensor is accurate to within ±2%, so a dead band of $d_b = 2\%$ is implemented into the embedded controller. The QP is solved only when the measured $\chi_{O2,T}$ is outside this dead band.

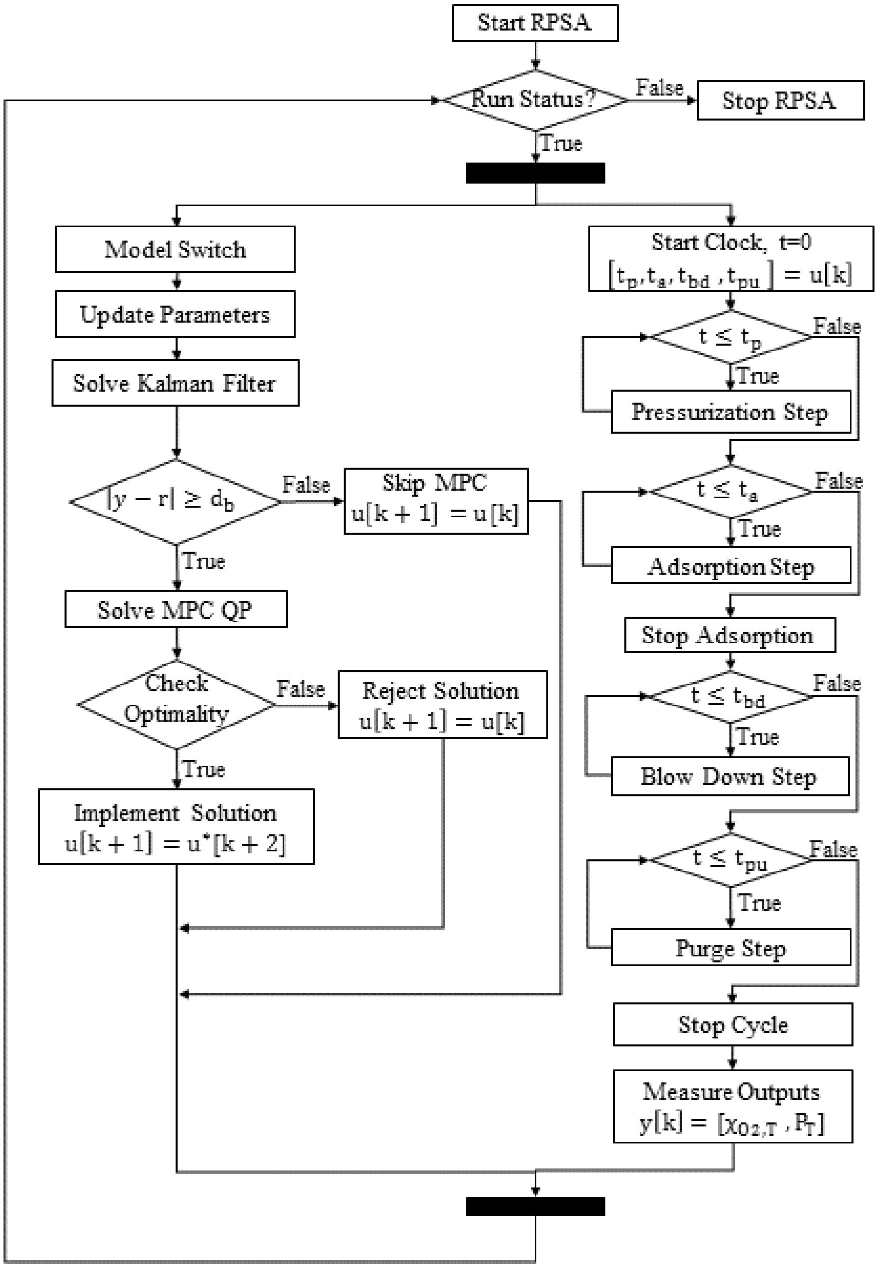


Figure 5.10: Embedded M-MPC Algorithm Flowchart: This flow chart summarizes the actions of the embedded controller in a single RPSA cycle. On the left, the M-MPC collects output measurements, determines the correct model, and makes a control decision. On the right, the embedded hardware operates the RPSA cycle by precisely timing solenoid valves and digital controllers.

5.6.4 Evaluation of the M-MPC

The embedded M-MPC is evaluated using several set point tracking cases across multiple operating regions to demonstrate both the set point tracking and model switching required by the M-MPC algorithm. Fig. 5.11 shows three set point tracking scenarios. Set points for $\chi_{O2,T}$ at 90%, 75% and 84% are tracked sucessfully by M-MPC within the acceptable control region. The acceptable region of ±2% is determined by the accuracy of the composition sensor. After the step change at cycle 400, the M-MPC switches models twice before the RPSA system operates in the same region as the new set point at 75%. After the second step change at cycle 800, the M-MPC only requires one model switch. In all cases, switching is smooth, and does not have a negative affect the output performance. The most important function of the M-MPC is automatically finding a u which corresponds to the desired set point value. The set points at 90% and 75% do not correspond to the known linearization points in Table 4.1, and $u(k = 250)$ and $u(k = 650)$ were found by the M-MPC. In this manner, the M-MPC can find new operating points between 73-92% O_2.

5.7 Closing Remarks

The embedded controller presented here performs two tasks for the RPSA device: operate the RPSA cycle and solve the MPC to make control decisions. The Raspberry-Pi® was an excellent platform to implement the MPC algorithm because of its flexibility and use of the Python coding language. The biggest challenge in using MPC for the RPSA system is coordinating the cyclic operation with online control calculations. Both the MPC and M-MPC algorithms were experimentally demonstrated

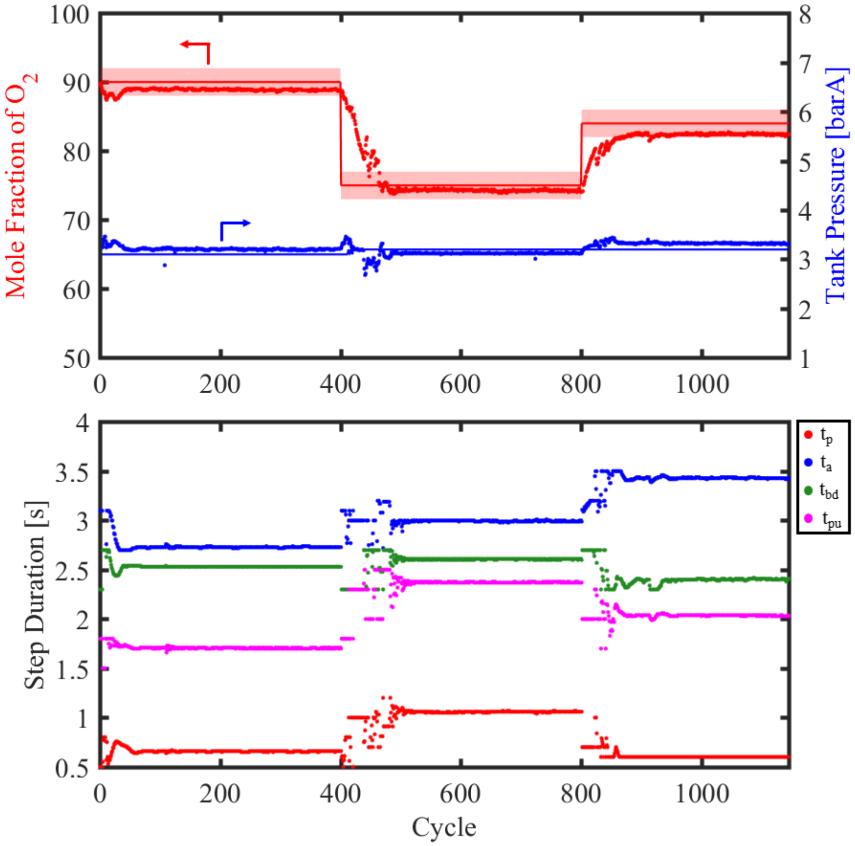


Figure 5.11: Evaluatation of the M-MPC using Set Point Tracking Scenarios: Three set points are considered at 90%, 75%, and 84%. Each set point exists in different operating regions of the RPSA device. In each case, the M-MPC tracks the desired set point to within the acceptable region (red shaded areas) by finding new cycle step durations.

to control the RPSA device, and improve its performance when subjected to process disturbances. This embedded controller can also be designed for future RPSA prototypes as commercialization of the single-bed design progresses.

Chapter 6

Steady State Optimization of MOC Devices

6.1 Motivation

In this chapter, steady state optimization from both a modeling and experimental point of view is discussed. Steady state optimization is widely studied in the literature, but the studies typically use a plant model to simulate a PSA system. Each PSA model uses different simplifying assumptions depending on the specific objective of the study. The plant model for the single-bed RPSA device which was used in this work has several short comings which preclude it from being used in an optimization study, because it cannot be accurately reconciled with experimental data. Several enhancements to the current model are proposed, as well as other additions which could be made in the future. Model-free optimization approaches which are not seen in the current RPSA literature have several advantages over model-based optimization. An experimentally-based approach using online process measurements is proposed to aid in RPSA device optimization.

6.2 Steady State Optimization of the RPSA Cycle

In MOC optimization, there are many variables that contribute to a so-called "optimal" MOC device, which change depending on the specific goal such as optimal cycle, minimal power consumption, etc. Steady state optimization searches for a CSS condition which satisfies all MOC device requirements, and meets all performance objectives. For the MOC device, there are eight possible decision variables which may or may not be utilized in a given objective. The possible decision variables: cycle step durations (t_p, t_a, t_{bd}, t_{pu}), feed flow rate (F_f), product flow rate (F_{O2}), purge flow rate (F_p), and adsorption pressure (P_a) are either used as variables or held constant. The distinction for each presented case study is detailed later. In this steady state optimization, the RPSA system pressure profiles in a single cycle are used to define the minimal characteristics of the RPSA cycle. Fig. 6.1 shows CSS pressure profiles for the inlet (P_{in}), outlet (P_{out}) and tank (P_T) in the RPSA cycle. From these profiles, five requirements can be quantified mathematically which will be formulated into constraints in the model-free optimization problem.

(1) At the end of the adsorption step, there must be a small pressure drop between the product end of the column and the storage tank,

$$(P_{out} - P_T)|_{t=t_p+t_a} \geq 0$$

(2) The feed end of the column must reach at least the adsorption pressure by the

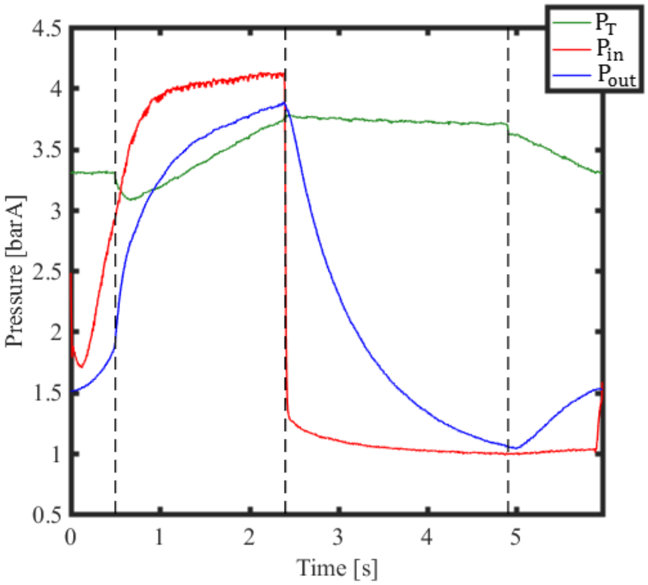


Figure 6.1: RPSA Pressure Profiles at CSS: The three measured system pressures in the RPSA device: inlet (P_{in}), outlet (P_{out}) and tank (P_T) are shown at CSS. Cycle steps are denoted by dashed lines, and optimization constraints are constructed at the end of certain cycle steps.

end of the adsorption step,

$$P_{in}|_{t=t_p+t_a} \geq P_a$$

(3) The pressure drop through the column at the end of the adsorption step should not be too large,

$$0 \leq (P_{in} - P_{out})|_{t=t_p+t_a} \leq \epsilon_1$$

where ϵ_1 is a tuning parameter.

(4) The pressure at the product end of the column must reach atmospheric pressure at the end of the blow down step,

$$P_{out}|_{t=t_p+t_a+t_{bd}} = P_{atm}$$

(5) The tank pressure must be above a certain pressure at the end of the purge step to ensure product and purge flow throughout the cycle,

$$P_T|_{t=t_{cycle}} \geq \epsilon_2 > P_{atm}$$

where ϵ_2 is a tuning parameter.

The performance requirements for the MOC device can also be formulated mathematically as constraints,

(6) The product composition must achieve the desired requirement,

$$\chi_{O2,T}|_{t=t_{cycle}} = \epsilon_3$$

where ϵ_3 is a tuning parameter.

Obviously, these requirements are specific the 4-step RPSA cycle used in this work, but different RPSA cycle characteristics can also be formulated into constraints in a similar manner. These decision variables and constraints can then be formulated into an optimization problem.

6.2.1 Formulating Model-Free Optimization Problems

To maintain generality for all optimization case studies, several definitions are used in the problem formulations. The possible decision variables, $D = [t_p, t_a, t_{bd}, t_{pu}, F_f, F_{O2}, F_p, P_a]$, will be either free variables, u, that the solver can manipulate, or they can be constants, d. The variables which affect the characteristics of the RPSA cycle are designated as outputs, $y = [P_{in}, P_{out}, P_T, \chi_{O2,T}]$. The outputs are an unknown function of u and d according to the nonlinear RPSA dyanmics,

$$y = \mathcal{H}(u, d)$$

where $\mathcal{H}$ represents an unknown, nonlinear function. A generic optimization problem can be expressed as,

$$
\begin{aligned}
\min_{u} \quad & J(u,d,y) \\
s.t. \quad & y = \mathcal{H}(u,d) \\
& g(u,d,y) \leq 0 \\
& q(u,d,y) = 0 \\
& u_{min} \leq u \leq u_{max}
\end{aligned}
\tag{6.1}
$$

where J is the objective function, g is a vector of inequality constraints, q is a vector of equality constraints and u_{min}, u_{max} are bounds on the decision variables. In model-free optimization, $\mathcal{H}$ is unknown, and the constraints cannot be expressed analytically to the optimization solver. The constraints $g(u,d,y)$ and $q(u,d,y)$ are added to the objective function as exact penalty functions. The objective function, J, can be written as,

$$
J(u,d,y) = f(u) + \mu \left(\sum_{i=1}^{N_{ineq}} \max\left[0, g_i\right] + \sum_{i=1}^{N_{eq}} |q_i| \right)
$$

where $f(u)$ is a known function of only the decision variables. The optimization problem can be rewritten as,

$$
\begin{aligned}
\min_{u} \quad & f(u) + \mu \left(\sum_{i=1}^{N_{ineq}} \max\left[0, g_i\right] + \sum_{i=1}^{N_{eq}} |q_i| \right) \\
s.t. \quad & u_{min} \leq u \leq u_{max}
\end{aligned}
\tag{6.2}
$$

Standard, nonlinear optimization algorithms can be used to solve this type of problem. In this work, a Sequential Least Squares Program (SLSQP) solver from the Python library SciPy is used.

6.2.2 Real-Time Optimization using Embedded Hardware

The embedded hardware used in the MPC implementation can be expanded and modified to perform steady state optimization on the lab-scale device. The Raspberry Pi hardware will perform three tasks in this application: operate the RPSA cycle, collect pressure and composition measurements, and run a nonlinear optimization solver using standard Python libraries. Additional sensors for inlet and outlet pressures allow the Raspberry Pi to measure all the output variables necessary in Eqn. 6.2. The chosen nonlinear solver is the Python SciPy implementation of a SLSQP algorithm.

A summary of the model-free optimization approach using the RPSA device is shown in Fig. 6.2. The nonlinear solver evaluates a given objective function which uses the measured outputs from the RPSA device. Each objective function evaluation will run the RPSA cycle with the current values of the decision variables for a specified number of cycles, N, and the required output measurements are collected every cycle. After N cycles, objective function value is calculated and sent back to the solver. $N = 50$ cycles must be chosen to allow the RPSA device to reach CSS, but not so long as to make the function evaluations take too much time. The solver uses these function evaluations to estimate the jacobian using a finite difference approximation, computes a search direction and checks optimality conditions. Once the conditions are satisfied, the solver terminates and returns the optimal solution.

Further control parameters on the optimization solver can be specified to improve the solution quality and computation time. A maximum number of iterations prevents

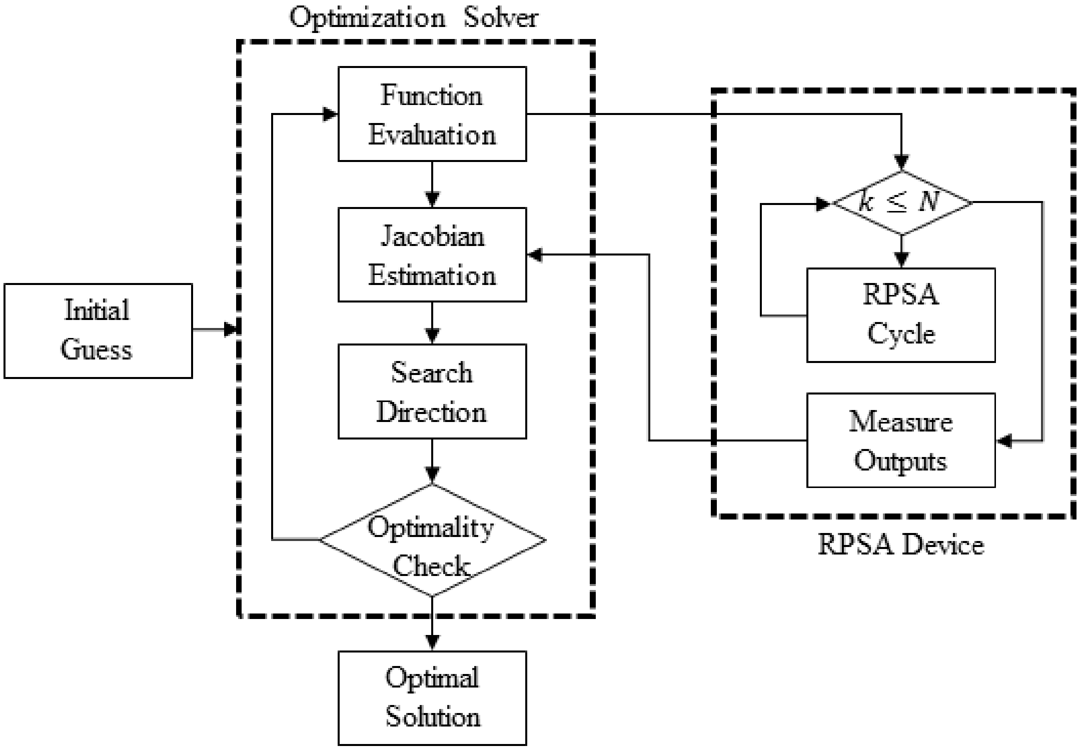


Figure 6.2: Steady State Optimization with Embedded Hardware: The nonlinear optimization solver requires objective function evaluations, and jacobian approximations to find new search directions. The objective function evaluations use the output measurements from the embedded hardware after running a specified number of cycles (N). The solver iterates this process until optimality conditions are met, and terminates with an optimal solution.

the solver from long computation time. A tolerance on the objective function can terminate the solver when the objective does not improve significantly over several iterations. A minimum jacobian evaluation step size ensures changes in decision variables are large enough to elicit a measurable change in the RPSA device. Bounds on the decision variables prevent the solver from improperly operating the RPSA device.

6.3 MOC Optimization Case Studies

The general optimization problem in Eqn. 6.2 is written to fit any specific objective in MOC device optimization. Three cases are shown to demonstrate how different objectives change the optimization formulation.

6.3.1 Case 1: Baseline Identification for MPC Design

In the design of the single linear model used the multivariable MPC or in the design of the piece-wise linear model for M-MPC, a known baseline or linearization point is required before performing system identification. This was previously done experimentally by manually manipulating the RPSA variables until a desired purity is achieved, but in steady state optimization, this procedure can be automated. The objective of this case study is to locate a set of cycle step duration which achieves the desired product composition. The inputs and constants are defined as,

$$u = [t_p,\ t_a,\ t_{bd},\ t_{pu}]$$

$$d = [F_f,\ F_{O2},\ F_p,\ P_a]$$

From knowledge of the RPSA cycle performance, bounds on the total cycle time are imposed in the optimization problem,

$$t_{cycle,min} \leq t_p + t_a + t_{bd} + t_{pu} \leq t_{cycle,max}$$

In the MPC design, a required purity of $\epsilon_3 = 90\%$ is used in the optimization problem to locate the correct baseline, while in the M-MPC design, ϵ_3 is changed to reflect different composition requirements. The optimization problem is given as,

$$\begin{aligned} \min_{u} \quad & \mu \left(\sum_{i=1}^{N_{ineq}} \max\left[0, g_i\right] + \sum_{i=1}^{N_{eq}} |q_i| \right) \\ s.t. \quad & t_{cycle,min} \leq t_p + t_a + t_{bd} + t_{pu} \leq t_{cycle,max} \\ & u_{min} \leq u \leq u_{max} \end{aligned} \tag{6.3}$$

The objective in this case can be considered a “feasibility” problem, because the objective function only includes the constraint penalty function terms. Any set of cycle step durations which satisfies the constraints and the purity requirements is considered optimal. A series of experimental optimization runs were performed for this case study. The results are summarized in Table 6.1. In each run, a target O_2 composition was set, and the optimization solver varied the cycle step durations until

Table 6.1: Optimization of RPSA Cycle Timing for Various O_2 Compositions

Target $\chi_{O2,T}$	$\bar{t_p}$ [s]	$\bar{t_a}$ [s]	$\bar{t_{bd}}$ [s]	$\bar{t_{pu}}$ [s]	$\bar{\chi_{O2,T}}$
90	0.50	2.34	2.44	1.14	89.51
85	0.50	2.12	2.42	0.92	84.21
80	0.60	2.37	2.43	1.17	79.57

the objective function in 6.3 was minimized. In each scenario, the solver was able to achieve the desired purity within a ±1% error which is within the tolerance of the composition sensor.

6.3.2 Case 2: Product Flow Rate Maximization

In this scenario, the RPSA must produce the maximum product flow rate while still producing the required product composition. When a device is constructed, the cycle step durations, feed flow rate and product flow rate are manually manipulated to maximize the product flow rate. A secondary objective in this scenario is to minimize the feed flow rate because this directly relates to the feed compressor specifications necessary in a commercial MOC product. The decision variables and constants in this situation can be defined as,

$$u = [t_p,\ t_a,\ t_{bd},\ t_{pu},\ F_f,\ F_{O2}]$$

$$d = [F_p,\ P_a]$$

The objective function seeks to minimize the feed flow rate, maximize product flow rate and penalize constraint violations.

$$J(u,d,y) = -\phi_1 F_f + \phi_2 F_{O2} + \mu \left(\sum_{i=1}^{N_{ineq}} \max\left[0, g_i\right] + \sum_{i=1}^{N_{eq}} |q_i| \right)$$

where $\mu >> \phi_2 > \phi_1$ are chosen tuning parameters to ensure the variables and constraints are prioritized correctly. The complete optimization problem is formulated as,

$$\begin{aligned} \min_{u} \quad & \phi_1 F_f - \phi_2 F_{O2} + \mu \left(\sum_{i=1}^{N_{ineq}} \max\left[0, g_i\right] + \sum_{i=1}^{N_{eq}} |q_i| \right) \\ s.t. \quad & t_{cycle,min} \leq t_p + t_a + t_{bd} + t_{pu} \leq t_{cycle,max} \\ & u_{min} \leq u \leq u_{max} \end{aligned} \tag{6.4}$$

This scenario cannot be experimentally demonstrated with the current hardware configuration because the actual flow rate measurements cannot be read by the Raspberry Pi. Improvements to the data acquisition circuitry could enable the Raspberry Pi to make these measurements.

6.3.3 Case 3: Power Consumption Minimization

In this final scenario, a preliminary optimization of power consumption is considered by relating the available decision variables to feed compressor specifications. The size, weight and other specifications of a compressor are directly related to the discharge pressure and flow rate requirements. These become the feed flow rate and adsorption pressure of the RPSA device. Minimizing these variables, while maintaining the performance requirements, enables a smaller, lighter compressor to be used in a commercial MOC device. The decision variables and constants in this scenario are given as,

$$\begin{aligned} u &= [t_p,\ t_a,\ t_{bd},\ t_{pu},\ F_f,\ P_a] \\ d &= [F_{O2},\ F_p] \end{aligned}$$

The objective function is similar as Case 2, and the optimization problem can be written as,

$$\begin{aligned} \min_{u} \quad & \phi_1 F_f + \phi_2 P_a + \mu \left(\sum_{i=1}^{N_{ineq}} \max\left[0, g_i\right] + \sum_{i=1}^{N_{eq}} |q_i| \right) \\ s.t. \quad & t_{cycle,min} \leq t_p + t_a + t_{bd} + t_{pu} \leq t_{cycle,max} \\ & u_{min} \leq u \leq u_{max} \end{aligned} \tag{6.5}$$

where $\mu >> \phi_1$, ϕ_2 are similar tuning parameters, but ϕ_1 and ϕ_2 may have similar values. Similar to Case 2, this problem cannot be experimentally demonstrated due to the same hardware limitations, and is shown only as an illustration.

6.4 Future Directions in MOC Optimization

The embedded hardware is a versatile tool for steady state optimization, and model-free optimization is a excellent option which requires little development time. Its main disadvantage is a lack of prediction capability as it relies only on function evaluations and output measurements to gauge optimality. One possible improvement could utilize machine-learning techniques to generate a high quality process model using online measurements from the RPSA device. Another option is to use a detailed process model such as Eqn. 2.1-2.7, but significant error occurs between this plant model and the RPSA device. A data-reconciled detailed process model is the ideal option to use in RPSA optimization. Detailed nonlinear process models have a high computational cost, and would require more powerful computing resources to solve. The Raspberry Pi® hardware would have to be reconfigured to communicate, possibly wirelessly, with a computing station which would run the nonlinear model and optimization solver. The Raspberry Pi® would operate the RPSA process while communicating measurements to reconcile with the process model. These improvements could help reduce the iterations required in a solver algorithm and provide better optimal solutions than a model-free approach. Improvements to the nonlinear RPSA model will reduce the model error in the current model.

6.5 Advances in RPSA Modeling for Optimization

A new three-component RPSA model improves on the model used in the MPC design, and it could be used in future optimization studies for the RPSA device. The model simulates a dry, synthetic air mixture of oxygen, nitrogen and argon. Argon is used to model all inert species present in ambient air. The column mass balance equations are expanded to include argon,

$$
\begin{aligned}
\frac{\partial \rho_g}{\partial t} &= \frac{\bar{\epsilon}}{\epsilon} D_L \frac{\partial^2 \rho_g}{\partial z^2} - \frac{1}{\epsilon}\frac{\partial Q}{\partial z} - \frac{\rho_b}{\epsilon} \sum_{i \epsilon \{O_2, N_2\}} \frac{\partial n_i}{\partial t} \\
\frac{\partial (\chi_{N2} \rho_g)}{\partial t} &= \frac{\bar{\epsilon}}{\epsilon} D_L \frac{\partial^2 (\chi_{N2} \rho_g)}{\partial z^2} - \frac{1}{\epsilon}\frac{\partial (\chi_{N2} Q)}{\partial z} - \frac{\rho_b}{\epsilon}\frac{\partial n_{N2}}{\partial t} \\
\frac{\partial (\chi_{Ar} \rho_g)}{\partial t} &= \frac{\bar{\epsilon}}{\epsilon} D_L \frac{\partial^2 (\chi_{Ar} \rho_g)}{\partial z^2} - \frac{1}{\epsilon}\frac{\partial (\chi_{Ar} Q)}{\partial z} - \frac{\rho_b}{\epsilon}\frac{\partial n_{Ar}}{\partial t} \\
1 &= \chi_{N2} + \chi_{O2} + \chi_{Ar}
\end{aligned}
\tag{6.6}
$$

The mass transfer between the gas and solid phases using Langmuir Isotherms is also expanded to include argon,

$$
\begin{aligned}
\frac{\partial n_{N2}}{\partial t} &= k_{N2}(n_{N2}^{\infty} - n_{N2}) \\
\frac{\partial n_{O2}}{\partial t} &= k_{O2}(n_{O2}^{\infty} - n_{O2}) \\
\frac{\partial n_{Ar}}{\partial t} &= k_{Ar}(n_{Ar}^{\infty} - n_{Ar})
\end{aligned}
\tag{6.7}
$$

The gas and solid phase energy balance are similar, and include heat effects from all components as before. Momentum balances and pressure drop are calculated as before. The storage tank component balances are expanded to correspond to the boundary conditions for the column component balances.

$$
\begin{aligned}
\frac{d(\chi_{N2,T} P_T)}{dt} &= \frac{F}{V_T}((\chi_{N2} P)|_{z=L,t} - \chi_{N2,T} P_T) - \frac{F_P}{V_T}\chi_{N2,T}(P_T - P_{atm}) \\
\frac{d(\chi_{Ar,T} P_T)}{dt} &= \frac{F}{V_T}((\chi_{Ar} P)|_{z=L,t} - \chi_{Ar,T} P_T) - \frac{F_P}{V_T}\chi_{Ar,T}(P_T - P_{atm}) \\
1 &= \chi_{N2,T} + \chi_{O2,T} + \chi_{Ar,T}
\end{aligned}
\tag{6.8}
$$

The addition of three equations increases the computational cost of the model, and so far has led to many numerical instabilities. The advantage to this latest model is the ability to capture the theoretical limit of 95% O_2 product composition, which becomes more important in steady state optimization than it was in MPC design.

Although this is the most detailed model for RPSA systems to date, the model still has several limitations which could be improved for use in any RPSA device optimization. The most significant shortcomings of this latest model can be summarized by these points.

(1) Water contamination of the adsorber material remains an unmodeled phenomena. Modeling water contamination can lead to improved prediction of the zeolite (and device) lifespan and a precise mass of zeolite material. Modeling water is challenging for several reasons. Including another gas species will increase the number of equations, and exacerbate the numerical instabilities which are already present. Water contamination also occurs at a much slower rate, and bed contamination could take months worth of RPSA cycles. An alternative to modeling water in the RPSA system is to model a desiccant layer in addition to the zeolite adsorber column, which could reduce the computational costs of this issue.

(2) The back pressure regulator is an important component of the lab-scale RPSA device because it adds resistance through the adsorber column, and improves the overall device performance. Modeling the regulator is difficult because its dynamics depend on the column and tank pressure solved for in the model. One potential modeling strategy is to define the end of the pressurization step as the point where the column pressure reaches the adsorption pressure. This

is problematic because now the model has an implicit problem not compatible with standard PDE solvers.

(3) A commercial RPSA device relies entirely on a compressor for feed gas supply, and the dynamics of the discharge pressure and flow rate has an enormous effect on the RPSA device performance. Modeling the compressor gives a more accurate portrayal of a commercial device, and could also be used to determine an optimal compressor specifications for a future device manufacturer.

Using a detailed nonlinear model in steady state optimization has the best chance of reliable, high-quality solutions to RPSA device designs, but this model must be validated with experimental measurements. Even this latest process model is not able to accurately predict the dynamics of the lab-scale RPSA unit. Until the model is validated, a model-free optimization strategy is the best option.

6.6 Closing Remarks

The model-free optimization approach shown here is an excellent option for the RPSA system because detailed plant models are difficult to accurately reconcile with experimental data. The optimization scenarios shown here can be modified or extended to meet a wide variety of objectives, but the model-free approach is not without limitations. Without a mathematical model, the optimization solver will most likely have difficulty with very complex objective functions and too many decision variables. The eventual goal should be to use a nonlinear optimization solver with a detailed plant model in a ”data-reconciled optimization formulation” where the model is compared to measured variables as the optimization problem is solved. This is a complex, difficult problem, and will take further development to solve. Improving the plant model in the ways explained here is a good start to reconciling the model with actual RPSA performance.

Chapter 7

Conclusions and Future Directions

7.1 Model Predictive Control for the RPSA System

PSA technology is a common industrial and commercial technology which is widely studied in the current literature. However, almost all studies focus on the modeling of PSA systems for steady state optimization, and dynamic modeling and control is almost nonexistent in the literature. These systems are complex, nonlinear, cyclic processes subject to flow reversals, discrete valve switching and adsorption effects which need some form of feedback control for reliable operation. The multivariable control strategy presented in this work builds on the current state of the art, and presents a new approach to RPSA control using MPC. Independently manipulating the cycle step durations is the key feature of the MPC strategy because the controller can better utilize the coupled relationships between the step durations and control variables. MPC was shown both in simulation and on a lab-scale unit to improve the

RPSA performance by rejecting process disturbance and tracking various set points.

Dynamic modeling was used in this work to build and evaluate a MPC *design strategy* which was first used in simulation using a detail process model, and then repeated experimentally. Each component of this design strategy was chosen to produce a high-quality MPC which can be implemented as an embedded controller for a commercial product. Sub-space identification is a excellent tool to generate reduced order models for use in MPC, and model identification using PRBS-type signals was successful both in simulation and experimentally. The main disadvantage to the design strategy is the linear approximation at an operating point for model identification, but the alternative, nonlinear MPC, would prohibit its use in a commercial device. More detailed modeling, either linear or nonlinear, could improve the MPC performance in theory, but as the model complexity increases, so does the computational cost. The piece-wise linear modeling approach took advantage of improved model prediction ability while still maintaining low computational cost. This approach allowed a complex system such as the RPSA device to be controlled by a relatively simple feedback controller.

This work also gives a systems-level perspective on RPSA design and operation which is not well represented in the current literature. Integrating a feedback controller into the RPSA system creates a closed-loop system which could behave differently than in open-loop, and could improve the overall RPSA device performance, but long-term studies are required to confirm this. Steady state optimization of the RPSA remains an attractive goal, and the work presented here presents a new approach in utilizing embedded hardware and process measurements to optimize a RPSA process.

Model-free optimization approaches are advantageous because they do not rely on potentially flawed process models, but fully integrating both nonlinear process models and embedded hardware should be the eventual goal because it would be a valuable contribution to the PSA industry.

7.2 Future Directions

Different modeling approaches could improve the MPC presented here. A fully nonlinear MPC using the detailed plant model could be developed which would provide superior feedback control for the RPSA system, but the nonlinear MPC would not be implementable in a commercial MOC device. If the control strategy is adapted for multi-bed PSA systems, the nonlinear MPC could be used in larger, industrial scale systems where more powerful computing resources are available. If a linear MPC formulation is desired, alternative methods for model reduction could be explored which give a more detailed linear model, perhaps integrating disturbance effects into control decisions. In any modeling decision, there is a trade-off between complexity and computational cost which must be considered before designing a MPC. Stability of the plant model, and then the entire closed-loop system, remains an open question, and a rigorous proof will require significant future work.

Steady state optimization using a combination of embedded hardware and process modeling should be a priority for future work. In both RPSA device design and generating a MPC, finding operating baseline points requires many experiments where the design variables are manually adjusted. Optimization can greatly reduce the number of experimental runs and aid designers in finding superior optimal operating points. A data-reconciled process model will best predict the RPSA response to changes in design variables, and help to reach optimums faster.

The single-bed design is currently undergoing development for future commercial manufacture of a new MOC device, and using the MPC can differentiate this new device in an already crowded market. Adding composition measurements and feedback control will provide more transparency to the end user of the device, and provide more concrete indications of when the device is performing properly, or when maintenance is required. By continuously monitoring the product composition and making adjustments to the cycle step durations, the MPC could extend the lifetime of the adsorber column as water contaminates the zeolite. The extent to which the MPC improves lifetime has yet to be quantified. The eventual success or failure of the MPC for a commercial product depends on the economics of including composition sensors and hardware to solve the MPC optimization problem. In this work, the MPC was shown to improve the closed-loop performance of the RPSA device, and was able to achieve all the desired objectives.

List of Publications

The presented dissertation was based on the following publications:

Peer-reviewed journal articles

M. D. Urich, R. R. Vemula and M. V. Kothare. (2018), "Implementation of an Embedded Model Predictive Controller for a Medical Oxygen Concentrator Device", *In preparation for submission, IFAC Journal of Systems and Control*

M. D. Urich, R. R. Vemula and M. V. Kothare. (2018), "Piece-Wise Linear Model Predictive Control of a Rapid Pressure Swing Adsorption System", *Under Review, IEEE Transactions on Control System Technology*

M. D. Urich, R. R. Vemula and M. V. Kothare. (2017), "Multivariable Model Predictive Control of a Rapid Pressure Swing Adsorption System", *AIChE Journal*, 64(4), 1234-1245. doi:10.102/aic.16011.

Peer-reviewed conference proceedings

M. D. Urich, R. R. Vemula and M. V. Kothare (August 2018), "Embedded Piece-Wise Linear Model Predictive Control of a Rapid Pressure Swing Adsorption

System", *Submitted to the 6th IFAC Conference on Nonlinear Model Predictive Control*, Madison, WI, USA

M. D. Urich, R. R. Vemula and M. V. Kothare (May 2017), "Multi-Model Predictive Control of a Rapid Pressure Swing Adsorption System", *In Proceedings of the 2017 American Control Conference*, pages: 4392-4397, Seattle, WA, USA

Patents

M. D. Urich, R. R. Vemula and M. V. Kothare (February 2018), "Embedded Multivariable Model Predictive Controller for a Medical Oxygen Concentrator", *Provisional patent application in preparation*

Refereed abstracts & talks

M. D. Urich, R. R. Vemula and M. V. Kothare (October 2018), "Embedded Model Predictive Control of a Medical Oxygen Concentrator Device", *Submitted to the 2018 AIChE Annual Meeting*, Pittsburgh, PA, USA

M. D. Urich, R. R. Vemula and M. V. Kothare (October 2018), "Using Hardware for On-Line Steady State Optimization of a Single-Bed Rapid Pressure Swing Adsorption System", *Submitted to the 2018 AIChE Annual Meeting*, Pittsburgh, PA, USA

M. D. Urich, R. R. Vemula and M. V. Kothare (September 2017), "Multivariable Model Predictive Control of a Rapid Pressure Swing Adsorption Medical Oxygen Concentrator", *In the 3rd Annual Dept. Chemical and Biomolecular Engineering Graduate Student Research Symposium*, Bethlehem, PA, USA

M. D. Urich, R. R. Vemula and M. V. Kothare (November 2016), "Multivariable Model Predictive Control of a Rapid Pressure Swing Adsorption Medical Oxygen Concentrator", *In the 2016 AIChE Annual Meeting*, San Francisco, CA, USA

Bibliography

[1] Oxygen concentrator market opportunities, market forcasts, and market strategies: 2011-2017. Technical report, WinterGreen Market Research Inc., March 2011.

[2] National Heart, Lung, and Blood Institute; National Institutes of Health; U.S. Department of Health and Human Services. COPD.

[3] R. R. Vemula, M. V. Kothare, and S. Sircar. Novel design and performance of a medical oxygen concentrator using a rapid pressure swing adsorption concept. *AIChE Journal*, 60(9):3330–3335, SEP 2014.

[4] R. L. Chatburn and T. J. Williams. Performance comparison of 4 portable oxygen concentrators. *Respiratory care*, 55(4):433–442, April 01 2010.

[5] W. Sun, Y. Shen, D. Zhang, H. Yang, and H. Ma. A systematic simulation and proposed optimization of the pressure swing adsorption process for N-2/CH4 separation under external disturbances. *Industrial & Engineering Chemistry Research*, 54(30):7489–7501, AUG 5 2015.

[6] Y. Tavan, S. H. Hosseini, and M. Olazar. A note on an integrated process of methane steam reforming in junction with pressure-swing adsorption to produce

pure hydrogen: Mathematical modeling. *Industrial & Engineering Chemistry Research*, 54(51):12937–12947, DEC 30 2015.

[7] Y. J. Kim, Y. S. Nam, and Y. T. Kang. Study on a numerical model and psa (pressure swing adsorption) process experiment for ch4/co2 separation from biogas. *Energy*, 91:732–741, NOV 2015.

[8] O. T. Qazvini and S. Fatemi. Modeling and simulation pressure-temperature swing adsorption process to remove mercaptan from humid natural gas; a commercial case study. *Separation and Purification Technology*, 139:88–103, JAN 2 2015.

[9] T. S. Bhatt, A. Sliepcevich, G. Storti, and R. Rota. Experimental and modeling analysis of dual-reflux pressure swing adsorption process. *Industrial & Engineering Chemistry Research*, 53(34):13448–13458, AUG 27 2014.

[10] J. Chen, S.-J. Hsu, and T.-Y. Wei. Optimization design for removal of radioactive kr from xe using pressure swing adsorption. *Chemical Engineering Research & Design*, 91(4):649–659, APR 2013.

[11] Y. Liu and F. Sun. Parameter estimation of a pressure swing adsorption model for air separation using multi-objective optimisation and support vector regression model. *Expert Systems with Applications*, 40(11):4496–4502, SEP 1 2013.

[12] M. M. F. Hasan, R. C. Baliban, J. A. Elia, and C. A. Floudas. Modeling, simulation, and optimization of postcombustion co2 capture for variable feed concentration and flow rate. 2. pressure swing adsorption and vacuum swing adsorption

processes. *Industrial & Engineering Chemistry Research*, 51(48):15665–15682, DEC 5 2012.

[13] A. W. Dowling, S. R. R. Vetukuri, and L. T. Biegler. Large-scale optimization strategies for pressure swing adsorption cycle synthesis. *AIChE Journal*, 58(12):3777–3791, DEC 2012.

[14] H. Khajuria and E. N. Pistikopoulos. Dynamic modeling and explicit/multi-parametric mpc control of pressure swing adsorption systems. *Journal of Process Control*, 21(1):151–163, JAN 2011.

[15] H. Khajuria and E. N. Pistikopoulos. Optimization and control of pressure swing adsorption processes under uncertainty. *AIChE Journal*, 59(1):120–131, JAN 2013.

[16] S. Ayoub, D. Tondeur, and M. A. Latifi. New formulation of optimization-based simulation of a pressure swing adsorption process: Hybrid dynamic optimization. *AIChE Journal*, 57(5):1367–1373, MAY 2011.

[17] A. Agarwal, L. T. Biegler, and S. E. Zitney. Simulation and optimization of pressure swing adsorption systems using reduced-order modeling. *Industrial & Engineering Chemistry Research*, 48(5):2327–2343, MAR 4 2009.

[18] M. A. Latifi, D. Salhi, and D. Tondeur. Optimisation-based simulation of a pressure swing adsorption process. *Adsorption-Journal of the International Adsorption Society*, 14(4-5):567–573, OCT 2008.

[19] J. C. Santos, A. F. Portugal, F. D. Magalhaes, and A. Mendes. Simulation and optimization of small oxygen pressure swing adsorption units. *Industrial & Engineering Chemistry Research*, 43(26):8328–8338, DEC 22 2004.

[20] D. Ko, R. Siriwardane, and L. T. Biegler. Optimization of pressure swing adsorption and fractionated vacuum pressure swing adsorption processes for co2 capture. *Industrial & Engineering Chemistry Research*, 44(21):8084–8094, OCT 12 2005.

[21] S. P. Knaebel, D. H. Ko, and L. T. Biegler. Simulation and optimization of a pressure swing adsorption system: Recovering hydrogen from methane. *Adsorption-Journal of the International Adsorption Society*, 11:615–620, 2005.

[22] L. Jiang, L. T. Biegler, and V. G. Fox. Simulation and optimization of pressure-swing adsorption systems for air separation. *AIChE Journal*, 49(5):1140–1157, MAY 2003.

[23] D. Ko, R. Siriwardane, and L. T. Biegler. Optimization of a pressure-swing adsorption process using zeolite 13x for co2 sequestration. *Industrial & Engineering Chemistry Research*, 42(2):339–348, JAN 22 2003.

[24] R. Rajasree and A. S. Moharir. Simulation based synthesis, design and optimization of pressure swing adsorption (psa) processes. *Computers & Chemical Engineering*, 24(11):2493–2505, NOV 1 2000.

[25] M. D. Urich, R. R. Vemula, and M. V. Kothare. Multivariable model predictive control of a novel rapid pressure swing adsorption system. *AIChE Journal*, 64(4):1234–1245, Apr 2018.

[26] J. B. Rawlings and D. Q. Mayne. *Model predictive control: Theory and design.* Nob Hill Pub., 2009.

[27] R. R. Vemula, S. Sircar, and M. V. Kothare. Oxygen concentrator system and method, May 16 2017.

[28] S. W. Chai, M. V. Kothare, and S. Sircar. Rapid pressure swing adsorption for reduction of bed size factor of a medical oxygen concentrator. *Industrial & Engineering Chemistry Research*, 50(14):8703–8710, JUL 20 2011.

[29] C. W. Skarstrom. Method and apparatus for fractionating gaseous mixtures by adsorption, July 1960. US Patent 2,944,627.

[30] R. R. Vemula, M. V. Kothare, and S. Sircar. Numerical simulation of rapid pressurization and depressurization of a zeolite column using nitrogen. *Adsorption-Journal of the International Adsorption Society*, 20(1):53–60, JAN 2014.

[31] Lennart Ljung. *System Identication: Theory for the user.* Pretence-Hall, Inc., 1987.

[32] P. Van Overschee and B. De Moor. N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1):75–93, 1994.

[33] J. A. Primbs and V. Nevistic. A new approach to stability analysis for constrained finite receding horizon control without end constraints. *IEEE Transactions on Automatic Control*, 45(8):1507–1512, 2000.

[34] R. Murray-Smith and T. Johansen. *Multiple Model Approaches To Nonlinear Modelling And Control.* Series in Systems and Control. Taylor & Francis, 1997.

[35] Z. Yang, Y. Li, and J. E. Seem. Multi-model predictive control for wind turbine operation under meandering wake of upstream turbines. *Control Engineering Practice*, 45:37–45, DEC 2015.

[36] X. Tao, N. Li, and S. Li. Multiple model predictive control for large envelope flight of hypersonic vehicle systems. *Information Sciences*, 328:115–126, JAN 20 2016.

[37] A. Vachon, A. Desbiens, E. Gagnon, and C. Berard. Launch ascent guidance by discrete multi-model predictive control. *Acta Astronautica*, 95:101–110, FEB-MAR 2014.

[38] K. Hu and J. Yuan. Multi-model predictive control method for nuclear steam generator water level. *Energy Conversion and Management*, 49(5):1167–1174, MAY 2008.

[39] R. R. Rao, B. Aufderheide, and B. W. Bequette. Experimental studies on multiple-model predictive control for automated regulation of hemodynamic variables. *IEEE Transactions on Biomedical Engineering*, 50(3):277–288, MAR 2003.

[40] X. Tao, D. Li, Y. Wang, N. Li, and S. Li. Gap-metric-based multiple-model predictive control with a polyhedral stability region. *Industrial & Engineering Chemistry Research*, 54(45):11319–11329, NOV 18 2015.

[41] M. Soliman, O. P. Malik, and D. T. Westwick. Multiple model predictive control for wind turbines with doubly fed induction generators. *IEEE Access Transactions on Sustainable Energy*, 2(3):215–225, JUL 2011.

[42] Qihong Chen, Lijun Gao, Roger A. Dougal, and Shuhai Quan. Multiple model predictive control for a hybrid proton exchange membrane fuel cell system. *Journal of Power Sources*, 191(2):473–482, JUN 15 2009.

[43] P.J. van Overloop, S. Weijs, and S. Dijkstra. Multiple model predictive control on a drainage canal system. *Control Engineering Practice*, 16(5):531–540, MAY 2008.

[44] L. Ozkan, M. V. Kothare, and C. Georgakis. Control of a solution copolymerization reactor using multi-model predictive control. *Chemical Engineering Science*, 58(7):1207–1221, APR 2003.

[45] G. Feng. Stability analysis of piecewise discrete-time linear systems. *IEEE Transactions on Automatic Control*, 47(7):1108–1112, Jul 2002.

[46] H. Lin and P. J. Antsaklis. Stability and stabilizability of switched linear systems: A survey of recent results. *IEEE Transactions on Automatic Control*, 54(2):308–322, Feb 2009.

[47] L. Ozkan and M. V. Kothare. Stability analysis of a multi-model predictive control algorithm with application to control of chemical reactors. *Journal of Process Control*, 16(2):81 – 90, 2006.

[48] C. Siew-Wah, S. Sircar, and M. V. Kothare. Miniature oxygen concentrators and methods, 2012.

[49] W. Marquardt. Nonlinear model reduction for optimization based control of transient chemical processes. In *AIChE Symposium Series*, pages 12–42. New York; American Institute of Chemical Engineers; 1998, 2002.

[50] M. D. Anderson, J. Dahl, and L. Vandenberghe. CVXOPT: A Python package for convex optimization., 2013. Version 1.6.

Biographcial Information

Matthew Urich received a Ph. D. in Chemical Engineering from Lehigh University with a focus on predictive control of a pressure swing adsorption system. In 2014, he received a Bachelor of Science in Engineering in Chemical Engineering and a minor in Chemistry from the University of Pittsburgh. He was named a P. C. Rossin Doctoral Fellow in September, 2016 and the recipient of the Chevron Scholars Award in Chemical Engineering in August, 2016. His research interests include the control and optimization of large-scale, multivariable processes particularly in the chemical industry. After receiving his doctorate from Lehigh University, Matthew joined Air Products, Inc. in Trexlertown, PA.