

**Advanced Strategies for Optimal Design and Operation of
Pressure Swing Adsorption Processes**

Submitted in partial fulfillment of the requirements for
the degree of
Doctor of Philosophy
in
Chemical Engineering

Anshul Agarwal

B.Tech., Chemical Engineering, Indian Institute of Technology Delhi

M.Ch.E., Chemical Engineering, Carnegie Mellon University

Carnegie Mellon University
Pittsburgh, PA

May, 2010

Acknowledgments

First and foremost, I would like to thank Professor Lorenz T. Biegler for his guidance, encouragement and support which made this research work meaningful and fruitful. His patience, tenacity, enthusiasm, and depth of knowledge has left a deep impression on me. His guidance has made me understand and appreciate the importance and potentials of mathematical programming in decision making.

My thanks go to the National Energy Technology Laboratory (NETL) for providing financial auspices over the duration of this project. I would like to thank Dr. Stephen E. Zitney and Dr. David C. Miller from NETL for their comments on my research and for ensuring that my work proceeded towards the vision of the project. Besides the committee chair Prof. Biegler, other committee members Prof. Ignacio E. Grossmann, Prof. B. Erik Ydstie, and Prof. Bruce H. Krogh deserve my gratitude for their helpful advice and suggestions.

I would also like to thank Yi-Dong Lang and Sree for constructive discussions on reduced-order modeling, and trust-region methods.

Finally, I would like to express my deepest gratitude to my parents for their incessant support, encouragement, belief, and love. Without that I wouldn't have come this far.

Abstract

With expanding areas of applications, increasing needs for efficient cycles, and growing demands for efficient modeling, it has become essential to develop new systematic strategies for optimal design and operation of PSA systems. Although industrial usage of PSA is widespread, we observe a drought of any systematic methodology to design PSA cycles in PSA literature due to inherent complexity of cyclic PSA processes. We present a generic PSA superstructure to synthesize optimal PSA configurations. The superstructure is rich enough to predict a number of different PSA operating steps, and their optimal sequence by solving an optimal control problem.

Because of low operating costs and high performance, PSA is considered as a promising option for both post-combustion and pre-combustion CO₂ capture. Since commercial PSA cycles consider CO₂ as a waste stream, cycle development specifically targeted towards high-purity CO₂ separation is essential. We utilize superstructure approach for this purpose and succeed in synthesizing optimal cycles which can separate CO₂ at a purity as high as 95%, or with a low power consumption of 465 kWh/tonne CO₂ captured, for post-combustion capture. When applied for pre-combustion capture, superstructure approach yields cycles with an extremely low power number of 46.8 kWh/tonne CO₂ captured.

Large number of spatial nodes required to capture steep adsorption fronts lead to a large set of DAEs, and thus to a challenging PSA optimization problem. We generate reduced-order models (ROMs) which are not only orders of magnitude smaller, but also reasonably accurate. Consequently, replacing DAEs with these ROMs yields a cheap optimization problem. How-

ever, a trust-region envelope is essential for optimization as a ROM is accurate only around the point where it is constructed. Optimization with a trust-region is successfully demonstrated for a H₂-CH₄ PSA case study. Promising preliminary results encourage us to formally devise a systematic adaptive trust-region strategy. We first develop an exact penalty trust-region algorithm and devise correction schemes to ensure convergence to actual local optimum. When demonstrated for the Skarstrom cycle for CO₂ capture, penalty algorithm converges in 92 iterations and 1.88 CPU hrs. To circumvent the difficulty of determining a penalty parameter, we also devise a hybrid filter-based trust-region framework. When applied to the PSA case study, filter algorithm converges within 51 iterations consuming 1.36 CPU hrs. Thus, initial results are quite promising and reveal the potential of using ROMs for optimization.

Contents

Acknowledgments	ii
Abstract	iii
Contents	v
List of Tables	x
List of Figures	xii
1 Introduction	1
1.1 PSA Overview	2
1.2 Research Challenges with PSA	4
1.2.1 Process Development	4
1.2.2 Computationally-efficient Simulation/Optimization	5
1.2.3 Modeling Equilibrium and Kinetic Behavior	6
1.2.4 Adsorbent Materials	7
1.3 Problem Statement and Scope of Work	7
1.4 Thesis Outline	8
2 Pressure Swing Adsorption	11
2.1 Adsorption Phenomena and Processes	12
2.2 Adsorbent Properties	13

2.3	PSA Modeling	14
2.3.1	Mass Transfer	14
2.3.2	Adsorption Isotherm	17
2.3.3	Material Balance	19
2.3.4	Energy Balance	20
2.3.5	Momentum Balance	21
2.4	PSA Operation	22
2.4.1	Operation Scheme	22
2.4.2	Cyclic Steady State	24
2.4.3	Boundary-matching for Multi-bed Operation	25
2.5	Simulation Methodologies	27
2.5.1	Spatial Discretization	29
2.5.2	Temporal Discretization	32
2.6	Concluding Remarks	33
3	PSA Superstructure	35
3.1	Motivation	36
3.2	Methodology	37
3.2.1	Superstructure	37
3.2.2	Cycle Realization	38
3.2.3	Optimal Control Problem	40
3.3	Model Equations	41
3.4	Solution Strategy	44
4	Superstructure Case Study: Post-combustion CO₂ Capture	47
4.1	Introduction	48
4.2	Literature Review	50
4.3	Case Setup	53

4.4	Case Studies and Computational Results	54
4.4.1	Case I: Optimization with a conventional configuration	54
4.4.2	Case II: Cycle synthesis to maximize CO ₂ recovery	56
4.4.3	Case III: Cycle synthesis to minimize power consumption	63
4.5	Comparison of Cycles from Case II and III	68
4.6	Conclusions and Future Work	70
5	Superstructure Case Study: Pre-combustion CO₂ Capture	72
5.1	Introduction and Previous Work	73
5.2	Case Setup	76
5.3	Case Studies and Computational Results	78
5.3.1	Case I: Cycle synthesis to maximize CO ₂ recovery	78
5.3.2	Case II: Cycle synthesis to minimize power consumption	86
5.4	Comparison of Cycles from Case I and II	93
5.5	Conclusions and Future Work	94
6	Reduced-order Modeling for Optimization	96
6.1	Motivation	97
6.2	Proper Orthogonal Decomposition (POD)	100
6.2.1	Concept	100
6.2.2	Optimality Property	102
6.2.3	POD and Singular Value Decomposition	103
6.2.4	Subspace Selection	104
6.3	Reduced-order Modeling	105
6.3.1	Methodology	105
6.3.2	Handling Boundary Conditions	107
6.3.3	Example - Burgers Equation	108
6.4	ROM-based Optimization	110

6.5	Case Study - Hydrogen PSA	112
6.5.1	PSA Process and Model Equations	112
6.5.2	Reduced-order Model	115
6.5.3	Comparison of Rigorous Model and ROM at CSS	119
6.5.4	ROM-based Optimization within a Trust-region	124
6.5.4.1	Case I: Optimization without velocities	124
6.5.4.2	Case II: Optimization with velocities	129
6.6	Conclusions	133
7	Trust-region Framework for ROM-based Optimization	134
7.1	Introduction	135
7.2	Optimization Problem	137
7.2.1	Trust-region Subproblem	137
7.2.2	Correction (scaling) for Objective and Constraints	138
7.3	Penalty-based Trust-region Algorithm	141
7.3.1	Penalty Formulation	141
7.3.2	Trust-region Algorithm	143
7.3.3	Convergence and Optimality	145
7.4	PSA Case Study - Post Combustion CO ₂ Capture	146
7.4.1	PSA Process and Model Equations	146
7.4.2	Reduced-order Model	149
7.4.3	Optimization using Algorithm I	151
7.4.3.1	Comparison of Rigorous Model and ROM at the Starting Guess	152
7.4.3.2	Optimization with Zero-order Correction	156
7.4.3.3	Optimization with First-order Correction	159
7.5	Hybrid Filter Trust-region Algorithm	162
7.5.1	Motivation	162
7.5.2	Filter	164

7.5.3	Trust-region Step Computation	166
7.5.4	Switching Condition	167
7.5.5	Algorithm Section I: Filter with ZOC	168
7.5.6	Algorithm Section II: Filter with FOC	170
7.5.7	Feasibility Restoration Phase	171
7.5.8	Trust-region Algorithm	172
7.5.9	Convergence and Optimality	176
7.6	PSA Case Study Revisited	178
7.7	Conclusions	181
8	Conclusions	184
8.1	Thesis Summary and Contributions	185
8.2	Directions for Future Work	189
	Bibliography	192
	A Nomenclature	206
	B Optimization Iterations	209
B.1	Exact Penalty Algorithm with ZOC	209
B.2	Exact Penalty Algorithm with FOC	210
B.3	Hybrid Filter Trust-region Algorithm: Tangential Subproblem	213
B.4	Hybrid Filter Trust-region Algorithm: Normal Subproblem	215

List of Tables

3.1	PSA Model Equations	42
4.1	PSA cycles suggested in the literature for post-combustion CO ₂ separation . . .	51
4.2	Zeolite 13X properties and model parameters [111]	53
4.3	Optimization results for Case I	56
4.4	Optimization results for case II	61
4.5	Accuracy validation for AMPL results of case II	62
4.6	Optimization results for case III	66
4.7	Accuracy validation for AMPL results of case III	67
5.1	Activated carbon properties and model parameters [100]	75
5.2	Optimization results for case I	84
5.3	Optimization results for case II	91
6.1	Physical properties and basic simulation parameters	113
6.2	Model equations of hydrogen PSA	114
6.3	Isotherm parameters for H ₂ and CH ₄ on activated carbon [108]	116
6.4	Molar flux variables for each operating step	116
6.5	Boundary conditions for each operating step	116
6.6	ROM for the hydrogen PSA process	118
6.7	Comparison of rigorous model and ROM based on the performance variables .	123
6.8	Decision variables and the root-point at which ROM is built	125

6.9	Optimization results for Case I	125
6.10	Perturbation results	128
6.11	Comparison of hydrogen performance for relaxed optimization	128
6.12	Optimization results for Case II	130
7.1	Model equations for isothermal PSA	147
7.2	Zeolite 13X properties and model parameters [111]	148
7.3	Boundary conditions for each operating step	148
7.4	ROM for the isothermal PSA process	150
7.5	Initial guess for optimization problem (7.31)	152
7.6	Comparison of rigorous model and ROM based on the performance variables .	156
7.7	Optimization results with ZOC	157
7.8	Perturbation results with ZOC	158
7.9	Optimization results with FOC	160
7.10	Perturbation results with FOC	162
7.11	Optimization results with hybrid filter trust-region algorithm	179
7.12	Perturbation results for Algorithm II	181
B.1	Iteration sequence for Algorithm I with ZOC for Problem (7.31)	209
B.2	Iteration sequence for Algorithm I with FOC for Problem (7.31)	210
B.3	Iteration sequence for tangential subproblems of Algorithm II for Problem (7.31)	213
B.4	Iteration sequence for normal subproblems of Algorithm II for Problem (7.31) .	215

List of Figures

2.1	A composite adsorbent pellet with different mass transfer resistances [156] . . .	14
2.2	Adsorption isotherms and change in equilibrium solid loading with pressure and temperature	17
2.3	A 2-bed 4-step Skarstrom cycle [176]	22
2.4	Time chart for a 5-bed 11-step H ₂ PSA process [100]	24
2.5	Boundary matchings required for the two-bed four-step PSA process	25
2.6	Approaches to match boundary values while simulating multi-bed PSA operations	26
2.7	Finite volume discretization scheme	30
3.1	A 2-bed PSA Superstructure	37
3.2	Realization of different operating steps by varying control variables	39
3.3	Realization of 4-step Skarstrom cycle from the superstructure	39
4.1	Optimal control profiles for case II	58
4.2	Optimal VSA configuration for case II	58
4.3	Gas-phase CO ₂ concentration profiles for case II	60
4.4	Purity-recovery trade-off curve for case II	62
4.5	Optimal control profiles for case III	64
4.6	Optimal VSA configuration for case III	64
4.7	Gas-phase CO ₂ concentration profiles for case III	65
4.8	Power-recovery trade-off curve, at 90% CO ₂ purity	68

5.1	Optimal control profiles for case I	79
5.2	Optimal VSA configuration for case I	80
5.3	Gas-phase CO ₂ concentration profiles for case I	82
5.4	Purity-recovery trade-off curve for case I	85
5.5	Optimal control profiles for case II	87
5.6	Optimal VSA configuration for case II	88
5.7	Gas-phase CO ₂ concentration profiles for case II	89
5.8	Power-recovery trade-off curve at 90% CO ₂ purity for case II	92
6.1	Comparison of original profile and ROM profiles for Burgers equation for varying subspace dimension	109
6.2	First six POD basis functions of methane mole fraction for adsorption	119
6.3	Singular values of gas-phase mole fraction of methane and temperature	120
6.4	Comparison of methane mole fraction for all the steps after CSS	121
6.5	Comparison of temperature profiles for all the steps after CSS	122
6.6	Comparison of methane mole fraction profile for adsorption step obtained after integrating in MATLAB and solving simultaneously in AMPL	123
6.7	Comparison of mole fraction of methane after optimizing Case I	127
6.8	Gas-phase methane mole fraction profiles for ROM for relaxed bounds in Case I	129
6.9	Comparison of mole fraction of methane after optimizing Case II	131
6.10	Methane mole fraction profiles for ROM for relaxed bounds in Case II	132
7.1	First six POD basis functions of CO ₂ mole fraction for adsorption	153
7.2	Singular values of mole fraction of CO ₂ and superficial gas velocity	154
7.3	Comparison of CO ₂ mole fraction for all the steps at the starting guess	155
7.4	Comparison of CO ₂ mole fraction for penalty TR algorithm with FOC	161
7.5	A filter with four (θ, f) -pairs	165
7.6	Comparison of CO ₂ mole fraction for hybrid filter TR algorithm	182

Without mathematics we cannot penetrate deeply into philosophy

Without philosophy we cannot penetrate deeply into mathematics

Without both we cannot penetrate deeply into anything

— **Leibniz**

Philosophy is written in this very great book

which always lies open before our eyes (I mean the universe),

but one cannot understand it unless one first learns to understand

the language and recognize the characters in which it is written.

It is written in mathematical language and

the characters are triangles, circles and other geometrical figures;

without these means it is humanly impossible to understand a word of it;

without these there is only clueless scrabbling around in a dark labyrinth.

— **Galileo**

Chapter 1

Introduction

Synopsis

Over the last few decades, pressure swing adsorption (PSA) processes have emerged successfully as cost-effective alternatives to the traditional gas separation processes, and thus have gained widespread acceptance. Although commercial utilization is widespread, PSA processes still present stiff research challenges in terms of process development, accurate modeling of mass transfer and adsorption phenomena, and adsorbent design, especially for emerging new applications. In this chapter, we highlight such challenges in brief, describe our approach in order to address a few of these, and define the scope of our research. An outline of this thesis is presented in the final section of this chapter.

1.1 PSA Overview

Separation of gases accounts for a major fraction of the production cost in chemical, petrochemical, and related industries. There has been a growing demand for economical and energy efficient gas separation processes. The new generation of more selective adsorbents developed in recent years has enabled adsorption-based technologies to compete successfully with traditional gas separation techniques, such as cryogenic distillation and absorption. The last few decades have seen a considerable increase in the applications of adsorptive gas separation technologies, such as pressure swing adsorption (PSA). Pressure swing adsorption is a versatile technology for separation and purification of gas mixtures. While initial applications of PSA included gas drying and purification of dilute mixtures, current industrial applications include solvent vapor recovery, air fractionation, production of hydrogen from steam-methane reformer (SMR) and petroleum refinery offgases, separation of hydrocarbons such as carbon monoxide-hydrogen, carbon dioxide-methane, and n-paraffins separation, and alcohol dehydration. Advent of commercial PSA operations started with the early patents on this subject granted to Skarstrom [176] and Guerin de Montgareuil and Domine [60]. Since then, PSA has become the state-of-the-art separation technology for applications like air fractionation and hydrogen production. Many of these processes are described in published books and review articles on this subject [31, 92, 106, 154, 156, 166, 169, 181, 206]. Moreover, Sircar [167] has given an extensive list of publications on PSA which highlights growth in the research and development of PSA technology.

PSA processes involve selectively adsorbing certain components of a gas mixture on a microporous-mesoporous solid adsorbent at a relatively high pressure, via gas-solid contact in a packed column, in order to produce a gas stream enriched in less strongly adsorbed components of the feed gas. The adsorbed components are then desorbed from the solid by lowering their gas-phase partial pressures inside the column to enable adsorbent re-usability. Desorbed gases, as a result, are enriched in the more strongly adsorbed components of the feed gas. No external heat is generally used for desorption. The selectivity in a PSA process comes

from differences in either adsorption equilibrium or adsorption kinetics between the components to be separated. While a PSA process carries out adsorption at superambient pressure and desorption at near-ambient pressure level, a vacuum swing adsorption (VSA) process undergoes adsorption at near-ambient pressure, while desorption is achieved under vacuum. Practical PSA/VSA processes are substantially sophisticated with multiple adsorber columns executing a wide variety of non-steady-state operating steps in a non-trivial sequence. Besides adsorption and desorption, such a sequence also involves a multitude of complementary operating steps essential to control product gas purity and recovery, and optimize overall separation efficiency. Each bed undergoes this sequence of steps repeatedly, and thus the entire PSA system operates in a cyclic manner.

Some of the advantages of PSA systems and key reasons for recent growth of this technology are as below [31, 168]:

- PSA and VSA processes operate at ambient temperatures and do not require any solvent for product recovery or adsorbent regeneration. As a result, their capital expenditure is quite less compared to cryogenic technologies. Primary operating cost for these processes comes from the energy requirements for compression and vacuum generation. Hence, PSA processes are cost-effective compared to traditional technologies, and are especially desirable when lower production rates or lower product purities are required.
- Pressure manipulation serves as an extra degree of thermodynamic freedom, thus introducing significant flexibility in process design as compared with conventional technologies such as distillation, extraction or absorption.
- Numerous microporous-mesoporous adsorbents are available which are specifically tailored and engineered for a particular application, thus exhibiting high selectivity and adsorption capacity which leads to extremely high purity and recovery separation.
- Optimum marriage between a material and a process while synthesizing the separation scheme drives innovation and leads to highly efficient designs for PSA processes.

Theoretical modeling of PSA systems has also been extensively studied to gain a clear understanding of this rather complex process. A summary of the published dynamic models has been compiled by Ruthven [156] and Nikolić et al. [136]. In general, PSA bed model is a set of fairly complex partial differential and algebraic equations (PDAEs) which reflect the transient nature of the process and capture the underlying physics in detail. With such models, it is now possible to accurately predict the dynamic behavior of a PSA process, and to adequately account for all the factors that affect the performance of any given PSA system.

1.2 Research Challenges with PSA

Although commercial applications of PSA processes have grown rapidly, existing as well as emerging new applications present exorbitant research challenges in terms of developing a systematic formulation for synthesizing multibed PSA cycles, obtaining optimal operation strategy with detailed PSA bed models, and obtaining exhaustive experimental data for the kinetic and equilibrium behavior for novel adsorbents to accurately model multicomponent adsorption isotherm and mass transfer phenomena. Ruthven [155] and Sircar [169, 170] have summarized the current and future research trends.

1.2.1 Process Development

Although high-performance cycles have been developed for individual commercial applications, design and synthesis of a PSA system for given commercial specifications has largely remained an experimental effort. It is not clear why a particular cycle is chosen or one performs better than other configurations. More importantly, so far no systematic algorithm or method has been developed in the literature to design and evaluate a PSA cycle configuration due to the inherent complexity of the process which involves a number of sequential but interacting unsteady-state cycle steps. *A priori* design of a practical cycle that can guarantee commercial specifications without the use of supporting data from a bench- or pilot-scale process is considered less desirable because of the expense and computational time involved in solving the

rigorous mathematical models.

Very few studies in the literature have tried to address this issue. All of these studies suggest simplistic formulations to determine minimum number of beds required in a PSA process for given kinds and fixed sequences of operating steps. Chiang [46] proposed simple arithmetic-based heuristics, while Smith et al. [177] extended Chiang's work to propose a mixed-integer nonlinear programming (MINLP) based approach to obtain optimal number of beds required to execute a fixed sequence of operating steps. Smith et al. [178, 179] also suggested a 3-step scenario to design an industrial PSA system, but again with a known cycle of operating steps. Recently, Nikolić et al. [136, 137] proposed a state-task network (STN) based framework to determine optimal number of beds. However, STN developed wasn't exhaustive and missed many basic steps.

It is clear that novel PSA cycle sequences are anticipated for upcoming applications as well as for high-efficiency separation for current applications. Although process designers commonly resort to simplified and specific models for the PSA process of interest, and utilize simplistic descriptions in order to achieve satisfactory designs, accurate and reliable industrial design requires more challenging effort which is systematic and rigorous.

1.2.2 Computationally-efficient Simulation/Optimization

Another research challenge, as briefly mentioned in the previous section, relates to developing computationally-efficient strategies to simulate and optimize PSA systems governed by rigorous mathematical models. The behavior in each bed is described by partial differential and algebraic equations (PDAEs) in space and time, constructed from conservation of heat, mass, and momentum augmented by transport and equilibrium equations. Such hyperbolic equations involve high nonlinearities arising from non-isothermal effects and nonlinear adsorption isotherms, with solution profiles represented by steep adsorption fronts. As a result, optimization of PSA systems for either design or operation presents a significant computational challenge to current optimization techniques and nonlinear programming algorithms [31].

Sophisticated optimization strategies have been developed and applied to PSA systems with significant improvement in the performance of the process. For optimization of a bench-scale and a rapid PSA process, Nilchan et al. [138, 139] proposed a complete discretization approach for the PDAEs. Smith et al. [177, 178, 179] suggested a mixed-integer nonlinear programming based approach to minimize number of beds. Ko et al. [110, 111] used an SQP-based approach for optimization of PSA and fractionated vacuum PSA (FVPSA) processes. Ko et al. [109] also formulated a multiobjective optimization problem for rapid PSA and temperature swing processes. Rajasree et al. [145] developed a simulation based approach for synthesis, design and optimization of PSA processes. Kapoor et al. [104] developed a simple optimization scheme for PSA systems based on black-box models and an interior penalty approach, and demonstrated it for three different PSA case studies. Kvamsdal et al. [118, 119, 120] optimized a PSA process for trace separation, and analyzed the effect of mass transfer and cyclic steady state convergence. Jiang et al. [99] used an SQP-based approach to solve PSA optimization problems and computed direct sensitivities to obtain derivatives. However, even the most efficient of these approaches can still be time consuming for large systems, which gives us a strong motivation to develop cost-efficient and robust optimization strategies for PSA processes.

Moreover, there is a strong need to incorporate spatially & temporally distributed models within flowsheet simulators, such as ASPEN and HYSYS, as they currently deal with lumped-parameter models which suffer from accuracy limitations. Inclusion of dynamic PDAE-based PSA model with other steady-state flowsheet models for overall flowsheet optimization is challenging, non-intuitive, and requires development of integrated optimization strategies which are convergent and robust [214].

1.2.3 Modeling Equilibrium and Kinetic Behavior

It is critical to develop reliable, analytical models for accurate prediction of the core properties (multicomponent gas adsorption equilibria, kinetics, and heat of adsorption) using a limited

data source, since they govern the performance of an adsorptive process and are vital for process design. Basic understanding of multicomponent gas-solid interactions is often very limited, especially for the heterogeneous adsorbents used in practice [169, 170]. An accurate knowledge of these interactions under all conditions of pressure, temperature, and gas compositions, which can vary widely in a practical PSA process, is needed for a reliable solution of the process models. In particular, there is a strong need to generate and compile a multicomponent adsorption database which can validate adsorption models used for PSA process design, and can compute aforementioned core properties. Although such databases exist [188, 105], they are quite old, not exhaustive and do not account for newly developed adsorbents for existing and emerging applications.

1.2.4 Adsorbent Materials

Synthesizing novel high-performance adsorbents presents research challenges of its own. Adsorbents are desired to provide large specific surface area for a large adsorption capacity, and to be selective enough to retain one or more components from the gaseous feed mixture. Beginning with amorphous adsorbents like silica gel and activated carbon, the range of industrial adsorbents have grown to include synthetic molecular sieves and next generation zeolites with symmetric and crystalline pore structures [155, 156]. The challenge is to synthesize significantly heterogeneous adsorbents which can provide substantially high selectivity for high-purity production, and to develop methods for quantitative characterization of adsorbent heterogeneity.

1.3 Problem Statement and Scope of Work

In this work, we address the challenges related to systematic PSA process synthesis, and developing computationally cheap strategies for simulation and optimization of PSA bed models, as listed in sections 1.2.1 and 1.2.2, respectively. Development of new adsorbent materials, models that compute adsorbent properties, and multicomponent adsorption database is beyond the scope of this work.

To address the issue of developing a systematic framework to develop novel PSA cycles, we present an optimization-based framework to generate optimal PSA cycles from a 2-bed PSA superstructure. The interconnections between the two beds are governed by time-dependent control variables. Different PSA operating steps are realized by varying these control variables. We achieve an optimal sequence of operating steps by solving an optimal control problem with the PDAEs of the PSA system. To demonstrate this framework, we limit the scope of this work to binary feed mixtures. Extending the formulation to multicomponent feed streams and multibed cycles is straight-forward.

In order to address the challenge associated with efficient computation of PSA bed models and optimization problem, we develop a model reduction based framework that systematically generates cost-efficient low-order representations of such large-scale PSA models. In particular, we obtain these reduced-order models (ROMs) using proper orthogonal decomposition (POD). POD basis functions are used within a Galerkin's projection framework to derive a low-order DAE system that accurately describes the dominant dynamics of the PDAE system. Further, these ROMs are used as surrogate models to define much smaller and computationally efficient optimization problem. Finally, we develop a novel trust-region framework to formulate a convergent and robust optimization algorithm which utilizes these ROM-based smaller optimization problems. In this work, we illustrate this framework with manageable two-bed PSA systems. However, the algorithm developed is generic and can be extended to multibed PSA systems with multicomponent feed mixtures.

1.4 Thesis Outline

In this thesis, our main focus is to introduce and develop these two novel ideas mentioned above (the 2-bed PSA superstructure to systematically generate optimal PSA cycles, and the novel trust-region framework for optimization using reduced-order models), and present a proof of concept for them using practical and computationally manageable PSA problems. We organize our work in eight chapters, and present an outline of each chapter in this section.

We begin with the concepts of Pressure Swing Adsorption (PSA) in Chapter 2. PSA is an adsorption-based separation process in which separation is achieved when an adsorbent preferentially adsorbs one or more components from a feed mixture. We provide a brief background on the principles of adsorption and describe different kinds of cyclic adsorption processes. Then we discuss mathematical modeling for a fixed bed PSA process, which involves characterizing mass, energy, and momentum balances together with mass transfer phenomena and adsorption equilibrium. In general, PSA processes are governed by coupled hyperbolic PDAEs. Numerical methodologies to simulate such PDAEs are also discussed.

Chapter 3 introduces the novel two-bed PSA superstructure to determine optimal PSA configurations. The superstructure consists of two beds, one of which acts as an adsorbing bed and the other as a desorbing bed. The interconnections between the two beds are governed by time-dependent control variables, such as fractions of the light and the heavy product recycle. The superstructure predicts different PSA operating steps by varying these control variables. An optimal sequence of operating steps is achieved by solving an optimal control problem. We realize that it is a singular control problem as the controls appear linearly. Solution strategy to solve this PDAE-constrained singular control problem is then discussed.

In Chapter 4, we demonstrate the superstructure approach for case studies related to post-combustion CO₂ capture. PSA is a promising option to effectively capture CO₂ from flue gas streams. However, most commercial PSA cycles do not focus on enriching the strongly adsorbed CO₂ as a product. It is necessary to develop PSA processes specifically targeted to obtain pure strongly adsorbed component. We present a fairly comprehensive review of the previous studies on PSA cycles for CO₂ production. This review highlights the difficulties associated with choosing one PSA cycle over another, and motivates development of a structured approach for PSA cycle design. Hence, we use the superstructure to synthesize optimal PSA cycles which maximize CO₂ recovery and minimize overall power consumption. Results obtained are quite encouraging and promising.

Chapter 5 illustrates the superstructure approach for case studies related to pre-combustion

CO₂ capture. PSA processes offer significant advantages for pre-combustion CO₂ capture in terms of performance, energy requirements and operating costs since the shifted synthesis gas (syngas) is available for separation at a high pressure with a high CO₂ concentration. Since the industrial PSA cycles focus on recovering H₂ at a very high purity but consider CO₂ as a waste stream, novel PSA cycle designs are anticipated which recover both H₂ and CO₂ at a high purity. With the help of the superstructure approach, we successfully synthesize optimal PSA configurations which maximize CO₂ recovery and minimize overall power consumption. This chapter, hence, demonstrates the versatility of the superstructure approach.

After this, the thesis focuses on addressing the challenge associated with efficient computation of PSA bed models and optimization problem. Chapter 6 describes a reduced-order modeling technique that can circumvent such computational challenges by generating cost-efficient low-order models which can be used as surrogate models in optimization problems. In particular, with method of snapshots, singular-value decomposition, and Galerkin's framework, we describe how proper orthogonal decomposition (POD) can be successfully used to construct reduced-order models which can be orders of magnitude smaller than the original model without losing accuracy. Further, we discuss ROM-based optimization, and describe how ROMs can be utilized to optimize in a trust-region around the point where it is constructed. Optimization strategy is illustrated with a hydrogen PSA case study.

In Chapter 7, we formally develop a trust-region framework for ROM-based optimization. First we develop an exact penalty-based trust-region algorithm, and develop correction schemes for objective and constraints to ensure global convergence with POD-based approximate models. Then we illustrate this algorithm with correction schemes for a two-bed four-step PSA case study for post-combustion capture. Next, highlighting drawbacks of penalty approach and benefits of a filter, we develop a hybrid filter trust-region algorithm for constrained ROM-based optimization. Filter algorithm is then demonstrated for the PSA case study.

Finally, we summarize the contributions of this thesis, and discuss directions for future work in Chapter 8. Nomenclature is listed in Appendix A.

Chapter 2

Pressure Swing Adsorption

Synopsis

Pressure Swing Adsorption (PSA) is an adsorption-based separation process where an adsorbent preferentially adsorbs one component (or a family of related components) from a feed mixture, thus achieving separation. To understand the design and operation of PSA processes, knowledge of adsorption principles and dynamic behavior of an adsorption column is vital. We first provide a brief background on the adsorption phenomena and its fundamentals. Then we discuss adsorption modeling in a fixed bed which, besides comprising mass, momentum and energy conservation, also involves characterizing adsorption equilibrium behavior and mass transfer resistances affecting adsorption kinetics. Further, we describe the transient cyclic operation of a two-bed as well as a multi-bed multi-step PSA processes with examples. Numerical methodologies to simulate PSA processes and obtain solution for the bed model are also discussed, which involve finite volume spatial discretization scheme which is mass conserving, avoids unphysical oscillations, and thus, is essential to simulate PSA bed models.

2.1 Adsorption Phenomena and Processes

As discussed by Coulson et al. [55], Yang [206], and Vermeulen et al. [190], adsorption involves contacting a free fluid phase (gas or liquid) with a rigid particulate phase which has the property of selectively taking up and storing one or more solute species originally contained in the fluid. Besides adsorption, the conditions for desorption must also exist as it is usually necessary to reuse the adsorbent. The fluid-solid interaction leads to a reduction in potential energy of the fluid molecules in the vicinity of adsorbent surface. As a result, the fluid molecules concentrate such that their molecular density in this region is substantially greater than the free fluid phase. The strength of such a surface interaction depends on the nature of both the solid adsorbent and the fluid adsorbate. Consequently, different substances adsorb with different affinities. Such a *selectivity* provides the basis to achieve separation in adsorption separation processes, such as Pressure Swing Adsorption.

If the fluid-surface interactions involves weak forces, such as van der Waals, we observe *physical adsorption* or *physisorption*. In contrast, if the forces are strong and involve electron transfer, the phenomena is called *chemisorption*. As highlighted by Ruthven et al. [156], adsorption separation processes utilize physisorption as it leads to economically viable desorption and regeneration of the spent adsorbent by manipulating external operating conditions. Adsorbent regeneration is an important aspect of adsorption separation processes. A given adsorbent has a finite solute uptake capacity from the free fluid phase and must be cleaned for re-utilization. Thus, the adsorption phenomena should be reversible. Such reversibility is economically achieved in case of physisorption compared to chemisorption.

Based on the principle of adsorption and adsorbent regeneration, adsorption separation processes are designed to operate in a cyclic manner. Often two fixed-bed adsorbers are provided, such that one is used for adsorption while the other is being regenerated. For separating components from gaseous mixtures, following two kinds of adsorption separation processes are generally employed [181]:

- **Temperature Swing Adsorption (TSA):** In this process, bulk separation of a mixed

gas is achieved by repeating adsorption at a lower temperature and desorption at a higher temperature. The cyclic operation in this case typically takes a rather long time because of a relatively large time constant of heat transfer due to poor thermal conduction in the adsorbent packed bed.

- **Pressure Swing Adsorption (PSA):** In this process, bulk separation of a mixed gas is achieved by repeating adsorption at a higher pressure and desorption at a lower pressure. In this case, the step time for desorption is of the same order of magnitude as that of the adsorption (sometimes even smaller). Hence, this process enjoys shorter cycle time and more productivity compared to TSA, and thus, is preferred.

An alternative to manipulate pressure and temperature is to alter the composition of the fluid phase to control the direction of adsorption. This operation, called **Concentration Swing Adsorption (CSA)**, is utilized when the free fluid phase is a liquid. Such aforementioned processes are collectively viewed as **Periodic Adsorption Processes (PAPs)**.

2.2 Adsorbent Properties

As mentioned before, physisorption is caused mainly by weak forces between fluid molecules and adsorbent surface. Thus, adsorbents are characterized by surface properties such as surface area. Moreover, the role of an adsorbent is to provide surface for selective adsorption of certain components from the fluid phase. Hence, the desirable properties which an adsorbent should possess are as follows [138]:

- **Capacity:** An adsorbent is desired to provide a large specific surface area for a large adsorption capacity. A low capacity adsorbent leads to longer and expensive adsorbent beds. The creation of a large internal surface area in a limited volume is commercially achieved by casting adsorbents from microporous materials. In addition to *micropores*, some adsorbents have larger pores called *macropores* which result from aggregation of fine powders into pellets. A typical porous adsorbent particle is illustrated in Figure 2.1.

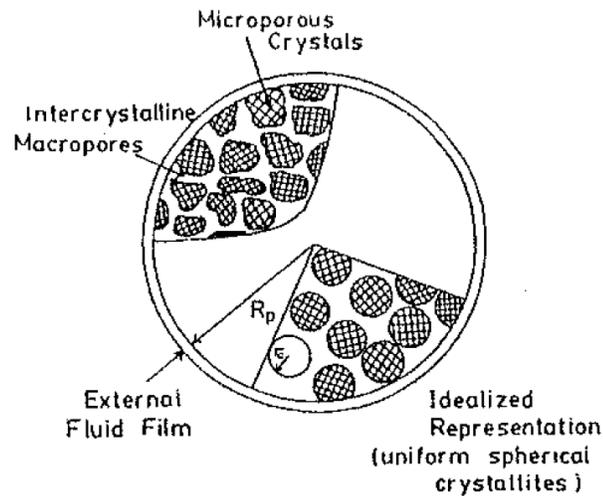


Figure 2.1: A composite adsorbent pellet with different mass transfer resistances [156]

Macropores function as diffusion paths of adsorbate molecules from outside the pellet to the micropores. As a result of such porous network, adsorption/desorption kinetics is controlled by intra-particle diffusion.

- **Selectivity:** An adsorbent must selectively retain one or more adsorbates from the fluid phase. This can be achieved either by *equilibrium selectivity*, in which species adsorb differently as a consequence of different equilibrium affinities, or by *kinetic selectivity*, in which relative adsorption depends on the difference of intra-particle diffusion rates among different adsorbate species.

2.3 PSA Modeling

2.3.1 Mass Transfer

At the microscopic level (see Figure 2.1), process of adsorption involves following steps in sequence (desorption step follows these steps in reverse):

1. The adsorbate diffuses from the bulk fluid phase to the external surface of the adsorbent pellet
2. From the external surface, adsorbate diffuses into and through the macropores.

3. If micropores exist, adsorbate diffuses further in the micropores before getting adsorbed onto the surface of the micropores, otherwise it adsorbs on the surface of macropores.

Consequently, the adsorbate encounters three different kinds of mass transfer resistances at each of the steps in this sequence, which are described as below:

- **External film resistance:** This exists in the external liquid film surrounding the adsorbent pellet. It can be characterized by using the system's Sherwood, Reynolds and Schmidt number. Typically this resistance is negligibly small in PSA systems [181, 156].
- **Macropore diffusive resistance:** This mass transfer resistance exists in the macropores of the adsorbent particle, and usually is the rate-controlling step. It depends on the relative magnitude of the pore diameter and the mean free path of the adsorbate under the operating conditions in the pores. When the pore diameter is much greater than the mean free path, *Bulk diffusion* ($D_{m,i}$) dominates the transport, and is estimated by Chapman-Enskog equation [181]. When the mean free path is much larger than the pore diameter, *Knudsen diffusion* dominates the transport, and is characterized by

$$D_K = 48.5d_p \sqrt{\frac{T}{M_w}} \quad (2.1)$$

Knudsen diffusion is usually dominant when the total pressure is quite low. In the intermediate case, effective diffusivity is obtained from Bosanquit's equation

$$D_{eff,i} = \frac{D_{m,i}D_K}{D_{m,i} + D_K} \quad i = 1, \dots, N_c \quad (2.2)$$

Finally, macropore diffusivity (called *Pore diffusion*) is characterized by following equation

$$D_{p,i} = \frac{\epsilon_p D_{eff,i}}{\tau} \quad i = 1, \dots, N_c \quad (2.3)$$

- **Micropore diffusive resistance:** Also known as *Surface diffusion*, this resistance exists in the micropores of the adsorbent pellet. For the adsorbents considered in this work,

either micropores don't exist or this resistance is negligible, i.e., adsorption occurs in the micropores instantaneously.

Depending on the mathematical complexity desired, mass transfer phenomena in an adsorption process can be described using either of the three models below.

- **Instantaneous equilibrium model**

This model is applicable when all the mass transfer resistances between the gas and the solid phases are negligible. As a result, equilibrium is attained instantaneously in the system with no mass transfer losses. It can be characterized as

$$\frac{\partial q_i}{\partial t} = \frac{\partial q_i^*}{\partial t} \quad i = 1, \dots, N_c \quad (2.4)$$

- **Pore diffusion model**

In the pore diffusion model, following detailed diffusion equation within the spherical adsorbent particle is solved to obtain the local rate of change of solid loading [207].

$$\frac{\partial q_i}{\partial t} = \frac{1}{r^2} \frac{\partial}{\partial r} \left(D_{p,i} r^2 \frac{\partial q_i}{\partial r} \right), \quad \forall r \in (0, d_p/2), \quad i = 1, \dots, N_c \quad (2.5)$$

Boundary condition is obtained by comparing internal flux with the external flux at the boundary of the pellet.

- **Linear driving force model (LDF)**

The LDF model [84] is obtained by assuming a parabolic solution profile for the pore diffusion model and then using an average solid loading for the entire adsorbent particle.

It is expressed as

$$\frac{\partial q_i}{\partial t} = k_i (q_i^* - q_i) \quad i = 1, \dots, N_c \quad (2.6)$$

while the mass transfer coefficient is given by

$$k_i = \frac{60 D_{p,i}}{d_p^2} \quad i = 1, \dots, N_c \quad (2.7)$$

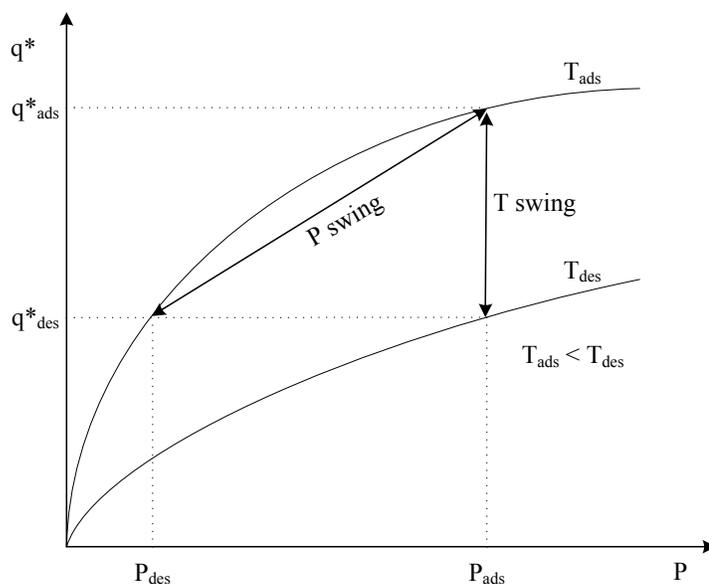


Figure 2.2: Adsorption isotherms and change in equilibrium solid loading with pressure and temperature

2.3.2 Adsorption Isotherm

An adsorbent in contact with the surrounding gaseous mixture for a sufficiently long time eventually attains equilibrium. In this state the amount of a component adsorbed on the surface is determined as shown in Figure 2.2 [154]. The relation between the equilibrium amount adsorbed and the total pressure of the fluid phase at a particular temperature is called an adsorption isotherm. In a general mathematical form, it is expressed as $[q^*]_T = f(P)$. Figure 2.2 also shows how adsorption/desorption is facilitated by changing total pressure or temperature of the system. We note that an adsorption process is always exothermic while desorption is always endothermic. Since the overall change in system's entropy is negative during adsorption, enthalpy change must be negative to ensure a net negative change in the Gibbs free energy (vice-versa for desorption). Consequently, adsorption is favored at a lower temperature, while desorption at a higher one. Similarly, at a high pressure, more adsorbate molecules interact with the molecules at the adsorbent surface leading to a higher adsorbent surface coverage and higher equilibrium solid loading. Hence, adsorption is favored at a high pressure while lowering the pressure facilitates desorption.

At sufficiently low adsorbate concentration (or partial pressure), adsorption isotherm attains a linear form, called *Henry's Law*:

$$q_i^* = k_i^H P_i \quad i = 1, \dots, N_c \quad (2.8)$$

Here k_i^H is Henry's constant and is inversely related to temperature ($k_i^H = k_0 e^{\Delta H_i^{ads}/RT}$). In general, nonlinear equilibrium relationship is described in numerous different mathematical forms, some of which are based on a simplified physical picture of adsorption/desorption phenomena while others are purely empirical and intended to correlate the experimental data. Commonly used mathematical models include single- and dual-site Langmuir model, Freundlich model, and Langmuir- Freundlich model.

- **Langmuir adsorption isotherm**

The Langmuir isotherm model is derived from mass action considerations and by balancing the occupied and unoccupied sites on the surface. It shows correct asymptotic behavior as it reduces to Henry's Law in the low-concentration region, and approaches saturation limit at high concentrations, which is a requirement for thermodynamic consistency in any physisorption-based system [156]. Single-site model

$$q_i^* = \frac{q_i^s b_i P_i}{1 + \sum_j b_j P_j} \quad \text{where} \quad q_i^s = k_i^1 + k_i^2 T \quad b_i = k_i^3 \exp(k_i^4/T) \quad i = 1, \dots, N_c \quad (2.9)$$

and dual-site model

$$q_i^* = \frac{q_{1i}^s b_{1i} P_i}{1 + \sum_j b_{1j} P_j} + \frac{q_{2i}^s b_{2i} P_i}{1 + \sum_j b_{2j} P_j} \quad i = 1, \dots, N_c \quad (2.10)$$

where $q_{mi}^s = k_{mi}^1 + k_{mi}^2 T \quad b_{mi} = k_{mi}^3 \exp(k_{mi}^4/T) \quad m = 1, 2$

are the commonly used formulations for Langmuir model.

- **Freundlich and Langmuir-Freundlich isotherms**

Depending on the separation system, Freundlich model

$$q_i^* = q_i^s b_i P_i^{1/n} \quad n > 1, \quad i = 1, \dots, N_c \quad (2.11)$$

or Langmuir-Freundlich model

$$q_i^* = \frac{q_i^s b_i P_i^{1/n}}{1 + \sum_j b_j P_j^{1/n}} \quad i = 1, \dots, N_c \quad (2.12)$$

are occasionally used as adsorption isotherms. Both these models are empirical in nature with no sound theoretical basis, and both of these do not reduce to Henry's Law in the low-concentration region. However, they can cogently represent the behavior of several systems over a wide range of conditions.

2.3.3 Material Balance

PSA processes are generally carried out with packed adsorption columns. The dynamic behavior of an adsorption column is governed by the coaction of adsorption kinetics, adsorption equilibrium, and fluid dynamics, and its understanding is vital for process modeling and analysis [106, 156, 169, 206].

Assuming an axially dispersed plug flow pattern in a fixed bed adsorption column, the transient component material balance for the bulk gas phase is given by

$$\epsilon_b \frac{\partial C_i}{\partial t} + (1 - \epsilon_b) \rho_s \frac{\partial q_i}{\partial t} + \frac{\partial v C_i}{\partial x} = D_L \frac{\partial^2 C_i}{\partial x^2} \quad i = 1, \dots, N_c \quad (2.13)$$

We often omit the axial dispersion term from this equation while numerically integrating it because the numerical dispersion resulting from spatial discretization of the flux term always exists for any discretization scheme. Thus, considering physical dispersion, together with the inevitable numerical dispersion, causes additional smearing of the solution profiles. Hence, instead of a parabolic formulation, the following hyperbolic form is used for component material

balance in the bulk gas phase

$$\epsilon_b \frac{\partial C_i}{\partial t} + (1 - \epsilon_b) \rho_s \frac{\partial q_i}{\partial t} + \frac{\partial v C_i}{\partial x} = 0 \quad i = 1, \dots, N_c \quad (2.14)$$

Here we assume no radial dependence for concentration and solid loading. Thus, in the equation above, C_i and q_i represent cross-sectional average values. Numerical dispersion in such hyperbolic systems is generally mitigated by considering a large number of spatial discretization nodes or by utilizing high-resolution spatial discretization techniques.

2.3.4 Energy Balance

As mentioned before, an adsorption process is accompanied by the evolution of heat, and temperature changes influence the adsorption equilibrium behavior. Thus, accounting for heat generation and transfer in adsorbent beds is essential for accurate modeling of PSA processes. The heat generated on the adsorbent surface is transported through conduction between adsorbent particles and through convection in the bulk gas phase. The extent of temperature variation in an adsorption chamber depends primarily on the heat of adsorption, bulk-phase transport properties, and heat transfer characteristics of the packed bed such as thermal conductivity and heat transfer coefficient [67].

In general, temperature difference is neglected and thermal equilibrium is assumed between the bulk gas phase and adsorbent particles. Moreover, heat transfer in the axial direction by thermal conduction is often negligible unless the operation is adiabatic at a very high flow rate. Based on these assumptions, the energy balance for the system is given by

$$\left(\epsilon_b \sum_i C_i (C_{pg}^i - R) + \rho_b C_{ps} \right) \frac{\partial T}{\partial t} - \rho_b \sum_i \Delta H_i^{ads} \frac{\partial q_i}{\partial t} + \frac{\partial v h}{\partial x} + U_A (T - T_w) = 0 \quad (2.15a)$$

$$C_{pg}^i = a_c^i + b_c^i T + c_c^i T^2 + d_c^i T^3 \quad i = 1, \dots, N_c \quad (2.15b)$$

$$h = \sum_i \left(C_i \int C_{pg}^i dT \right) \quad (2.15c)$$

Here we also consider temperature dependence of heat capacities and heat transfer through the wall of the column. As in the case of material balance, here T represents average temperature across cross-section. The effective heat transfer coefficient U_A comprises contributions from both fluid-to-wall as well as fluid-to-particle heat transfer coefficients. Fluid-to-particle heat transfer coefficient can be obtained from Carberry equation and it depends on system's Nusselt number, Prandtl number, and Reynolds number [181]. Fluid-to-wall heat transfer coefficient is usually obtained from empirical correlations. If only fluid-to-wall heat transfer is assumed, U_A reduces to

$$U_A = \frac{4h_w}{D} \quad (2.16)$$

where D is the column diameter and h_w fluid-to-wall heat transfer coefficient.

2.3.5 Momentum Balance

As the bulk fluid flows through the void spaces between adsorbent particles, it experiences a pressure drop due to viscous energy losses and drop in kinetic energy. Ergun equation is commonly used to describe such a pressure drop along the bed length

$$-\frac{\partial P}{\partial x} = \frac{150\mu(1-\epsilon_b)^2}{d_p^2\epsilon_b^3}v + \frac{1.75}{d_p} \left(\frac{1-\epsilon_b}{\epsilon_b^3} \right) \left(\sum_i M_w^i C_i \right) v|v| \quad (2.17)$$

The first term on the right-hand side represents losses due to viscous flow (laminar part), while the second term accounts for the drop in kinetic energy (turbulent part).

Often pressure drop across the bed is assumed negligible and is not considered in the analysis of the dynamic behavior of a PSA process [48, 47, 144]. Cruz et al. [58] suggest that such an assumption is valid for bench-scale PSA processes. They suggest that an overall material balance to obtain velocity profile along the bed length can be avoided and a constant or linear velocity profile is acceptable for PSA processes with low Reynolds number.

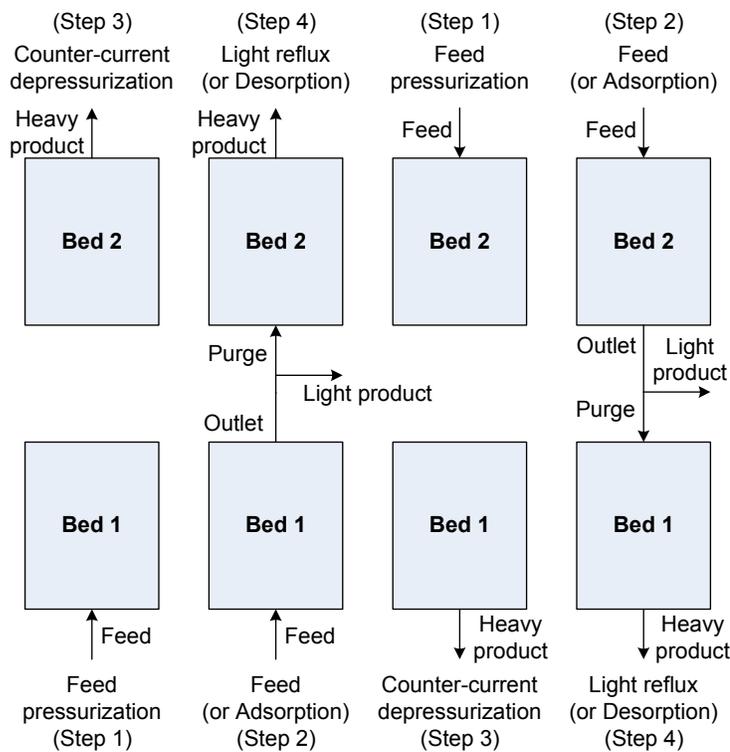


Figure 2.3: A 2-bed 4-step Skarstrom cycle [176]

2.4 PSA Operation

As described in section 2.1, fixed-bed pressure swing adsorption processes typically operate in a cyclic manner undergoing adsorption and desorption steps periodically in one or more packed beds. Desorption step renders clean beds re-usable for adsorption step. In PSA, bulk separation is achieved by “swinging” between high and low pressure levels for adsorption and desorption, respectively. Adsorption is carried out at a superambient pressure while the desorption is achieved at a near-ambient pressure level. In a vacuum PSA process (VPSA), desorption is carried out under vacuum. Larger pressure difference allows efficient separation as well as less adsorbent usage [206, 156].

2.4.1 Operation Scheme

A typical two-bed operation mode of PSA cycle is shown in Figure 2.3. The early PSA cycles developed by Skarstrom [176] and Air Liquide [60] utilized such an operating strategy. The

operation consists of four distinct steps, pressurization, adsorption, depressurization (counter-current), and light reflux (or desorption). In the first step (Step 1), bed 1 is pressurized by high-pressure feed gas from feed end, while bed 2 is depressurized in a counter-current fashion (Step 3) and strongly-adsorbed component (heavy product) is removed. Next, high-pressure feed gas is continued through bed 1 where adsorption of heavy product (Step 2) takes place while product gas enriched in weakly-adsorbed component (light product) leaves the top. During this period, a fraction of the light product gas is drawn out to bed 2 at low pressure to purge and further desorb the accumulated heavy adsorbate counter-currently, called the light reflux step (Step 4). Further, the beds interchange roles and execute previous steps of the other bed. Eventually, both beds repeat all the four steps in a cyclic manner.

The idea behind the light reflux step is to flush the void spaces within the bed and to ensure that at least the upper end of the bed, where light product is withdrawn, is completely free of the heavy product. Moreover, counter-current operation during depressurization and light reflux steps prevents retention of the heavy product at least at the upper end of the bed, thereby reducing the amount of purge used in Step 4. Increasing the purge amount increases the light product purity but inevitably reduces its recovery. Also, the operating pressure during Step 2 substantially influences light product losses during Step 3 and 4. Higher adsorption pressure increases losses during Step 3, while low pressure increases losses during Step 4 [156].

The Skarstrom cycle represents the most basic operation of a PSA process. To improve the purity and recovery of light or heavy products or both, as well as to design PSA processes for multi-component feed mixtures, several modified configurations have been proposed in the literature with a multitude of distinct operating steps such as light product or heavy product pressurization, desorption with heavy product as purge gas, co-current depressurization, pressure equalization etc. [43, 62, 129, 196, 213, 202]. The processes differ from one another with respect to the kinds of operating steps as well as the sequence in which these steps are carried out.

Industrial PSA operations adopt more sophisticated modes to increase product purity and

Stage 1				Stage 2			Stage 3			Stage 4			Stage 5		
Bed A	Feed			EQ1	EQ2	Pr purge	EQ3	Blow- down	Purge	EQ3	EQ2	Idle	EQ1	Repress	
Bed B	EQ1	Repress		Feed			EQ1	EQ2	Pr purge	EQ3	Blow- down	Purge	EQ3	EQ2	Idle
Bed C	EQ3	EQ2	Idle	EQ1	Repress		Feed			EQ1	EQ2	Pr purge	EQ3	Blow- down	Purge
Bed D	EQ3	Blow- down	Purge	EQ3	EQ2	Idle	EQ1	Repress		Feed			EQ1	EQ2	Pr purge
Bed E	EQ1	EQ2	Pr purge	EQ3	Blow- down	Purge	EQ3	EQ2	Idle	EQ1	Repress		Feed		

Figure 2.4: Time chart for a 5-bed 11-step H₂ PSA process [100]

recovery or minimize overall power consumption and operating costs. A practical PSA/VPSA process can be fairly complex with a multicolumn design executing a wide variety of nonisothermal, nonisobaric, and non-steady-state operating steps in a non-trivial sequence. For instance, Figure 2.4 shows the time chart for a 5-bed 11-step PSA process which separates hydrogen from a multicomponent feed mixture at a purity of more than 99.9999% [100]. Besides the conventional feed (adsorption) and purge (light reflux) steps, the cycle comprises multiple pressure equalization steps (EQ) and unconventional blowdown (depressurization) with pressure equalization step.

2.4.2 Cyclic Steady State

PSA processes are no more complex than most of the conventional separation processes, but they are different in one essential feature: the process always operates under transient conditions. Since the time intervals for operating steps are usually short and boundary condition around the PSA beds change as we switch from one operating step to another, the process never reaches a steady state. Consequently, behavior of a transient PSA process is always described by a set of partial differential equations (PDEs) which requires more complex solution procedure.

PSA processes differ from the conventional separation processes in one more feature: they operate under cyclic steady state (CSS). At CSS, conditions in each bed at the end of each cycle are exactly the same as those at the beginning of the cycle. In other words, although the process remains dynamic within a cycle, the transient behavior of the entire cycle remains

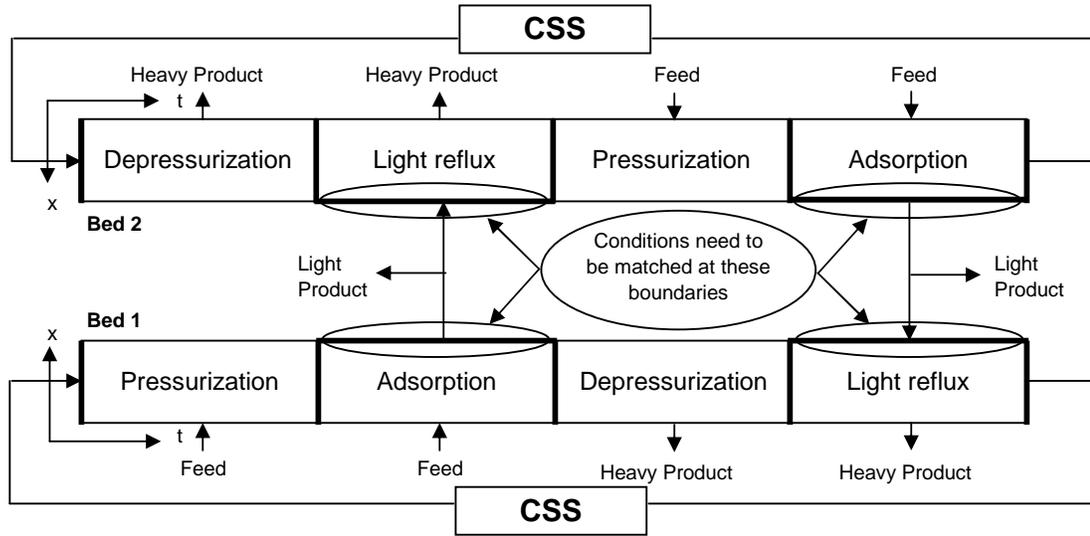


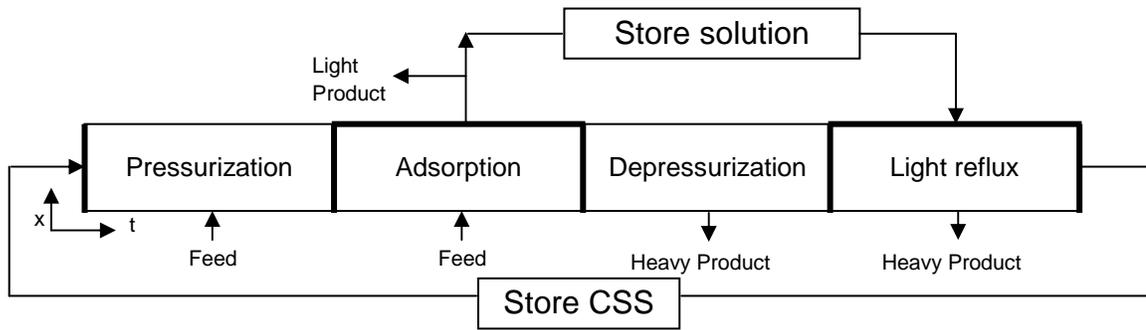
Figure 2.5: Boundary matchings required for the two-bed four-step PSA process

constant and repeats itself invariably from cycle to cycle. Mathematically, CSS is represented by matching the initial conditions of the PDEs with the solution obtained at the end of the cycle. Thus, we note that the initial conditions required to solve the set of PDEs of a PSA process are themselves parametric and should be computed simultaneously with the solution of PDEs.

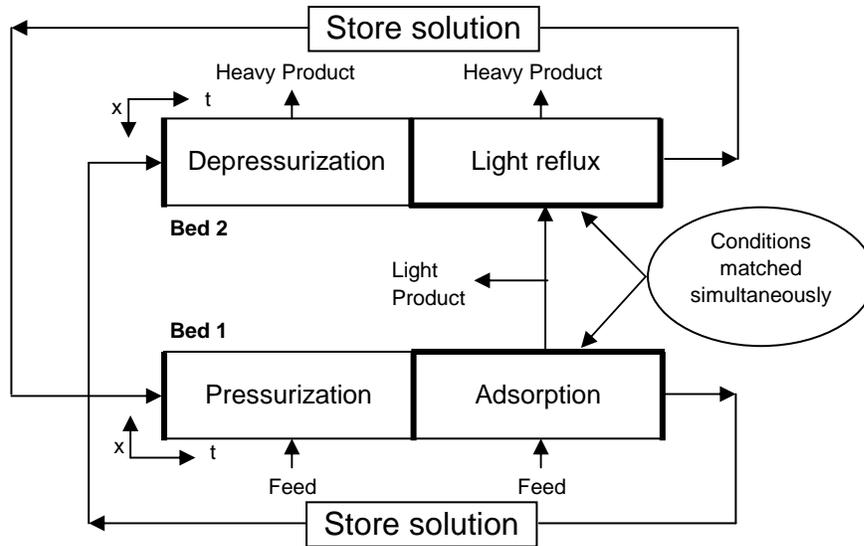
The number of cycles required by an actual PSA process to go from start-up to CSS are system dependent, but typically quite large. For instance, the number of cycles required to reach CSS are around 500 for H_2 PSA while 2000 for O_2 VPSA process [114]. Normally, CSS is determined by solving the PDE system repeatedly for each step of the cyclic process in sequence, using the final concentration profile for each step as the initial condition for the next step in the cycle. Such computations are bulky since the procedure is repeated sufficiently for a large number of cycles.

2.4.3 Boundary-matching for Multi-bed Operation

In a multi-bed PSA operation, beds interact with each other over the cycle as material flows from one bed to another, such as during the reflux and pressure equalization steps. We need to capture such an interaction at the boundary of the beds while simulating the multi-bed system.



(a) Unibed approach



(b) Multibed approach

Figure 2.6: Approaches to match boundary values while simulating multi-bed PSA operations

Moreover, due to CSS, conditions for the same bed should be matched at the beginning and the end of the cycle. For instance, Figure 2.5 illustrates such boundary matchings with the spatial domain and time chart of the two-bed four-step process shown in Figure 2.3. First, since a small amount of the light product exiting during the adsorption step is recycled as a reflux to the second bed for desorption, the conditions at the ends of the beds need to be matched. Next, since the final bed conditions at the end of an operating step serve as the initial condition for the next step, bed profiles should be matched. Finally, for CSS, bed profiles at the beginning and the end of the cycle should be matched. The boundaries where conditions need to be matched are shown in bold in Figure 2.5.

Unibed and *Multibed* formulations are two different strategies to implement aforementioned boundary matchings. As shown in Figure 2.6(a), Unibed approach involves simulating a single bed for all the operating steps in a cycle. Since all the PSA beds follow same sequence of steps and demonstrate identical dynamic behavior, one bed is sufficient to simulate the entire PSA process. Hence, only one set of bed equations are solved with varying boundary conditions for all the steps. To simulate bed interactions and match boundary information for different beds, storage buffers are used in the model implementation. In contrast, in Multibed approach we simulate all the beds in the PSA flowsheet but only for a portion of the cycle. As illustrated in Figure 2.6(b) for a 2-bed 4-step process, this portion of the cycle is selected such that it covers all the operating steps of the cycle among all the beds. Thus, we solve bed equations for all the beds but only for one set of operating steps. Such an approach accurately simulates bed interactions by matching boundary information simultaneously. By imposing a bed profile match at the beginning of one bed and end of another bed, solution of the entire PSA cycle is eventually obtained. Ling Jiang [99] provides a detailed description of Unibed and Multibed approaches.

2.5 Simulation Methodologies

As discussed in section 2.3, PSA processes are mathematically modeled by coupled hyperbolic partial differential and algebraic equations (PDAEs) distributed in both space and time. Obtaining analytical solution without making any approximation is close to impossible for such a highly coupled set of PDAEs. Low dimensional PDAEs of this type (with simplifications) can be solved analytically or directly by the PDE package CLAWPACK [123]. However, for large-scale systems numerical methods employing discretization for the spatial or the temporal or both domains are essential. There are two distinct approaches to numerically simulate the set of PDAEs:

- **Method of Lines (MOL)**

Method of Lines is a two-step approach [160]. First, PDAEs are discretized in the spatial

domain thus converting them to a set of differential algebraic equations (DAEs). Next, DAEs are integrated using standard time integration routines. One of the advantages with MOL is that with a high-resolution spatial discretization as well as with the error checking mechanisms present in the commercial time integration routines, we can achieve high order accuracy in both dimensions. A solution of the hyperbolic PDAEs defining PSA systems is usually characterized by steep adsorption fronts in the spatial domain, which is aggravated with higher adsorption pressure. In order to capture such steep fronts, it is essential to use a spatial discretization scheme which not only avoids physically unrealistic oscillations, but also ensures minimal numerical dispersion and negligible smearing (damping) with fewer discretization nodes. Moreover, it should also ensure extreme accuracy with mass, momentum, and energy conservation, which is vital for PSA-based separations. Because first- and second-order finite difference and finite element methods do not often mitigate such numerical noise with hyperbolic systems, high-resolution schemes such as upwind-based finite volume methods are utilized for spatial discretization [57, 124, 125, 197].

- **Complete Discretization (CD)**

In Complete Discretization approach, we discretize both spatial and temporal domains simultaneously, which leads to a large set of nonlinear algebraic equations. Such an approach allows seamless addition of the CSS condition to the discretized bed model, thus allowing entire system to be solved simultaneously by a Newton-based equation solver. Hence, CD is attractive and efficient for simpler PSA models. Nilchan and Pantelides [138, 139] successfully demonstrated this approach for small-scale PSA problems and solved the resulting algebraic system using a nonlinear equation solver inside gPROMS [2]. However, one of the drawbacks of CD is that in the absence of any error checking mechanism for temporal integration, pre-determined temporal discretization can lead to an inaccurate solution due to an insufficient number of temporal discretization nodes. While a large number of nodes can mitigate this drawback, it can consequently make the

problem computationally challenging to solve. Moreover, initialization of thousands of variables simultaneously also presents a significant challenge. Hence, after solving with the CD approach with fewer nodes, we verify the accuracy of the results with a more accurate MOL approach for our case studies.

We present a brief description of the spatial and the temporal discretization schemes used in this work in the subsequent sections.

2.5.1 Spatial Discretization

PSA processes are convection dominated and finding accurate discretization scheme that resolves sharp adsorption fronts is an additional challenge. Although higher-order methods accurately predict behaviour for smooth regions, they introduce unrealistic oscillations around steep zones in the spatial profile. First-order methods can be inherently bounded and can ensure no oscillatory behavior for steep zones. However, due to low order accuracy, fronts lose their sharpness (numerical smearing) unless hundreds of spatial nodes are used. Therefore, high-resolution methods are utilized for convection dominated processes which ensure second or higher order accuracy for smooth regions together with low order accuracy without oscillations for steep regions and discontinuities in the spatial profile [124, 70]. In particular, we use the *finite volume method* with flux limiters in this work since it not only avoids these aforementioned issues but also is well suited to model hyperbolic conservation laws, given its inherent conservative property [58, 114, 124].

Finite volume methods with flux limiters utilize the theory of flux-corrected transport (FCT) which incorporates anti-diffusion to negate the effects of excessive smearing. FCT removes numerical dispersion from the discretized equations to keep the fronts sharp ensuring no oscillations [32]. On the other hand, finite volume methods with a flux limiter conceptually are exact opposite of FCT. In other words, they introduce additional numerical dispersion around steep zones to avoid oscillations. However, the theory of flux limiters has been derived from FCT [91, 125].

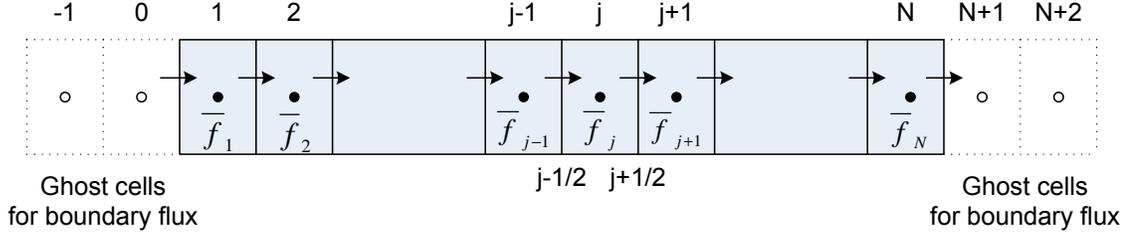


Figure 2.7: Finite volume discretization scheme

In a finite volume method, the spatial domain is divided into discrete volume elements (or cells) and we define average values for state variables over each element. For instance, for one-dimensional finite volume method, spatial division is done as shown in Figure 2.7, and average density over each element j is defined as

$$\int_{x_{j-1/2}}^{x_{j+1/2}} f(x) dx = \Delta_j \bar{f}_j \quad (2.18)$$

Here $j \pm 1/2$ are walls of volume j , Δ_j is the length of the volume j , and \bar{f}_j is the volume average of $f(x)$ which for example can represent bulk phase mass concentration or enthalpy. PDAEs are then integrated in the spatial domain and the state variables are replaced by their cell average values. For instance, Equation (2.14) after applying finite volume discretization becomes (for j -th cell and i -th component)

$$\epsilon_b \frac{d\bar{C}_{i,j}}{dt} + (1 - \epsilon_b) \rho_s \frac{d\bar{q}_{i,j}}{dt} + \frac{1}{\Delta_j} (v_{j+1/2} C_{i,j+1/2} - v_{j-1/2} C_{i,j-1/2}) = 0 \quad (2.19)$$

Here $v_{j+1/2} C_{i,j+1/2}$ and $v_{j-1/2} C_{i,j-1/2}$ are mass fluxes across the walls of volume j , resulting from the approximation of the integral in Equation (2.18). Since Equation (2.17) (or any other equation) always evaluates velocity at cell walls, only $C_{i,j+1/2}, C_{i,j-1/2}$ (or in general, wall values of the densities) need to be approximated in terms of cell average values by interpolation. For upwind finite volume methods, such an interpolation depends on the direction of the flux. In this work, we use the following flux direction-based formulation to approximate wall values of densities, such as bulk phase mass concentration or enthalpy [59, 91, 125]

$$\text{For } v_{j+1/2} \geq 0 \quad f_{j+1/2} = \bar{f}_j + \frac{1}{2}\theta(r_{j+1/2})(\bar{f}_{j+1} - \bar{f}_j) \quad (2.20a)$$

$$r_{j+1/2} = \frac{\bar{f}_j - \bar{f}_{j-1}}{\bar{f}_{j+1} - \bar{f}_j} \quad (2.20b)$$

$$\text{For } v_{j+1/2} < 0 \quad f_{j+1/2} = \bar{f}_{j+1} + \frac{1}{2}\theta(r_{j+1/2})(\bar{f}_j - \bar{f}_{j+1}) \quad (2.20c)$$

$$r_{j+1/2} = \frac{\bar{f}_{j+1} - \bar{f}_{j+2}}{\bar{f}_j - \bar{f}_{j+1}} \quad (2.20d)$$

Here \bar{f}_j is the cell average value for state variable such as gas-phase concentration C , solid loading q , or temperature T , $\theta(r)$ is the flux limiter and $r_{j+1/2}$ is a ratio which measures the smoothness of the profile. If $r_{j+1/2}$ is close to 1, the profile is presumably smooth. If $r_{j+1/2}$ is far from 1, there must be a sharp discontinuity at x_j . Depending on the value of $r_{j+1/2}$, $\theta(r)$ applies proper correction. If the profile is smooth, $\theta(r)$ preserves second or higher-order nature of the discretization, otherwise near steep regions $\theta(r)$ reduces the order of the discretization to eliminate oscillations.

Flux limiters take various forms to perform aforementioned functions. Darwish et al. [59] and Hirsch [91] give a detailed description of several flux limiters such as Minmod, Superbee, and Van Leer limiters. While Minmod limiter is too diffusive and Superbee doesn't perform adequately for smooth regions, Van Leer has properties between the two and thus is more desirable. Hence, we use the Van Leer flux limiter for our case studies, which has the following form

$$\theta(r) = \frac{r + |r|}{1 + |r|} \quad (2.21)$$

Boundary conditions are incorporated in the finite volume scheme with the help of "ghost cells", as illustrated in Figure 2.7. Ghost cells are required because Dirichlet or Neumann boundary conditions specified for the problem apply to the walls of the first or N^{th} finite volume and they need to be translated to corresponding cell average value to get accounted for in the discretization scheme. Thus, boundary conditions at the walls are usually translated to the average values of fictitious ghost cells using some form of extrapolation. For instance, if f_{inlet} and f_{outlet} are given as boundary conditions and a linear extrapolation scheme is chosen,

they get translated as

$$\bar{f}_0 = 2f_{inlet} - \bar{f}_1 \quad (2.22a)$$

$$\bar{f}_{-1} = 4f_{inlet} - 3\bar{f}_1 \quad (2.22b)$$

$$\bar{f}_{N+1} = 2f_{outlet} - \bar{f}_N \quad (2.22c)$$

$$\bar{f}_{N+2} = 4f_{outlet} - 3\bar{f}_N \quad (2.22d)$$

where \bar{f}_0 , \bar{f}_{-1} , \bar{f}_{N+1} , and \bar{f}_{N+2} are average values for ghost cells.

2.5.2 Temporal Discretization

For temporal discretization of the DAE system obtained after spatial discretization, we employ orthogonal collocation on finite elements (OCFE) technique for our work [40, 69]. OCFE is a discretization scheme which combines the method of weighted residuals with the finite element methods. The state temporal profiles are approximated at a finite number of points - the collocation points - by a family of polynomials.

To illustrate the concept, we consider the following set of ordinary differential equation (ODE)

$$\frac{dy}{dt} = f(y(t), t), \quad y(t_0) = y_0 \quad (2.23)$$

For discretization, the time domain is partitioned into n_E finite elements of length h_i , $i \in [1, \dots, n_E]$ such that $\sum_{i=1}^{n_E} h_i = t_f - t_0$, where t_0 and t_f are initial and final time, respectively. Thus, time at the end of each element i is defined as $t_i = t_0 + \sum_{m=1}^i h_m$. Next, we represent the time derivative of the state as a Lagrange polynomial of order n_C , where n_C is the number of collocation points, for each element i [17]. This leads to

$$\begin{aligned} t &= t_{i-1} + h_i \tau, \quad \tau \in [0, 1], \quad t \in [t_{i-1}, t_i], \quad i = 1, \dots, n_E \\ y(t) &= y_{i,0} + h_i \sum_{j=1}^{n_C} \Omega_j(\tau) \left(\frac{dy}{dt} \right)_{i,j} \end{aligned} \quad (2.24)$$

Here τ_j , $j = 1, \dots, n_C$ are the collocation points which are usually the roots of an orthogonal polynomial of degree n_C , $y_{i,0}$ is the value of the state at the beginning of the element i ,

$\frac{dy}{dt}_{i,j} = \frac{dy}{dt}(t_{i,j})$ with $t_{i,j} = t_{i-1} + h_i\tau_j$, and $\Omega_j(\tau)$ is a polynomial of order n_C , satisfying

$$\begin{aligned}\Omega_j(\tau) &= \int_0^\tau l_j(\tau)d\tau, \quad \tau \in [0, 1], \quad t \in [t_{i-1}, t_i] \quad i = 1, \dots, n_E \\ \text{where } l_j(\tau) &= \prod_{k=1, \neq j}^{n_C} \frac{\tau - \tau_k}{\tau_j - \tau_k}, \quad j = 1, \dots, n_C \\ \Omega_j(0) &= 0, \quad \frac{d\Omega_j}{d\tau_k} = \delta_{j,k}, \quad j, k = 1, \dots, n_C\end{aligned}\tag{2.25}$$

Here we use Radau collocation points with $\tau_j < \tau_{j+1}$, $j = 1, \dots, n_C - 1$, and $\tau_{n_C} = 1$ for every element. Since the last collocation point lies at the end of the finite element i , continuity of the state profiles is easily ensured by setting $y_{i,0} = y(t_{i-1,n_C}) = y(t_{i-1})$.

$$y_{i,0} = y_{i-1,0} + h_{i-1} \sum_{j=1}^{n_C} \Omega_j(1) \left(\frac{dy}{dt} \right)_{i-1,j}\tag{2.26}$$

While state variables are continuous across the finite elements, control variables can present discontinuities at the boundaries of the elements. We prefer Radau collocation points because they allow us to set constraints easily at the end of each element in an optimization problem, and to stabilize the system more efficiently if high index DAEs are present [33, 28]. To determine polynomial coefficients $\frac{dy}{dt}_{i,j}$, we substitute Equation (2.24) into Equation (2.23) and enforce the resulting algebraic equations at the collocation points τ_j , which leads to

$$\left(\frac{dy}{dt} \right)_{i,j} = \frac{dy}{dt}(t_{i,j}) = f(y(t_{i,j}), t_{i,j}), \quad j = 1, \dots, n_C, \quad i = 1, \dots, n_E\tag{2.27}$$

where $y(t_{i,j})$ is obtained from Equation (2.24).

2.6 Concluding Remarks

Beginning with a review of the fundamentals of adsorption phenomena, we described how the physics of pressure swing adsorption processes can be modeled mathematically in the form of PDAEs. It is clear from the description that the fundamentals at the molecular level are relatively well understood and characterized, and it is possible to construct fairly

accurate models to predict dynamic behavior of pressure swing adsorption processes. Currently, research efforts are aimed at improvements in multicomponent mixture isotherms, and better understanding of mass transfer phenomena, axial dispersion and fluid transport within packed beds [169, 206].

We also discussed operation strategies for two-bed as well as multi-bed PSA processes, and mentioned that industrial PSA systems carry out high purity separations with the help of sophisticated cycles involving complex sequences of operating steps. Designing such complicated sequences and PSA processes is generally non-intuitive, and thus a systematic methodology is desired which can ameliorate the arduous task of synthesizing PSA cycles. In the next few chapters, we present a novel optimization-based framework to design optimal PSA cycles, and illustrate it with the help of examples motivated from the application of PSA for carbon capture.

Finally, we showed that bed model for PSA processes is defined by hyperbolic partial differential and algebraic equations (PDAEs) with high nonlinearities arising from non-isothermal effects and nonlinear adsorption isotherms, and with solution profiles represented by steep adsorption fronts. Optimization of such systems presents a significant computational challenge to current algorithms. To address this, we present in subsequent chapters a novel optimization algorithm based on reduced-order modeling.

Chapter 3

PSA Superstructure

Synopsis

This chapter presents a systematic optimization-based formulation to synthesize optimal PSA cycles for a given application. In particular, we present a novel PSA superstructure to simultaneously determine new configurations and design parameters. The superstructure consists of two beds, one of which acts as an adsorbing bed and the other as a desorbing bed. The interconnections between the two beds are governed by time-dependent control variables, such as fractions of the light and the heavy product recycle. The superstructure is rich enough to predict a number of different PSA operating steps, which are accomplished by varying these control variables. An optimal sequence of operating steps is achieved by solving an optimal control problem for the superstructure with the partial differential and algebraic equations of the PSA system and the cyclic steady state condition. We also present the PDAEs for the bed model with the connectivity equations of the superstructure. Large-scale optimization capabilities have enabled us to adopt a complete discretization methodology to solve the optimal control problem as a large-scale nonlinear program.

3.1 Motivation

Industrial PSA/VSA systems are quite intricate involving multiple adsorber columns which execute a non-intuitive complex sequence of nonisothermal, nonisobaric, and non-steady-state operating steps. Synthesizing such configurations for given commercial specifications has so far relied on thumb rules, past experiences in adsorption design, or immense experimental effort with bench- or pilot-scale processes. A systematic methodology to design, evaluate and optimize novel PSA cycle configurations hasn't been reported in the literature to date due to the inherent complexity of the process. Cycle design with accurate, reliable, and rigorous PSA bed models is considered prohibitive because of the expense and computational time involved.

Very few studies in the literature have tried to address this issue. All of these studies suggest simplistic formulations to determine minimum number of beds required in a PSA process for given kinds and fixed sequences of operating steps, but do not discuss how these steps should be chosen to form a cycle. Zhang and Webley [211] outlined an approach for cycle development by understanding the roles of individual operating steps and adsorption fronts. However, they identified optimal configurations with the help of a pre-decided set of operating steps and a simplified mathematical model. Chiang [46] proposed simple arithmetic-based heuristics, while Smith et al. [177] extended Chiang's work to propose a mixed-integer nonlinear programming (MINLP) based approach to obtain optimal number of beds required to execute a fixed sequence of operating steps. Smith et al. [178, 179] also suggested a 3-step scenario to design an industrial PSA system, but again with a known cycle of operating steps. Recently, Nikolić et al. [136, 137] proposed a state-task network (STN) based framework to determine optimal number of beds, with operating steps forming the states of the network. The kinds and sequences of operating steps chosen were fixed in their case as well. Moreover, STN developed wasn't exhaustive and missed many basic steps such as product repressurization, co-current depressurization, and desorption with purge stream coming from another bed.

In contrast, we present a novel superstructure-based approach to obtain optimal sequence of operating steps in a PSA cycle without any assumption on the kinds of steps that should

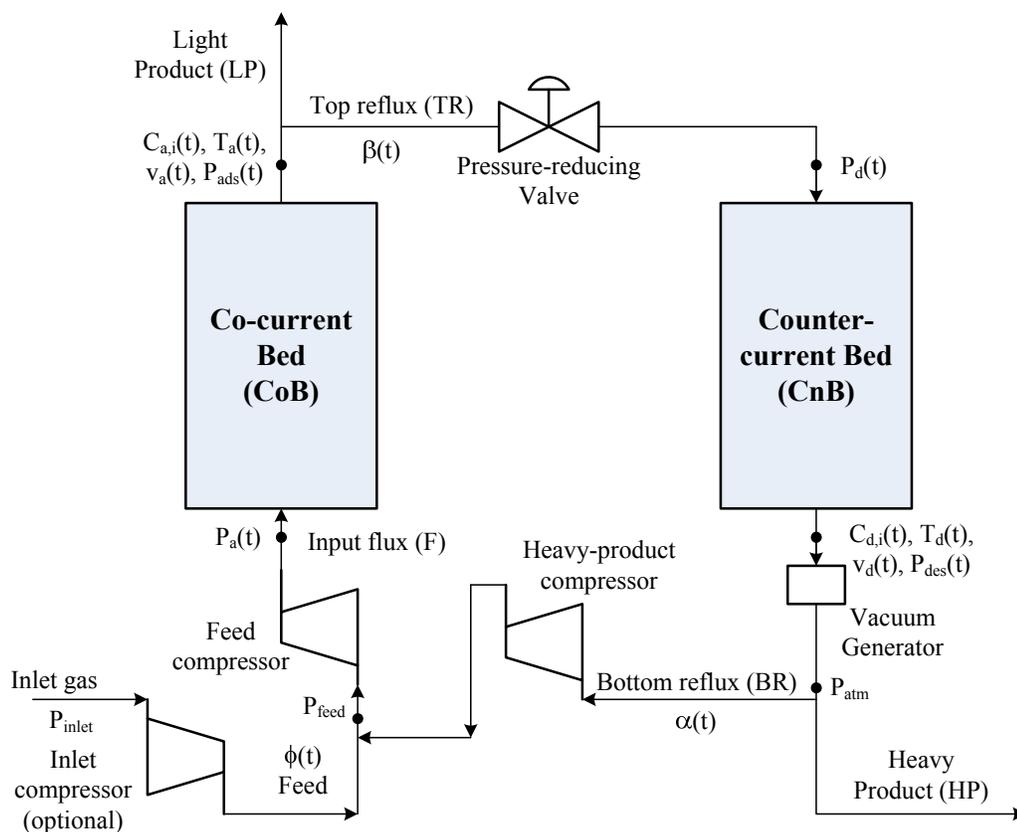


Figure 3.1: A 2-bed PSA Superstructure

be included in the cycle. With development of optimization strategies for process synthesis, it is natural to consider a superstructure based approach to design PSA processes. As discussed in the next section, this approach relies on the formulation of an optimal control problem.

3.2 Methodology

3.2.1 Superstructure

Figure 3.1 illustrates the proposed 2-bed PSA superstructure. It has a co-current bed (CoB) and a counter-current bed (CnB) that determine co-current and counter-current operating steps in the cycle, respectively. We consider only two beds to ensure that the direction of the flow, and thus the superficial velocity, remains co-current for CoB and counter-current for CnB. This strategy avoids flow reversals in the bed, and does not require additional bed connections with embedded logical conditions in order to realize different operating steps. This

superstructure is consistent with the concept of *unibed* models [114], where no more than two beds interact at the same time, and the steps can be grouped into *adsorbing steps* and *desorbing steps*. Consequently, it can accomplish a wide variety of operating steps with just a single bed connection, as shown in Figure 3.1. Furthermore, this helps to avoid discrete variables and uses only continuous variables for the optimization problem.

The superstructure is designed to get the light product from the upper end (light end) of CoB and heavy product from the lower end (heavy end) of CnB. The time dependent variables $\beta(t)$ and $\alpha(t)$ determine the fraction of the light product and the heavy product streams that go in the top and the bottom reflux, respectively. If the feed gas (or inlet gas) comes at a low pressure which is close to atmospheric, it is first compressed from its pressure P_{inlet} to P_{feed} through the optional inlet compressor, before being compressed from P_{feed} to P_a using the feed compressor. The time dependent feed fraction $\phi(t)$ determines the feeding strategy. For CoB, pressure is specified at the light end by P_{ads} , while the pressure at the other end P_a is determined from the pressure drop in this bed. The velocity v_a , concentration for i^{th} component $C_{a,i}$, and temperature T_a at the light end are determined from the outlet flux. Similarly for CnB, pressure is specified at the heavy end by P_{des} , while $C_{d,i}$, T_d and v_d are obtained from the output flux, and P_d is obtained from the pressure drop. The superstructure also incorporates compressors and valves to account for different pressure levels in the beds, and a vacuum generator to extract the strongly-adsorbed component.

3.2.2 Cycle Realization

It is possible to accomplish a wide variety of different operating steps of a PSA process by varying control variables $\alpha(t)$, $\beta(t)$, $\phi(t)$, $P_{ads}(t)$ and $P_{des}(t)$. For instance, as shown in Figure 3.2, if we set $\alpha = 0.5$, $\beta = 0$ and $\phi = 0$ then the feed and the top reflux are turned off, and a part of the heavy-product is recycled back to the co-current bed as a bottom reflux. Thus, this results in a counter-current depressurization (CnD) step for the counter-current bed and a heavy reflux (HR) step for the co-current bed. Similarly, if we set $\alpha = 0$, $\beta = 0.5$ and $\phi = 0$

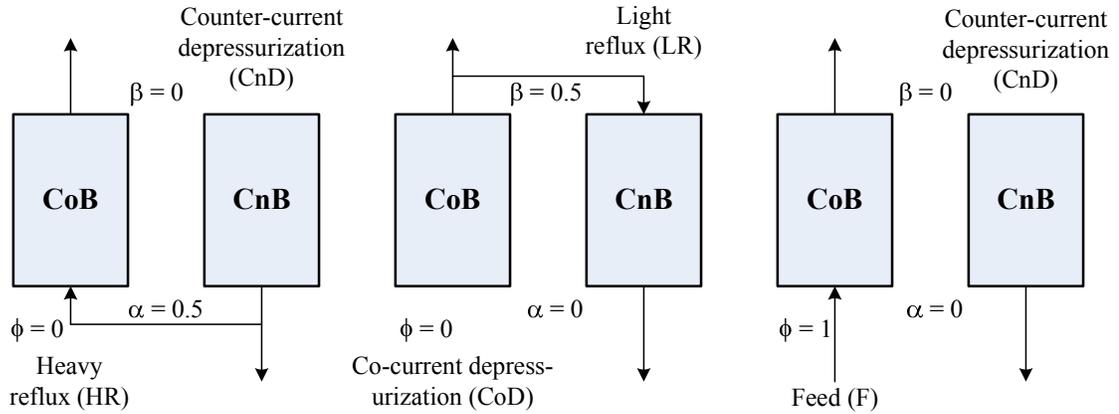


Figure 3.2: Realization of different operating steps by varying control variables

then we realize a co-current depressurization (CoD) step and a light reflux (LR) step, or if we set $\alpha = 0$, $\beta = 0$ and $\phi = 1$ then we get an adsorption step (F) and a counter-current depressurization (CnD) step.

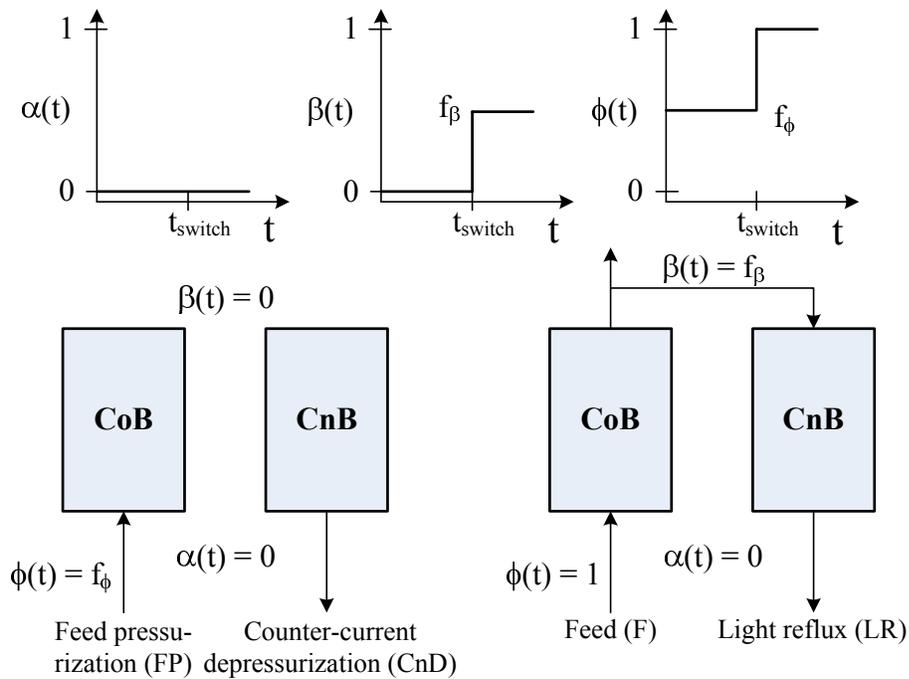


Figure 3.3: Realization of 4-step Skarstrom cycle from the superstructure

As a consequence, temporal profiles of $\alpha(t)$, $\beta(t)$, $\phi(t)$, $P_{ads}(t)$ and $P_{des}(t)$ result in a sequence of operating steps. By translating these temporal profiles into a sequence of meaningful operating steps, we eventually obtain a complete PSA cycle. For instance, the profiles of $\alpha(t)$,

$\beta(t)$, and $\phi(t)$ shown in Figure 3.3 translate into the classical 2-bed 4-step Skarstrom cycle (FP,F,CnD,LR) [176]. CoB generates pressurization (FP) and feed (F) steps, while CnB simultaneously generates depressurization (CnD) and light reflux (LR) steps. Thus, the overall cycle includes these four steps (FP,F,CnD,LR). In an actual 2-bed PSA unit, after performing its steps, CoB will follow the steps of CnB and vice-versa. However, in the mathematical framework, this is realized by giving final conditions of CoB as the initial conditions for CnB and vice-versa, thus modeling the true 2-bed behavior. In other words, we utilize *Multibed* approach and simulate both beds simultaneously (*cf.* section 2.4.3).

3.2.3 Optimal Control Problem

It is possible to obtain a plethora of distinct PSA configurations with this superstructure because of an endless number of shapes that the profiles of $\alpha(t)$, $\beta(t)$, $\phi(t)$, $P_{ads}(t)$ and $P_{des}(t)$ can take. As a consequence of this, we obtain an optimal sequence of operating steps, along with other decision variables such as cycle time, step times and bed dimensions, by solving the following optimization problem.

$$\begin{aligned}
\min \quad & \Phi(z(x, t_f), y(x, t_f), \alpha(t_f), \beta(t_f), \phi(t_f), P_{ads}(t_f), P_{des}(t_f), z_0, p) \\
\text{s.t.} \quad & f\left(\frac{\partial z}{\partial t}, \frac{\partial z}{\partial x}, z(x, t), y(x, t), \alpha(t), \beta(t), \phi(t), P_{ads}(t), P_{des}(t), z_0, p\right) = 0 \\
& z_{CoB}(x, 0) = z_{CnB}(x, t_f) \\
& z_{CnB}(x, 0) = z_{CoB}(x, t_f) \\
& s(z(x, t), y(x, t), \alpha(t), \beta(t), \phi(t), P_{ads}(t), P_{des}(t), p) = 0 \\
& g(z(x, t), y(x, t), \alpha(t), \beta(t), \phi(t), P_{ads}(t), P_{des}(t), p) \leq 0 \\
& 0 \leq (\alpha(t), \beta(t), \phi(t)) \leq 1 \\
& b_L \leq (P_{ads}(t), P_{des}(t), p) \leq b_U
\end{aligned} \tag{3.1}$$

Here Φ is the objective function related to overall power consumption, component purity or recovery. It can depend upon differential variables $z(x, t)$, algebraic variables $y(x, t)$, control variables $\alpha(t)$, $\beta(t)$, $\phi(t)$, $P_{ads}(t)$, and $P_{des}(t)$, initial conditions z_0 and other decision variables

p . The first equation represents the PDAE-based model for the PSA system, while the second equation is the cyclic steady state (CSS) condition (see Table 3.1). As mentioned before, the CSS condition is implemented by giving the final conditions of CoB as the initial condition for CnB and vice versa. Additional constraints for the optimization problem are given by the algebraic equations s and the inequalities g . The control variables $\alpha(t)$, $\beta(t)$ and $\phi(t)$ are fractions bounded between 0 and 1. Other control variables, $P_{ads}(t)$ and $P_{des}(t)$, and decision variables p are bounded between their respective bounds b_L and b_U .

It is important to note that although optimal 2-bed PSA configurations are construed from the optimal profiles of $\alpha(t)$, $\beta(t)$, $\phi(t)$, $P_{ads}(t)$ and $P_{des}(t)$, multibed cycles (with more than two beds) follow immediately from these solutions. These are generated by staggering the steps over multiple beds and ensuring that a bed with a product flow step occurs at all points in time.

3.3 Model Equations

We consider a detailed PDAE-based mathematical model for the optimal control problem. The model is fairly general and can also be extended beyond the following assumptions:

1. All of the gases follow the ideal gas law.
2. There are no radial variations in temperature, pressure and concentrations of the gases in the solid and the gas phase.
3. The gas and the solid phases are in thermal equilibrium and bulk density of the solid phase remains constant.
4. Pressure drop along the bed is calculated by the Ergun equation.
5. The adsorption behaviors are described by the dual-site Langmuir isotherm.
6. The adsorption rate is approximated by the linear driving force (LDF) expression. Sircar and Hufton [172] demonstrated that the LDF model is sufficient to capture the kinetics

Table 3.1: PSA Model Equations

Component mass balance

$$\epsilon_b \frac{\partial C_i}{\partial t} + (1 - \epsilon_b) \rho_s \frac{\partial q_i}{\partial t} + \frac{\partial(vC_i)}{\partial x} = 0 \quad i \in \{L, H\} \quad (3.2)$$

LDF equation

$$\frac{\partial q_i}{\partial t} = k_i(q_i^* - q_i) \quad i \in \{L, H\} \quad (3.3)$$

Energy balance

$$\left(\epsilon_b \sum_i C_i (C_{pg}^i - R) + \rho_b C_{ps} \right) \frac{\partial T}{\partial t} - \rho_b \sum_i \Delta H_i^{ads} \frac{\partial q_i}{\partial t} + \frac{\partial(vh)}{\partial x} + U_A(T - T_w) = 0 \quad (3.4)$$

$$C_{pg}^i = a_c^i + b_c^i T + c_c^i T^2 + d_c^i T^3 \quad i \in \{L, H\}$$

$$h = \sum_i \left(C_i \int C_{pg}^i dT \right)$$

Dual-site Langmuir Isotherm

$$q_i^* = \frac{q_{1i}^s b_{1i} C_i RT}{1 + \sum_j b_{1j} C_j RT} + \frac{q_{2i}^s b_{2i} C_i RT}{1 + \sum_j b_{2j} C_j RT} \quad i \in \{L, H\} \quad (3.5)$$

$$q_{mi}^s = k_{mi}^1 + k_{mi}^2 T \quad b_{mi} = k_{mi}^3 \exp\left(\frac{k_{mi}^4}{T}\right) \quad i \in \{L, H\} \quad m = 1, 2$$

Ergun equation

$$-\frac{\partial P}{\partial x} = \frac{150\mu(1 - \epsilon_b)^2}{d_p^2 \epsilon_b^3} v + \frac{1.75}{d_p} \left(\frac{1 - \epsilon_b}{\epsilon_b^3} \right) \left(\sum_i M_w^i C_i \right) v|v| \quad (3.6)$$

Ideal gas equation

$$P = RT \sum_i C_i \quad (3.7)$$

Bed connection equations (see Figure 3.1)

$$F_i(t) = \phi(t) v_{feed} C_{feed,i} + \alpha(t) (-v_d(t)) C_{i,d}(t) \quad i \in \{L, H\} \quad (3.8)$$

$$TR_i(t) = \beta(t) v_a(t) C_{a,i}(t) \quad i \in \{L, H\} \quad (3.9)$$

$$LP_i(t) = (1 - \beta(t)) v_a(t) C_{a,i}(t) \quad i \in \{L, H\} \quad (3.10)$$

$$BR_i(t) = \alpha(t) (-v_d(t)) C_{d,i}(t) \quad i \in \{L, H\} \quad (3.11)$$

$$HP_i(t) = (1 - \alpha(t)) (-v_d(t)) C_{d,i}(t) \quad i \in \{L, H\} \quad (3.12)$$

Cyclic Steady State (CSS) condition for beds CoB and CnB

$$z_{CoB}(x, 0) = z_{CnB}(x, t_f), \quad z_{CnB}(x, 0) = z_{CoB}(x, t_f), \quad (3.13)$$

$$z(x, t)^T = [C_L(x, t), C_H(x, t), q_L(x, t), q_H(x, t), T(x, t)]$$

of adsorption because estimation of the separation performance of an adsorptive process requires several sets of averaging of kinetic properties and the effect of local characteristics are lumped during integration.

Based on the above assumptions, the mathematical model for the PSA superstructure is listed in Table 3.1. The equations are written for light product L and heavy product H . Here we consider a lumped mass transfer coefficient for the LDF equation. Since a smaller magnitude of U_A makes energy balance a weak function of the ambient temperature, T_w is assumed constant. As a convention, flow in the counter-current bed is considered negative and a minus sign is used for v_d . Since the bed model is based on fluxes, the feed throughput and the bed diameter can be adjusted as long as the specified feed flux is achieved and the model assumptions are not violated. The purities and recoveries of light (L) and heavy (H) products are calculated by the following equations.

$$\text{purity}_L = \frac{\int (1 - \beta(t))v_a(t)C_{a,L}(t) dt}{\int (1 - \beta(t))v_a(t) \sum_i C_{a,i}(t) dt} \quad (3.14a)$$

$$\text{purity}_H = \frac{\int (1 - \alpha(t))(-v_d(t))C_{d,H}(t) dt}{\int (1 - \alpha(t))(-v_d(t)) \sum_i C_{d,i}(t) dt} \quad (3.14b)$$

$$\text{recovery}_L = \frac{\int (1 - \beta(t))v_a(t)C_{a,L}(t) dt}{Q_{feed,L}} \quad (3.14c)$$

$$\text{recovery}_H = \frac{\int (1 - \alpha(t))(-v_d(t))C_{d,H}(t) dt}{Q_{feed,H}} \quad (3.14d)$$

$$Q_{feed,i} = \int \phi(t)v_{feed}C_{feed,i} dt \quad i \in \{L, H\} \quad (3.14e)$$

Here Q_{feed} is the feed flux. The total power consumption, given by the following equations, is the sum of the work done by the compressors and the vacuum generator. We note that in the following equations we do not consider the compression work for the light or the heavy

product. If required, it can be directly included in the following equations.

$$\begin{aligned}
W_{total} = & \int \frac{\gamma RT_d}{\gamma - 1} \left[\frac{\sum_i (\phi(t) v_{feed} C_{feed,i} + \alpha(t) (-v_d) C_{d,i})}{\eta_c} \left(\left(\frac{P_a}{P_{feed}} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right) \right. \\
& + \frac{\sum_i \alpha(t) (-v_d) C_{d,i}}{\eta_h} \left(\min \left\{ \left(\frac{P_{feed}}{P_{atm}} \right)^{\frac{\gamma-1}{\gamma}}, \left(\frac{P_{feed}}{P_{des}} \right)^{\frac{\gamma-1}{\gamma}} \right\} - 1 \right) \\
& + \frac{\sum_i (-v_d) C_{d,i}}{\eta_v} \max \left\{ 0, \left(\left(\frac{P_{atm}}{P_{des}} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right) \right\} \left. \right] \\
& + \frac{\gamma}{\gamma - 1} \frac{\phi(t) v_{feed} P_{feed}}{\eta_{fg}} \left(\left(\frac{P_{feed}}{P_{inlet}} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right) dt \tag{3.15a}
\end{aligned}$$

$$\text{Power} = \frac{W_{total}}{\int (1 - \alpha(t)) v_d(t) C_{d,H}(t) dt} \tag{3.15b}$$

Here, the *max* function ensures that the work done by the vacuum generator is zero when P_{des} is more than the atmospheric pressure P_{atm} . Similarly, since the vacuum generator discharges heavy reflux at P_{atm} , the *min* function ensures a proper upstream pressure for the heavy product compressor. Since *min* and *max* functions introduce non-differentiability, the following smoothing approximations are adopted [20]. A value of 0.01 is used for ϵ in the following equations.

$$\min(f_1(x), f_2(x)) = f_1(x) - \max(0, f_1(x) - f_2(x)) \tag{3.16a}$$

$$\max(0, f(x)) = 0.5 \left(f(x) + \sqrt{f(x)^2 + \epsilon^2} \right) \tag{3.16b}$$

3.4 Solution Strategy

We adopt a complete discretization approach to solve the system of PDAEs in Table 3.1. The PDAEs are converted into a set of algebraic equations by discretizing the state and the control variables both in space and time. As a result, the PDAE-constrained optimal control problem (3.1) gets converted into a large-scale nonlinear programming (NLP) problem. One of the advantages of this approach is that it directly couples the solution of the PDAE system with the optimization problem. The model equations are solved only once at the optimum and the

excessive computational effort of getting intermediate solutions is avoided [28]. However, the performance of this approach substantially depends upon the optimization solver, and therefore it is crucial to choose an efficient NLP solver. Hence, we use the state-of-the-art NLP solver IPOPT 3.4 for our case studies. This interior point solver uses a barrier method to handle inequalities and exact second derivative information for faster convergence to the optimum [195].

To capture steep adsorption fronts, avoid oscillations in the solution, and model conservative properties of the system, we apply a first-order finite volume method for spatial discretization. For the temporal domain, we apply orthogonal collocation on finite elements with a Radau collocation scheme. Radau collocation allows us to set constraints at the ends of the finite elements [103]. A 3-point collocation scheme is used for state variables while control variables are considered to be piecewise constant. While control variables are allowed to be discontinuous, we ensure state variables demonstrate continuity in their profiles. We also consider a moving finite element strategy in which the size of each temporal finite element is considered a decision variable. With moving finite elements, it is possible to locate optimal breakpoints of the control variables with variable element lengths. Appropriate bounds are imposed on the variable element lengths of each finite element to guarantee accuracy of the discretization.

Because spatial discretization together with a pre-determined temporal discretization without any error checking mechanism for temporal integration can cause inaccuracies to creep in the NLP solution obtained from IPOPT, verification of the solution with an accurate dynamic simulation is essential. Therefore, we perform dynamic simulations in MATLAB [1] at the optimal values of the decision variables obtained from IPOPT. The DAE system obtained after applying method of lines is integrated in MATLAB at the optimal values of the decision variables. The profiles and performance variables obtained from MATLAB are then compared with those obtained from IPOPT. For the method of lines approach in MATLAB, we use a first-order finite volume method for spatial discretization and *ode15s* for temporal integration.

Finally, because all of the control variables appear linearly, problem (3.1) is a singular optimal control problem. In singular control problems, the optimal control profiles cannot be determined directly from the stationarity condition of the Hamiltonian. The Euler-Lagrange equations obtained after applying the maximum principle to (3.1) are high index in nature and ill-conditioned. This doesn't affect optimal controls that lie at their bounds where solution is "bang-bang". However, a bang-bang solution may not always be guaranteed with complex nonlinear state equations of PSA system as the Hamiltonian derivative w.r.t. controls can be zero for some time interval, leading to a singular control profile. Repeated time differentiations of the Hamiltonian derivative can recover the control, but identifying the beginning and the end of a singular arc is often difficult. Applying orthogonal collocation to singular problems can also reflect this behavior with an ill-conditioned reduced Hessian and solutions characterized by oscillations that do not abate with increasing mesh refinement [102]. To address singular problems, several approaches have been suggested which propose regularizations to improve eigenvalues of the reduced Hessian, and to guarantee a unique solution [97, 180, 101]. In particular, regularizations have been performed through coarse discretization of the control profile [161]. Applying a coarse control discretization hampers rapid decay of the eigenvalues of reduced Hessian, thus allowing a reasonable solution to the singular problem. Hence, we adopt this relatively simple regularization heuristic to ameliorate the singular nature of the control problem (3.1). Such an approach, coupled with moving finite element strategy, improves the bang-bang nature of the optimal solution, locates singular arcs and minimizes their length with fewer finite elements.

Chapter 4

Superstructure Case Study: Post-combustion CO₂ Capture

Synopsis

Recent developments have shown PSA/VSA to be a promising option to effectively capture CO₂ from flue gas streams. In most commercial PSA cycles, the weakly adsorbed component in the mixture is the desired product, and enriching the strongly adsorbed CO₂ is not a concern. On the other hand, it is necessary to concentrate CO₂ to high purity to reduce CO₂ sequestration costs and minimize safety and environmental risks. Thus, it is necessary to develop PSA processes specifically targeted to obtain pure strongly adsorbed component. We demonstrate the superstructure approach for case studies related to post-combustion CO₂ capture. In particular, optimal PSA cycles are synthesized which maximize CO₂ recovery and minimize overall power consumption. The results show the potential of the superstructure to predict PSA cycles with up to 98% purity and recovery of CO₂. Moreover, for purity of over 90%, these cycles can recover CO₂ from atmospheric flue gas with a low power consumption of 465 kWh/tonne CO₂ captured. Superstructure approach is, therefore, quite useful for assessing PSA for post-combustion CO₂ capture.

4.1 Introduction

Today, fossil fuels provide about 85% of the global energy demand and the outlook is that they will remain the dominant source of energy for decades to come. Consequently, global energy-related CO₂ emissions, especially from power plants that burn fossil fuels, have increased, thereby increasing CO₂ concentration levels in the atmosphere [93]. One option to mitigate the emission of CO₂ is to capture it from emission sources, store it in the ocean or underground, or use it for enhanced oil or coal bed methane recovery. Before CO₂ can be sequestered it must be separated and concentrated from flue gas with a low CO₂ concentration.

There are a variety of approaches to CO₂ separation from other flue gas components, such as gas absorption, membranes, cryogenic distillation, gas adsorption and others, each with their own pros and cons [3]. Absorption is a well-established method for separating CO₂. Currently, absorption based technologies are commercially utilized for CO₂ capture, in which different kinds of amines are used as solvents for absorbing CO₂ from flue gas. The greatest advantage of absorption processes is that these amine-based solvents can be easily regenerated. Moreover, these processes can capture CO₂ with purity higher than 95% which is enough for sequestration. However, solvent regeneration energy is quite high for absorption processes. Typical values of energy requirement for the leading absorption technologies range from 765 to 950 kWh/tonne CO₂ captured (excluding energy requirement for CO₂ compression) [94]. Moreover, solvent can form corrosive solutions with the flue gas. Another technology is cryogenic distillation in which CO₂ is captured by liquefying. One of its biggest advantage is that its product is liquid CO₂ which is ready for transport for sequestration. Moreover, CO₂ recovery can exceed 99.9%. However, these processes are extremely energy intensive and cannot tolerate H₂O, O₂, SO_x, and NO_x in the feed stream [3]. Membrane separation processes, on the other hand, though being simple, suffer from the lack of membranes which are either not selective enough or not very permeable to CO₂. This results in a low-purity CO₂ product. Recent developments have shown pressure/vacuum swing adsorption to be a promising option for separating CO₂. PSA/VSA operate at ambient temperatures and do not require any solvent or thermal energy

for CO₂ recovery or sorbent regeneration. Only feed compression and vacuum generation constitute the key energy requirements which can be quite low. Moreover, sorbents can be designed which can withstand H₂O, O₂, SO_x, and NO_x in the flue gas stream.

PSA processes have been widely applied for the removal of CO₂ from various feed mixtures, such as CO₂ in the steam reformer off-gas, natural gas and flue gas mixtures [173]. They are also commercially used to remove trace amounts of CO₂ from air [113]. In these commercial PSA cycles, the weakly adsorbed (or light) component in the mixture is the desired product and enriching the strongly adsorbed (or heavy) component (in this case, CO₂) is not a concern. On the other hand, for CO₂ sequestration, it is necessary to concentrate CO₂ to a high purity to reduce the compression and the transportation cost. Moreover, safety and environmental issues are additional reasons for concentrating CO₂ to a high purity.

Typically adsorbents preferentially adsorb CO₂ from a flue gas mixture, consequently making it a heavy product. The conventional PSA cycles are inappropriate for concentrating heavy product because the light product purge step (or the light reflux step) in these cycles uses a portion of the light product gas, which necessarily dilutes the heavy component in the heavy product stream. As a result, a pure light component is easy to attain from such cycles, but not a pure heavy component. Thus, it is necessary to develop PSA processes specifically targeted to obtain pure strongly adsorbed component, CO₂ in this case.

Because the product purity of the heavy component is limited by the gas mixture occupying the void spaces in the bed, its purity can be increased by displacing the gas mixture in the void spaces with a pure heavy product gas. For instance, for the separation of N₂-CO₂ mixture, the displacement can be accomplished by purging the bed with CO₂ after the adsorption step in the PSA cycle. Hence, to obtain a pure heavy product gas, a heavy product pressurization step or a heavy reflux step is necessary in the cycle, similar to a light product pressurization step or a light reflux step in the conventional PSA cycles. This idea was first suggested in a patent by Tamura [183], and has been incorporated in most of the PSA cycles that have been suggested in the literature for high purity CO₂ separation from flue gas.

A fairly comprehensive review of the previous studies on PSA cycles for concentrating CO₂ from flue gas is presented in the next section. This review highlights the difficulties associated with choosing one PSA cycle over another for a given application. From these studies it is not clear why a particular cycle was chosen or one performed better than other configurations. Hence, we apply more structured superstructure approach to develop optimal PSA configurations to separate CO₂ from a flue gas mixture.

4.2 Literature Review

Since 1992, when the Japanese power industry started investigating flue gas CO₂ removal using gas adsorption [90, 95, 96, 158, 209], a multitude of PSA/VSA cycles have been developed in the literature to produce pure CO₂ from a flue gas mixture. We provide a summary of these studies in Table 4.1. In this table, y_f is the CO₂ % in feed, while p_{CO_2} and r_{CO_2} are CO₂ purity and recovery in the heavy product stream, respectively, and P_l is the vacuum/low pressure used to extract CO₂ at high purity. The terminology for various operating steps in a cycle is adopted from Reynolds et al. [151]. Most of these studies are bench-scale and deal with extremely small feed throughput.

Ritter and co-workers have studied numerous PSA cycles for CO₂ capture from a feed at high temperature using K-promoted Hydrotalcite as the adsorbent [151, 149, 150]. They have emphasized the importance of including heavy reflux step to obtain heavy product at a high purity. They compared seven different 4-bed 4-step, 4-bed 5-step and 5-bed 5-step cycle configurations with and without heavy reflux step. In another work [148], they analyzed nine different PSA configurations and achieved better purities and recoveries for CO₂, although, at an extremely small feed throughput. Kikkinides et al. [107] studied a 4-bed 4-step vacuum swing process and improved CO₂ purity and recovery by allowing significant breakthrough of CO₂ from the light end of the column undergoing heavy reflux, and then recycling the effluent from this light end back to the column with the feed. Chue et al. [52] compared activated carbon and zeolite 13X using a 3-bed 9-step VSA process. They suggested that despite a

Table 4.1: PSA cycles suggested in the literature for post-combustion CO₂ separation

PSA cycle configuration	operating step sequence ^a	Ads ^b	y_f (%)	p_{CO_2} (%)	r_{CO_2} (%)	P_l (kPa)	Feed throughput (kgmol/hr)	Ref.
5-bed 5-step	F,HR,CnD,LR,LPP	HTlc	15	72	82	11.49	0.001	[151] ^d
5-bed 5-step	F,HR,CnD,LR,LPP	HTlc	15	76	49	11.49	0.003	[151] ^d
4-bed 4-step	F,HR,CnD,LPP	HTlc	15	83	17	11.49	0.001	[151] ^d
5-bed 5-step	F,HR,CnD,LR,LPP	HTlc	15	98.7	98.7	11.64	0.00052	[148] ^d
5-bed 5-step	F+R,HR,CnD,LR,LPP	HTlc	15	98.6	91.8	11.64	0.00052	[148] ^d
4-bed 4-step	F,HR,CnD,LPP	HTlc	15	99.2	15.2	11.64	0.006	[148] ^d
4-bed 4-step	F+R,HR,CnD,LPP	HTlc	15	99.2	15.2	11.64	0.006	[148] ^d
4-bed 4-step	LPP,F+R,HR,CnD	AC	17	99.9	68	10.13	16.19	[107] ^d
3-bed 8-step	FP,F,CoD,R,N,HR,CnD,N	13X	16	99	45	6.67	0.049	[52]
4-bed 8-step	FP,F,HR,LEE,CnD,LR,LEE,N	NaX	13	95	50	10	1.116	[182] ^{c,d}
2-bed 4-step	FP,F,CnD,LR	13X	10	70	68	4	0.331	[142] ^d
2-bed 6-step	LEE,FP,F,LEE,CnD,LR	13X	10	82	57	6.67	0.331	[142] ^d
3-bed 5-step	FP,F,HR,CnD,LR	13X	10	83	54	6.67	0.331	[142] ^d
2-bed 4-step	FP,F,CnD,LR	13X	8.3	78	50	101.3	0.004	[85] ^c
3-bed 8-step	FP,F,CoD,LEE,HPP,HR,CnD,LEE	AC	17	99.8	34	10.13	0.027	[132] ^{c,d}
3-bed 7-step	FP,F,LEE,HR,N,CnD,LEE	AC	13	99	55	10.13	0.204	[133] ^{c,d}
3-bed 8-step	FP,F,CoD,LEE,HPP,HR,CnD,LEE	13X	13	99.5	69	5.07	0.025	[50] ^{c,d}
2-bed 4-step	HPP,FP,CoD,CnD	13X	20	48	94	5.07	—	[51]
2-bed 5-step	HPP,FP,F,CoD,CnD	13X	20	43	88	5.07	—	[51]
3-bed 4-step	LPP,F,CnD,LR	13X	20	58	75	5.07	—	[51]
3-bed 6-step	LPP,FP,F,HR,CoD,CnD	13X	20	63	70	5.07	—	[51]
2-bed 4-step	FP,F,CnD,LR	13X	15	72	94	90	30.35	[111]
1-bed 4-step	FP,F,CoD,CnD	13X	15	90	94	70	1.741	[111]
2-bed 4-step	LPP,F,CnD,LR	13X	15	52	66	10	0.007	[86] ^c
3-bed 5-step	LPP,F,HR,CnD,LR	13X	15	83	66	10	48.57	[86] ^c
3-bed 6-step	F,LEE,CnD,LEE	13X	12	83	60	4	0.193	[44] ^{c,d}
3-bed 9-step	F,LEE,HR,CnD,LEE	13X	12	95	60	5	0.193	[212] ^{c,d}
3-bed 9-step	F,LEE,I,LEE,CnD,LEE,FP	13X	12	92.5	75	3	0.327	[203] ^d

^aCycle-step legend: CnD-counter-current depressurization; CoD-co-current depressurization; FP-feed pressurization; F-feed or adsorption; HPP-heavy product pressurization; HR-heavy reflux; LEE-light end equalization; LPP-light product pressurization; LR-light reflux; N-null or idle; R-recycle. ^bAdsorbent legend: HTlc-K-promoted Hydrotalcite; NaX, 13X-molecular sieve zeolites; AC-activated carbon. ^cStudies with experimental results. ^dMulticomponent study.

high heat of adsorption of CO₂, zeolite 13X is better because of its higher working capacity, lower purge requirement, and higher equilibrium selectivity. PSA cycle sequences that took advantage of both light and heavy reflux steps were explored by Takamura et al. [182] and Park et al. [142]. Takamura et al. studied a 4-bed 8-step VSA process while Park et al. analyzed three different cycle configurations for VSA processes. While the pure CO₂ rinse step and the equalization step in the 3-bed 5-step cycle improved their CO₂ purity and recovery, it didn't decrease their power requirements, which were 106.91 kWh/tonne CO₂ for the 2-bed 6-step cycle, and 147.64 kWh/tonne for the 3-bed 5-step cycle. Though the power consumption was quite low, the feed throughput of 0.331 kgmol/hr was also on the lower side. The conventional 2-bed 4-step Skarstrom cycle was also studied by Gomes et al. [85], in which they didn't apply vacuum to recover CO₂. Their work also showed that the light reflux step itself is not sufficient to obtain pure heavy component.

Na et al. [132, 133] and Choi et al. [50] studied 3-bed 8-step and 3-bed 7-step VSA configurations experimentally as well as numerically. Light reflux step was not used for any of these configurations, while heavy reflux was used in all of them. The 2-bed cycles of Chou and Chen [51] did not use any kind of reflux steps while the 3-bed cycles used both light and heavy reflux steps. The 2-bed cycles were unconventional as flow reversal was implemented in between the pressurization and depressurization steps. Similarly, the 3-bed 6-step cycle incorporated an unusual co-current light product pressurization step. They couldn't go beyond 63% CO₂ purity, which was achieved using the 3-bed 6-step cycle. Ko et al. [111] optimized a 2-bed 4-step PSA process to minimize power consumption, and a 1-bed 4-step fractionated VPSA process to increase CO₂ purity to 90% and recovery to 94%. Grande et al. [86] studied a classical Skarstrom cycle with light product pressurization and a 3-bed 5-step process which included a pure CO₂ rinse step after the adsorption step. Their scale-up study showed that a purity of 83% and a recovery of 66% is possible with the 3-bed 5-step process at a much higher feed throughput of 48.57 kgmol/hr.

Webley and co-workers [44, 212, 203, 211] have done an extensive research in the field of CO₂ separation by adsorption. Chaffee et al. [44] and Zhang et al. [212] studied two different VSA processes. For a low feed throughput of 0.193 kgmol/hr for both the cycles, they achieved a low power consumption of 192 kWh/tonne CO₂ for the 3-bed 6-step and 240 kWh/tonne CO₂ for the 3-bed 9-step cycle. Xiao et al. [203] studied a similar 3-bed 9-step cycle and were able to increase CO₂ recovery to 75%. In another study, Zhang and Webley [211] compared numerous VSA cycle configurations, and showed that CO₂ purity can be increased by particularly incorporating heavy reflux and equalization steps.

While this review offers some trends and guidelines, a fully systematic methodology is still required to design PSA cycle configurations. In the subsequent sections, we demonstrate application of the superstructure approach to obtain optimal PSA cycles for post-combustion carbon capture.

Table 4.2: Zeolite 13X properties and model parameters [111]

Parameter	Value	
Bulk porosity (ϵ_b)	0.34	
Particle diameter (d_p)	0.002 m	
Adsorbent density (ρ_s)	1870 kg m ⁻³	
Bulk density (ρ_b)	1234.2 kg m ⁻³	
Heat capacity of solid (C_{ps})	450.54 J kg ⁻¹ K ⁻¹	
Heat transfer coefficient (U_A)	926.7 J m ⁻³ sec ⁻¹ K ⁻¹	
Gas viscosity (μ)	1.7857×10 ⁻⁵ kg m ⁻¹ sec ⁻¹	
Gas constant (R)	8.314 J mol ⁻¹ K ⁻¹	
Mass transfer coefficient (k)	CO ₂ =0.1631 sec ⁻¹ N ₂ =0.2044 sec ⁻¹	
Heat of adsorption (ΔH^{ads})	CO ₂ =23011.14 J mole ⁻¹ N ₂ =14452.72 J mole ⁻¹	
Ambient temperature (T_w)	298 K	
Isotherm parameters		
	CO ₂	N ₂
k_1^1	2.817269	1.889581
k_1^2	-3.51×10 ⁻⁴	-2.25×10 ⁻⁴
k_1^3	2.83×10 ⁻⁹	1.16×10 ⁻⁹
k_1^4	2598.203	1944.606
k_2^1	3.970888	1.889581
k_2^2	-4.95×10 ⁻³	-2.25×10 ⁻⁴
k_2^3	4.41×10 ⁻⁹	1.16×10 ⁻⁹
k_2^4	3594.071	1944.606

4.3 Case Setup

We consider an 85%-15% N₂-CO₂ feed mixture which is a typical composition of a post-combustion flue gas stream. As an initial study, the focus is on a binary feed mixture. A multicomponent feed mixture also having water, oxygen and other trace components will be considered in the future extensions of this work. We assume that the flue gas enters at atmospheric pressure at a temperature of 310 K, and a maximum velocity (v_{feed}) of 50 cm/sec. Since the inlet pressure P_{inlet} is atmospheric, we assume optional inlet compressor is present in the superstructure which compresses inlet gas to pressure P_{feed} . Feed pressure P_{feed} varies with the case studies. Zeolite 13X is chosen as the adsorbent to separate CO₂; Chue et al. [52] suggested it to be a preferable adsorbent over others for this separation system. The adsorbent

properties for 13X and other model parameters are listed in Table 4.2 [111].

Usually a large number of spatially discretized nodes are required to capture steep adsorption fronts. Such fine spatial discretization, together with temporal discretization, leads to a very large set of algebraic equations which becomes extremely expensive to solve. Although a large number of elements improve accuracy, it makes the problem computationally challenging to solve. Hence, to get the solution in a reasonable amount of time, we consider 20 spatial finite volumes and around 24-26 temporal finite elements for the optimization problem. NLP solution from IPOPT is verified with more accurate dynamic simulations in MATLAB at the optimum, as mentioned in section 3.4.

We consider three different cases to explore different facets of the superstructure approach. The first case study optimizes the 2-bed 4-step Skarstrom configuration, obtained after fixing the control variables in the superstructure, and shows the ineffectiveness of such traditional cycles for high-purity CO₂ separation. The second case then finds an optimal PSA configuration which separates CO₂ at high purity and recovery. Finally, in the third case, we find an optimal configuration which achieves high-purity separation with minimal power requirements.

4.4 Case Studies and Computational Results

4.4.1 Case I: Optimization with a conventional configuration

First, we explore the potential of the conventional 2-bed 4-step Skarstrom cycle (*cf.* section 2.4.1) for post-combustion CO₂ capture. For this, we fix the profiles of $\alpha(t)$, $\beta(t)$ and $\phi(t)$ over time, as shown in the Figure 3.3. While f_β is chosen as 0.3, f_ϕ before t_{switch} is fixed to 0.35. This ensures that the superficial velocity is close to zero towards the light end of CoB during the FP step. In this case, P_{ads} and P_{des} remain constant for the entire cycle. The inlet pressure P_{feed} is fixed to 300 kPa, as considered by Gomes et al. [85] as well.

With this configuration we maximize CO₂ recovery. Since the lack of any heavy reflux step in the configuration may not enable a high purity separation, a relatively low value of 40%

is chosen for the lower bound on CO₂ purity. Besides P_{ads} and P_{des} , we also consider bed length $BLen$, and cycle time T_c as decision variables. Since a moving finite element strategy is adopted, the length of each finite element is also considered as an optimization variable. Because none of the decision variables are functions of time, the optimal control problem (3.1) becomes a dynamic optimization problem, which becomes the following NLP after discretizing PDAEs in both space and time.

$$\begin{aligned}
\max \quad & \text{CO}_2 \text{ recovery (from Equation (3.14d))} \\
\text{s.t.} \quad & c(w) = 0 \text{ (fully discretized Equations (3.2)-(3.13))} & (4.1a) \\
& \text{CO}_2 \text{ purity} \geq 0.4 \text{ (from Equation (3.14b))} & (4.1b) \\
& P_{ads} \geq P_d & (4.1c) \\
& P_{des} \leq P_{feed} & (4.1d) \\
& P_a \geq P_{feed} & (4.1e) \\
& 1 \text{ m} \leq BLen \leq 6 \text{ m} & (4.1f) \\
& 20 \text{ sec} \leq T_c \leq 2400 \text{ sec} & (4.1g) \\
& 101.32 \text{ kPa} \leq P_{ads} \leq 1000 \text{ kPa} & (4.1h) \\
& P_{des} \geq 10 \text{ kPa} & (4.1i)
\end{aligned}$$

Here w and $c(w) = 0$ represent the set of completely discretized variables and model equations, respectively. Constraint (4.1c) ensures that the pressure always drops around the pressure reducing valve in the superstructure. Similarly, constraints (4.1d) and (4.1e) ensure that the gas is never expanded by the heavy gas and the feed compressors, respectively. The rest of the inequalities are bounds on the decision variables.

With 24 temporal finite elements and 20 spatial finite volumes, we solved the NLP in AMPL [78] using IPOPT. Table 4.3 includes a summary of the optimization results. With 35,022 variables and 29 degrees of freedom, we were able to solve it to optimality in around 3 CPU hours on an Intel Quad core 2.4 GHz machine with 8 GB RAM. Optimal moving finite element lengths and cycle time of 2140 sec. yield an optimal step time of 760 sec. for the pressurization (and depressurization) step, and 410 sec. for the feed (and light reflux) step. Such a long pressurization step is due to a small amount of feed during that step, which requires longer time for bed to get pressurized and CO₂ to adsorb. At the optimum, the cycle

Table 4.3: Optimization results for Case I

No. of variables	35022	
No. of equations	34993	
CPU time	176.94 min.	
Optimal step times		
Step 1 (and 3)	760 sec	
Step 2 (and 4)	410 sec	
Optimal parameters	$BLen=5.51$ m	$T_c=2140$ sec
	$P_{ads}=276.43$ kPa	$P_{des}=21.75$ kPa
	Accuracy check	
	Full discretization	MATLAB verification
N ₂ purity	91.25%	90.74%
N ₂ recovery	85.88%	85.94%
CO ₂ purity	40%	38.65%
CO ₂ recovery	53.36%	50.22%

handles a feed flux of $96.4 \text{ kgmol m}^{-2} \text{ hr}^{-1}$, which is higher than the corresponding 2-bed 4-step case studies in Table 4.1. At a purity of 40%, a maximum CO₂ recovery of only 53.4% was achieved. Such poor performance proves the point made in the introduction; classical cycles without heavy reflux cannot produce heavy product at high purity since a light reflux step dilutes the heavy product and decreases its purity. Table 4.3 also lists the MATLAB verification of the AMPL results. We considered 20 spatial finite volumes for MATLAB as well. A comparison of the purities and the recoveries indicates reasonable accuracy for the complete discretization approach.

4.4.2 Case II: Cycle synthesis to maximize CO₂ recovery

Since a high-purity CO₂ separation wasn't achieved by the Skarstrom cycle, in this case we solve the optimal control problem (3.1) to obtain an optimal configuration which yields better performance. For this, a few modifications are essential to the optimization problem presented in the previous case. The control variables $\alpha(t)$, $\beta(t)$ and $\phi(t)$ are freed to let them achieve an optimal sequence of operating steps. The pressures P_{ads} and P_{des} are converted back to time dependent control variables. To keep this case comparable to the previous one, we fix

the bed length to 5 meters. A desired CO₂ purity of at least 95% is chosen. Besides this, we impose a lower bound on feed flux Q_{feed} , in the absence of which the optimizer may force the feed fraction $\phi(t)$ to zero in order to maximize CO₂ recovery. Finally, we add the equation for power consumption (3.15b) to the optimal control problem. A 72% efficiency is assumed for all the compressors and the vacuum generator unit [30]. As in the previous case, we fix the flue gas inlet pressure P_{feed} to 300 kPa to achieve a reasonable Q_{feed} . The following large-scale NLP results after complete discretization of state and control variables in the optimal control problem.

$$\begin{aligned}
 \max \quad & \text{CO}_2 \text{ recovery (from Equation (3.14d))} \\
 \text{s.t.} \quad & c(w) = 0 \text{ (fully discretized Equations (3.2)-(3.13))} \\
 & \text{CO}_2 \text{ purity} \geq 0.95 \text{ (from Equation (3.14b))} \\
 & Q_{feed,L} + Q_{feed,H} \geq 80 \text{ kgmol m}^{-2}\text{hr}^{-1} \\
 & P_{ads} \geq P_d \\
 & P_{des} \leq P_{feed} \\
 & P_a \geq P_{feed} \\
 & 0 \leq \alpha(t_i), \beta(t_i), \phi(t_i) \leq 1 \quad \forall t_i \\
 & 20 \text{ sec} \leq T_c \leq 2400 \text{ sec} \\
 & 101.32 \text{ kPa} \leq P_{ads}(t_i) \leq 600 \text{ kPa} \quad \forall t_i \\
 & P_{des}(t_i) \geq 50 \text{ kPa} \quad \forall t_i
 \end{aligned} \tag{4.2}$$

As in the previous case study, $c(w)$ is the set of completely discretized PDAEs with CSS condition. We choose a lower bound of 50 kPa for the vacuum generated, which is not a substantially high vacuum. Similarly, the chosen upper bound of 600 kPa for P_{ads} is also not substantially high. No bounds are specified for the purity and the recovery of nitrogen. We impose a lower bound of 80 kgmol m⁻² hr⁻¹ on the total feed flux. Because the bound is not on the feed throughput, a bigger diameter PSA bed will be able to handle much higher feed throughput and the optimal configuration need not change. For instance, for a 3 meter bed diameter, one PSA column will be able to handle a significantly high feed throughput of 565 kgmol/hr for the same optimal configuration. Also note that the value of 80 kgmol m⁻² hr⁻¹ is significantly higher than feed flux chosen in the literature studies in Table 4.1, as the focus here is to synthesize industrial scale PSA systems.

The NLP was solved in AMPL with 26 temporal finite elements and 20 spatial finite

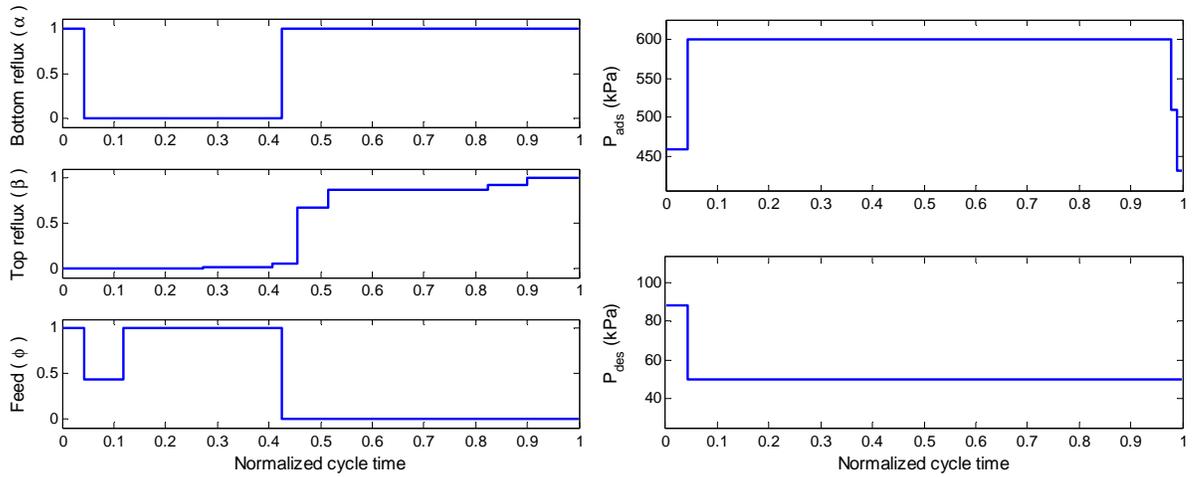


Figure 4.1: Optimal control profiles for case II

volumes. The optimal control profiles are shown in Figure 4.1. The profiles are drawn against cycle time normalized between 0 and 1. These profiles suggest an optimal 2-bed 6-step VSA process, illustrated in Figure 4.2.

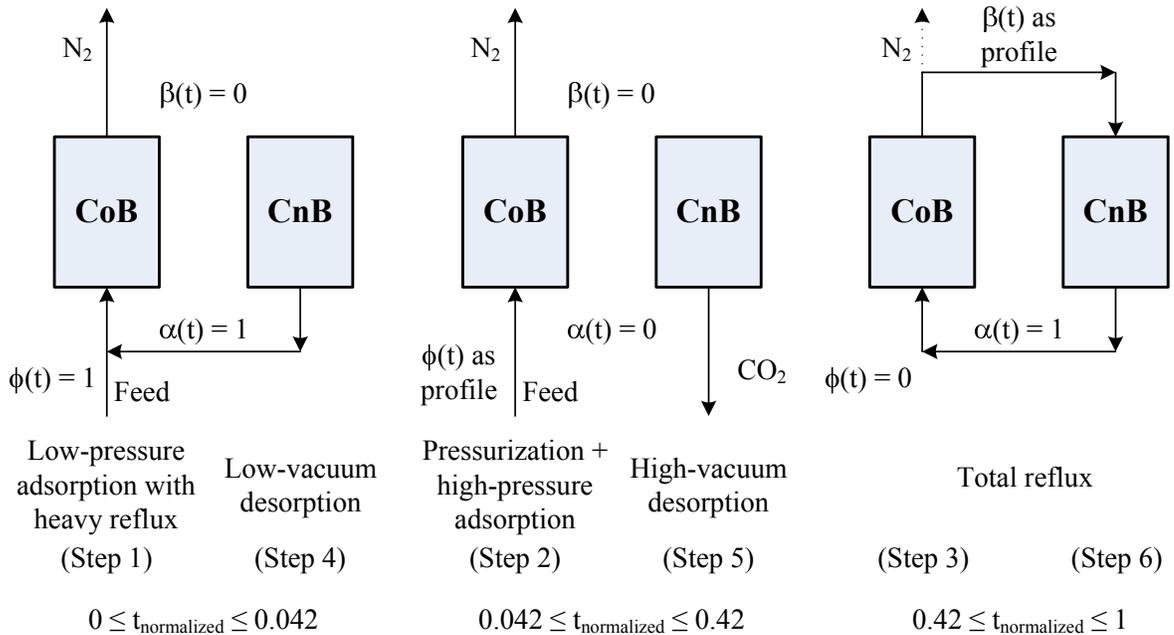


Figure 4.2: Optimal VSA configuration for case II

The cycle starts with $\alpha(t)=1$, $\beta(t)=0$, and $\phi(t)=1$. This suggests a bottom reflux from CnB to CoB and feed being fed to CoB. From the profiles of $P_{ads}(t)$ and $P_{des}(t)$, CoB operates at

around 450 kPa while CnB operates at around 85 kPa during this step. Thus, we have a low-pressure adsorption step with a heavy reflux for CoB (step 1) and a low-vacuum desorption step for CnB (step 4) with desorbed CO₂ being sent as a heavy reflux from CnB to CoB. After this step, both top and bottom reflux disappear, while $\phi(t)$ indicates continuation of the feed to CoB. CoB gets pressurized to the upper bound of 600 kPa and N₂ is withdrawn at a high pressure, while CO₂ is extracted from CnB at a vacuum of 50 kPa. This suggests a pressurization and high pressure adsorption step for CoB (step 2) and a high vacuum desorption step for CnB (step 5). The feed fraction $\phi(t)$ drops at the beginning of this step to facilitate CoB pressurization. We observe a drop in the CO₂ concentration in CoB (see step 2 in Figure 4.3) because of its low concentration in feed. Also, because of the application of vacuum, the gas-phase CO₂ concentration decreases sharply for both step 4 and 5, as evident in Figure 4.3.

Further, the pressures in the beds are held at their same respective levels, while $\alpha(t)$ becomes 1, $\beta(t)$ approaches 1, and the feed is stopped completely. Because there is no feed nor product at this time, we have a total reflux step (step 3 and 6), in which both the beds are connected to each other and a recirculation of the components occurs within the system. A small amount of N₂ is withdrawn at the beginning of this step, and is shown as a dotted line in Figure 4.2. A decrease in $P_{ads}(t)$ and an increase in $P_{des}(t)$ towards the end of this step halts this recirculation. After the total reflux step, the co-current bed follows the steps of the counter-current bed and vice-versa. This completes the cycle.

From Figure 4.2 together with Figure 4.3, we observe a couple of key aspects of this cycle. First, we observe an extensive use of heavy reflux in the cycle (step 1 and 3) to enrich gas-phase CO₂ concentration towards heavy end of CoB to ensure high-purity CO₂ production. During both steps 1 and 3, desorbed CO₂ from CnB is sent as a heavy reflux to CoB which enriches the adsorbed CO₂ concentration towards the heavy end of CoB. This is evident from the gas-phase CO₂ concentration profile for steps 1 and 3 in Figure 4.3. Second, we observe a completely novel total reflux step in the cycle. Moreover, this is the longest step and runs for almost 60% of the total cycle duration. This is a vital step to improve CO₂ purity as it eschews

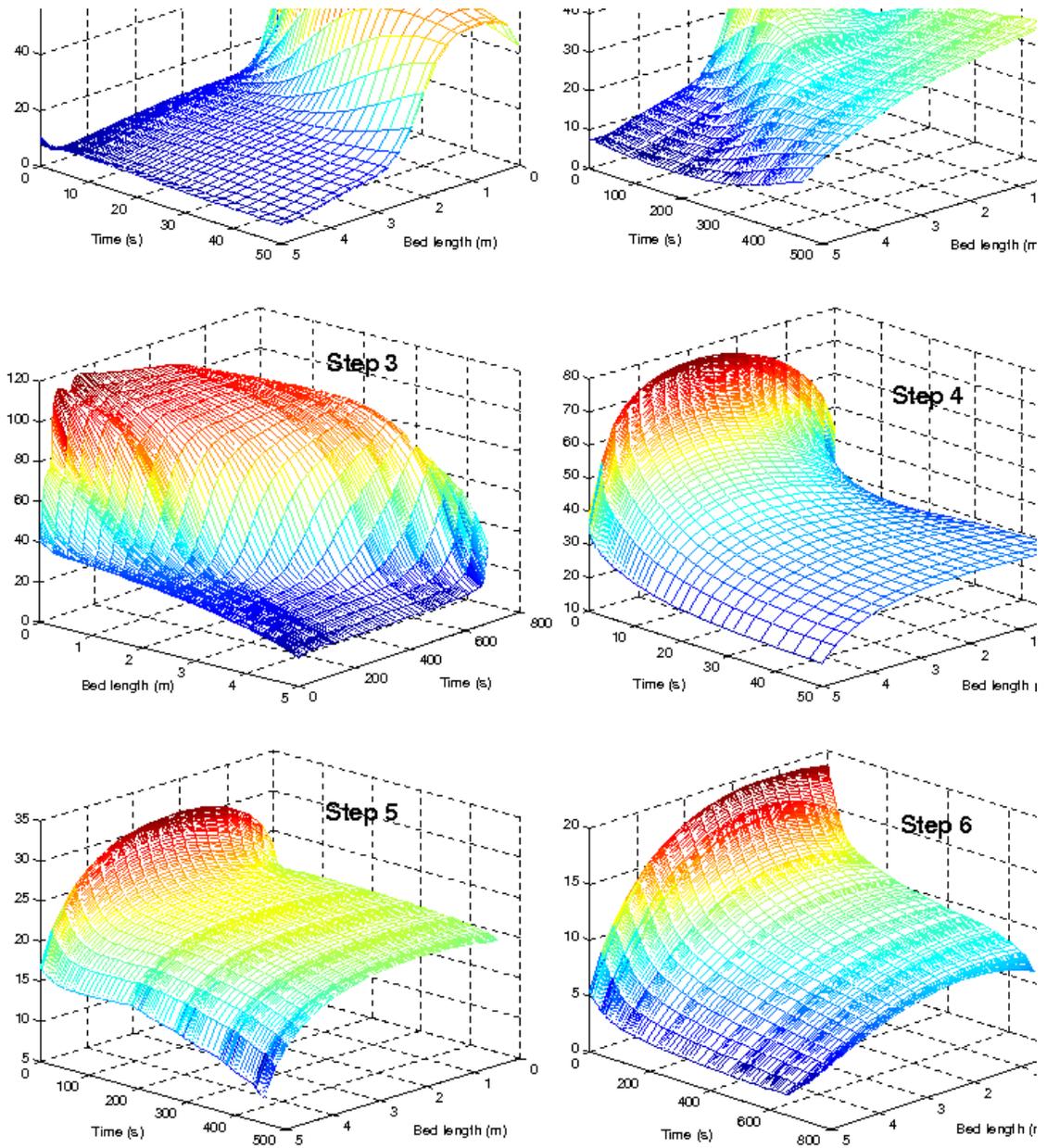
Figure 4.3: Gas-phase CO₂ concentration profiles for case II

Table 4.4: Optimization results for case II

No. of variables	50162
No. of equations	49956
CPU time	756.22 min.
Optimal step times	
Step 1 (and 4)	50 sec
Step 2 (and 5)	405 sec
Step 3 (and 6)	685 sec
Optimal cycle time	2280 sec
Feed flux	80 kgmol m ⁻² hr ⁻¹
Power consumption	637.25 kWh/tonne CO ₂ captured
CO ₂ purity	95%
CO ₂ recovery	80.09%

external influence and re-arranges component distribution within the system. During steps 3 and 6, nitrogen from CoB to CnB purges CO₂ out of CnB from its heavy end and enriches itself towards the light end of CnB while pushing its front. Similarly CO₂ from CnB to CoB purges nitrogen out of CoB and enriches itself towards the heavy end of CoB and pushes its front towards the light end of CoB. Step 3 in Figure 4.3 confirms such a movement of CO₂ front.

The optimization results for this case are summarized in Table 4.4. With 50,162 variables and 206 degrees of freedom, it was solved to optimality in approximately 12.5 CPU hours on the Intel Quad core 2.4 GHz machine with 8 GB RAM. At the optimum, the feed flux attained its lower bound of 80 kgmol m⁻² hr⁻¹. For such a high feed flux, we obtained a reasonable power consumption of 637.25 kWh/tonne CO₂ captured. Also, an optimum CO₂ recovery of 80% at a purity of 95%, for such a high feed flux, is substantially better than the literature studies for post-combustion capture that deal with high feed throughput. These results confirm our assertion that steps like heavy reflux are essential for high-purity CO₂ separation.

In Table 4.5 we provide a validation of the optimal results obtained from AMPL with method of lines simulations in MATLAB for varying number of spatial finite volumes. The results from AMPL are in good agreement with those from MATLAB, and the accuracy doesn't suffer as we consider a large number of finite volumes in MATLAB. This indicates that the

Table 4.5: Accuracy validation for AMPL results of case II

Spatial finite volumes	AMPL results		MATLAB verification	
	20	20	40	80
N ₂ purity	96.58%	96.28%	96.19%	96.16%
N ₂ recovery	99.26%	99.26%	99.28%	99.29%
CO ₂ purity	95%	94.95%	95.04%	95.07%
CO ₂ recovery	80.09%	78.26%	77.70%	77.51%

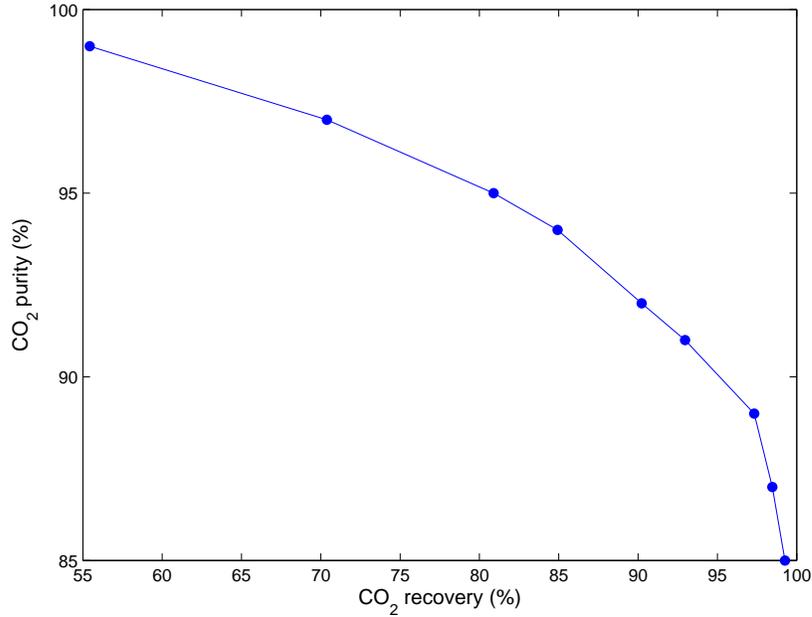


Figure 4.4: Purity-recovery trade-off curve for case II

complete discretization approach is reasonably accurate.

Figure 4.4 shows a trade-off curve between CO₂ purity and recovery. We construct this curve by varying the lower bound on CO₂ purity in the NLP (4.2), and then maximizing CO₂ recovery for each lower bound. As a result, it is possible that a different optimal cycle configuration is achieved at each point plotted on the curve. However, each configuration is the best possible cycle for a particular CO₂ purity. Consequently, this yields an optimal purity-recovery trade-off curve. The curve shows that if a very high purity CO₂ separation is desired then the recovery falls drastically. A similar trend is observed with the purity when a

very high CO₂ recovery is sought. The intermediate section of the curve is a preferable region to operate.

4.4.3 Case III: Cycle synthesis to minimize power consumption

Although we achieved a high purity separation in the previous case, the power consumption was also quite high. Therefore, the objective of this case is to obtain an optimal configuration which yields a high-purity separation at minimal power requirements. To achieve this, few minute modifications are done to the NLP (4.2). While the lower bound on CO₂ recovery is relaxed to 85%, the lower bounds on CO₂ purity and feed flux are relaxed to 90% and 65 kgmol m⁻²hr⁻¹, respectively. To minimize the work done in compressing flue gas from P_{inlet} to P_{feed} (in Equation (3.15a)), we consider P_{feed} a decision variable instead of fixing it to 300 kPa. Appropriate bounds are imposed on P_{feed} . The efficiency is kept same as 72% for all compressors and vacuum generator. The rest of the optimization problem remains same, and is as below. The NLP was solved in AMPL with 24 temporal finite elements and 20 spatial finite volumes. The optimal control profiles are shown in Figure 4.5. Since optimal feeding strategy and pressure profile are different compared to previous case, we obtain an entirely different 2-bed 8-step VSA configuration, illustrated in Figure 4.6.

$$\begin{aligned}
 \min \quad & \text{Power (from Equation (3.15b))} \\
 \text{s.t.} \quad & c(w) = 0 \text{ (fully discretized Equations (3.2)-(3.13))} \\
 & \text{CO}_2 \text{ purity} \geq 0.9 \text{ (from Equation (3.14b))} \\
 & \text{CO}_2 \text{ recovery} \geq 0.85 \text{ (from Equation (3.14d))} \\
 & Q_{feed,L} + Q_{feed,H} \geq 65 \text{ kgmol m}^{-2}\text{hr}^{-1} \\
 & P_{ads} \geq P_d \\
 & P_{des} \leq P_{feed} \\
 & P_a \geq P_{feed} \\
 & 0 \leq \alpha(t_i), \beta(t_i), \phi(t_i) \leq 1 \quad \forall t_i \\
 & 20 \text{ sec} \leq T_c \leq 2400 \text{ sec} \\
 & 101.32 \text{ kPa} \leq P_{feed} \leq 600 \text{ kPa} \\
 & 101.32 \text{ kPa} \leq P_{ads}(t_i) \leq 600 \text{ kPa} \quad \forall t_i \\
 & P_{des}(t_i) \geq 50 \text{ kPa} \quad \forall t_i
 \end{aligned} \tag{4.3}$$

The cycle begins with $\alpha(t)=1$, $\beta(t)=0$ and $\phi(t)$ close to one. This suggests a heavy reflux from CnB to CoB and feed being fed to CoB. From the profiles of $P_{ads}(t)$ and $P_{des}(t)$, the

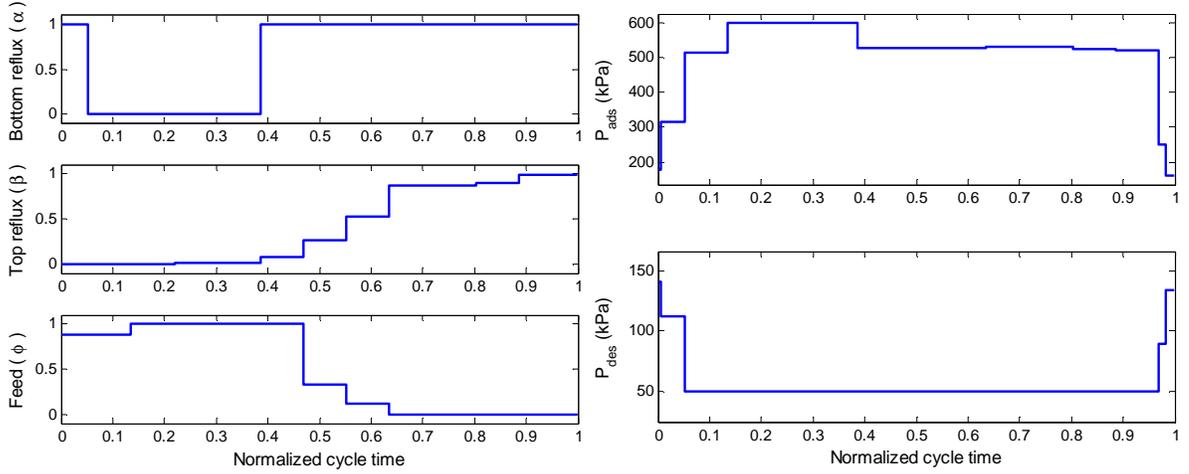


Figure 4.5: Optimal control profiles for case III

pressure rises in CoB and falls in CnB during this step. Thus, we have a pressurization step for CoB (step 1) and a depressurization step for CnB (step 5), with heavy reflux increasing the adsorbed-phase CO_2 concentration towards the heavy end of CoB.

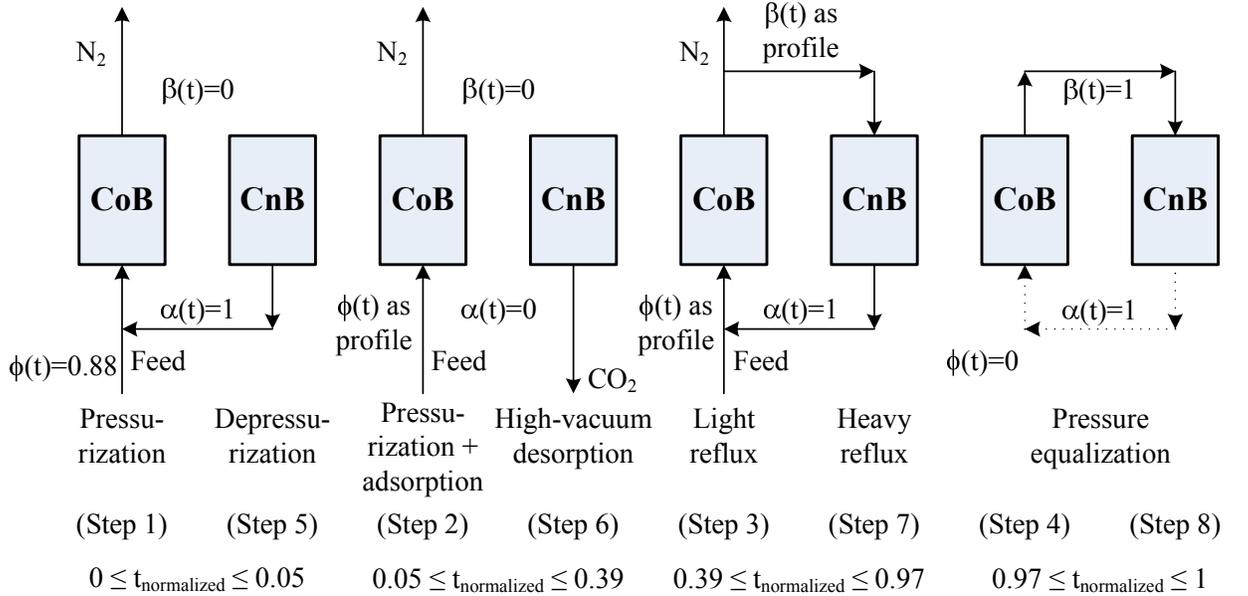


Figure 4.6: Optimal VSA configuration for case III

Next, both $\alpha(t)$ and $\beta(t)$ go to zero, while $P_{ads}(t)$ and $P_{des}(t)$ attain their maximum and minimum allowed values, respectively. This suggests an adsorption step with the removal of light product for CoB (step 2), and a high vacuum desorption step for CnB (step 6), during

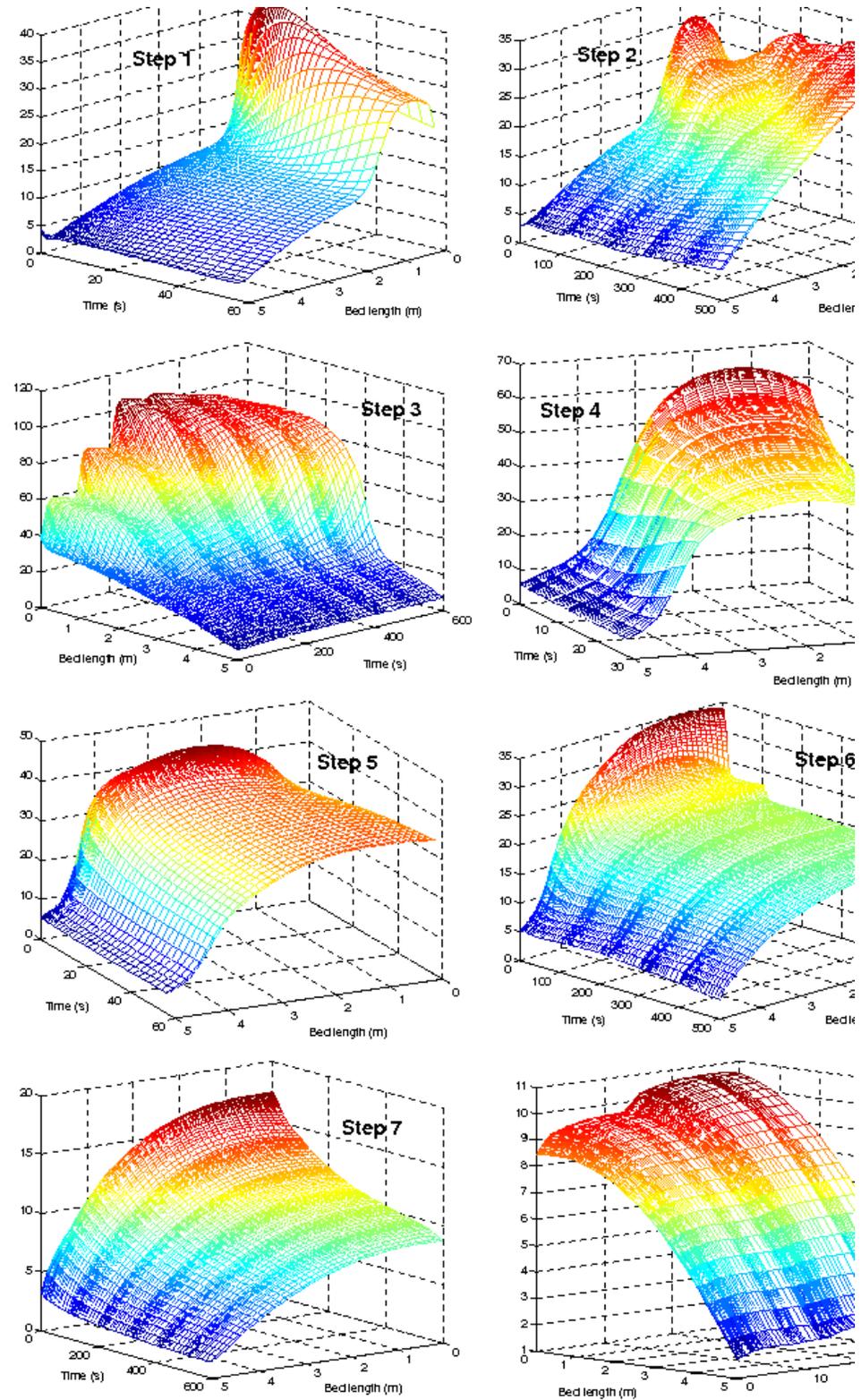
Figure 4.7: Gas-phase CO₂ concentration profiles for case III

Table 4.6: Optimization results for case III

No. of variables	46313
No. of equations	46122
CPU time	273.08 min.
Optimal step times	
Step 1 (and 5)	56.77 sec
Step 2 (and 6)	500.03 sec
Step 3 (and 7)	614.79 sec
Step 4 (and 8)	28.41 sec
Optimal cycle time	2400 sec
Optimal P_{feed}	182.31 kPa
Feed flux	65 kgmol m ⁻² hr ⁻¹
Power consumption	464.76 kWh/tonne CO ₂ captured
CO ₂ purity	90%
CO ₂ recovery	85%

which high purity CO₂ is collected. After this step we observe that both $\alpha(t)$ and $\beta(t)$ go to 1. However, unlike the total reflux step in previous case study, $\beta(t)$ doesn't go to 1 at once and nitrogen is still constantly removed from the system. Feed is also fed to CoB for a considerable amount of time at the beginning of this step. Therefore, this translates into a heavy reflux step for CoB (step 3) and a light reflux step for CnB (step 7). Nevertheless, the intent of this step is similar to that of the total reflux step: enrich the N₂ front towards the light end of CnB, and CO₂ front towards the heavy end of CoB. The gas phase CO₂ concentration profiles for both these steps in Figure 4.7 validate this behavior. After this, $\alpha(t)$ and $\beta(t)$ remain at 1, while $P_{ads}(t)$ starts to drop, $P_{des}(t)$ starts to jump sharply, and the two pressures come very close to each other. In fact, $P_{ads}(t)$ and $P_d(t)$ are approximately equal during this step. This translates into a short pressure equalization step (step 4 and 8). Since the heavy reflux from CnB is negligible, we show it as a dotted line for this step in Figure 4.6. Clearly, from Figure 4.7, CO₂ concentration drops substantially in CoB and rises steadily in CnB during the equalization step. After this, CoB follows the steps of CnB and vice-versa.

Figure 4.6 together with the gas-phase CO₂ concentration profiles in Figure 4.7 illustrate several key aspects of the cycle. First, as in the previous case, heavy reflux step is used as

Table 4.7: Accuracy validation for AMPL results of case III

Spatial finite volumes	AMPL results		MATLAB verification	
	20	20	40	80
N ₂ purity	97.38%	97.16%	97.03%	96.99%
N ₂ recovery	98.33%	98.36%	98.40%	98.42%
CO ₂ purity	90%	90.01%	90.16%	90.22%
CO ₂ recovery	85%	83.72%	82.93%	82.68%

the only step to enrich adsorbed-phase CO₂ concentration towards the heavy end of the bed. For more than 60% of the cycle time, CnB provides heavy reflux to CoB for gas-phase CO₂ enrichment, thus ensuring high-purity CO₂ production. Such enrichment and movement of CO₂ adsorption front is evident from the gas-phase CO₂ concentration profile for step 1 and step 3 in Figure 4.7. Second, we observe that adsorption pressure P_{ads} operates at a lower level for most of the duration of the cycle and attains its upper bound only for a short duration. This leads to savings in power consumption. The third key aspect of this VSA cycle is the pressure equalization step (steps 4 and 8), which leads to additional power savings. Gas-phase CO₂ concentration profiles for step 4 and 8 in Figure 4.7 illustrate sharp CO₂ desorption in CoB while adsorption in CnB.

Table 4.6 summarizes the optimization results. With 46,313 variables and 191 degrees of freedom in the NLP, the optimal solution was obtained in approximately 4.5 CPU hours. At the optimum, the feed flux, CO₂ purity, and CO₂ recovery were at their respective lower bounds of 65 kgmol m⁻² hr⁻¹, 90% and 85%. Under these conditions, and at an optimum P_{feed} of 182.3 kPa, we achieved a power consumption of 464.76 kWh/tonne CO₂ captured, which is over 27% lower than Case 2. Table 4.7 lists a validation of the optimal results obtained from AMPL with accurate simulations in MATLAB for varying number of spatial finite volumes. As observed in the previous case, the purities and recoveries are in reasonable agreement, even for a large number of finite volumes.

Figure 4.8 shows the trade-off curve between power consumption and CO₂ recovery. As in the previous case, we construct the curve by varying the lower bound on CO₂ recovery,

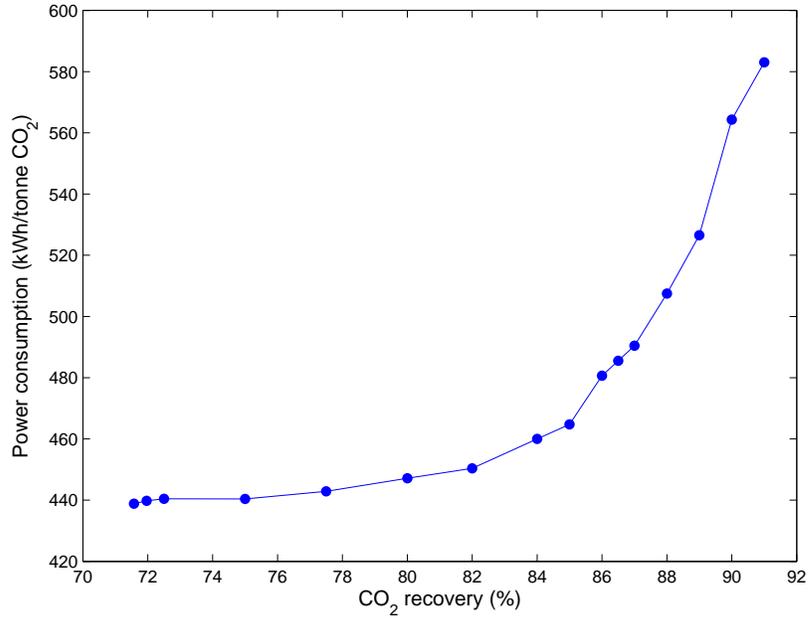


Figure 4.8: Power-recovery trade-off curve, at 90% CO₂ purity

while keeping purity at 90%, and optimizing NLP (4.3) multiple times. Thus, we obtain an optimal trade-off curve, although it is possible to obtain a different optimal cycle configuration at each point plotted on the curve. As expected, the curve shows that the power requirements increase if a high-recovery separation is desired. However, the increase is almost linear up to a recovery level of 84%. The power requirements then start growing steeply if more than 84% CO₂ recovery is sought.

4.5 Comparison of Cycles from Case II and III

Case studies discussed in the previous section clearly demonstrate that we can obtain substantially different PSA configurations after performing superstructure optimization with different objectives. In all the case studies above, the final optimal cycle is governed by the required specifications, constraints and objective function. However, optimal PSA configurations have some similarities as well, which convey that the superstructure approach finds some common features as a necessary requirement for optimal performance. For instance, the optimal cycles

obtained in both case II and case III depend heavily on the heavy reflux step to enrich gas-phase CO₂ concentration towards the heavy end of the adsorber bed. Both case studies employ this step to enhance CO₂ purity in the final product, and run it for a substantial 60-65% of the total cycle duration. This not only asserts that such a step is vital for producing heavy product at a high purity, but also proves that the superstructure approach yields intuitive and meaningful configurations since all of the literature studies have included heavy reflux step to boost CO₂ purity in the final product. Besides heavy reflux step, both cycles in case II and III are similar in terms of the optimal cycle duration, which is close to 40 minutes for both cases. Moreover, both cycles do not collect CO₂ product when a light reflux stream is present as it necessarily dilutes CO₂ product, and both employ vacuum for almost entire duration of the cycle to improve CO₂ recovery.

Barring heavy reflux step and other minor similarities, optimal cycles obtained in case II and III are quite different from each other, especially in the power consumption aspect. For case II, P_{ads} is at its upper bound while P_{des} is at its lower limit for almost entire cycle. Hence, the total power consumption is quite high for case II. In contrast, optimal profile for P_{ads} takes a lower value for the most part of the cycle and attains upper bound only for a short duration. This leads to substantial power savings. One of the key differences between cycles of case II and III is the pressure equalization step. This leads to additional power savings as it avoids uneconomical pressure drop when cycle transitions from step 4 to step 5 in case III. However, lack of a pressure equalization step causes such a pressure drop in case II between step 3 and 4 when pressure drops significantly from 450 kPa to 90 kPa. Only reason for such a contrast in the optimal solution is the absence of any constraint on power consumption in the problem formulation for case II. To avoid this, an upper bound on the power consumption can be used for case II in future. Besides power aspect, cycles in case II and III differ in steps 3 and 6. Case III doesn't incorporate a total reflux step unlike case II. Instead, that step is still a combination of light and heavy reflux steps in case III with the presence of external feed and nitrogen removal.

To deduce multibed cycles for a continuous cycle operation from the optimal two-bed solutions, a coordination of step times will be required which will depend upon whether we need a continuous N₂ removal, or a continuous product CO₂ collection or a continuous feed to the system. In any case, continuous flow can be maintained either through feed or product buffer tanks, or by adding parallel beds and ensuring that step times are integral multiples of each other to avoid overly complicated cycles.

4.6 Conclusions and Future Work

A fairly extensive review of the previous work on post-combustion CO₂ capture reveals that a systematic methodology is still required for the design of PSA cycles. To address this, we assess the applicability of the superstructure approach in this context. It is illustrated for three case studies of post-combustion CO₂ capture. The first case study optimizes the standard 2-bed 4-step Skarstrom cycle, and shows that such conventional cycles, which focus on separating light product at a high purity, fail to produce heavy product at a high purity because of the absence of a heavy reflux step. To obtain high-purity separation, the superstructure is optimized in the second case study. A 2-bed 6-step VSA cycle is derived from the solution of the optimal control problem. With this configuration, we are able to recover about 80% of CO₂ at a substantially high purity of 95%, and at a significantly high feed flux of 80 kgmol m⁻² hr⁻¹, but with a power consumption of 637 kWh/tonne CO₂ captured. Thus, in the third case study, we focus on developing optimal configuration which yields high-purity separation with minimal power requirements. We construe a 2-bed 8-step VSA configuration from the optimal profiles, with which, at 90% purity and 85% recovery, CO₂ is extracted with a substantially low power consumption of 465 kWh/tonne CO₂ captured. Hence, with the proposed superstructure approach, we are able to design optimal configurations that make pressure swing adsorption a promising option for high purity CO₂ capture from flue gas streams.

A complete discretization approach is used to solve the optimal control problem as a large-scale nonlinear program, using the nonlinear optimization solver IPOPT. Verifications of the

accuracy of the discretization scheme show this approach is reasonably accurate in capturing the dynamics of PSA systems governed by hyperbolic PDAEs and steep adsorption fronts, and can be used for PSA systems with efficient NLP solvers like IPOPT. To improve upon the accuracy of the results and eliminate the verification step, a sensitivity-based sequential approach, similar to Jiang et al. [100], will be developed in future to solve the optimal control problem.

Finally, our superstructure based methodology, demonstrated for post-combustion capture in this work, is quite generic and can be extended to many other PSA applications. In the next chapter, we demonstrate it for pre-combustion carbon capture.

Chapter 5

Superstructure Case Study: Pre-combustion CO₂ Capture

Synopsis

PSA/VSA technology has been widely applied for H₂ production from the effluent streams of a shift converter. It also offers significant advantages for pre-combustion CO₂ capture in terms of performance, energy requirements and operating costs since the shifted synthesis gas (syngas) is available for separation at a high pressure with a high CO₂ concentration. Most commercial PSA cycles recover H₂ at very high purity, but do not focus on enriching the strongly adsorbed CO₂. Thus, a major limitation exists with the use of these conventional PSA cycles for high purity CO₂ capture. Novel PSA cycle designs are anticipated which recover both H₂ and CO₂ at a high purity. We demonstrate the superstructure approach for case studies related to pre-combustion CO₂ capture. In particular, optimal PSA cycles are synthesized which maximize CO₂ recovery or minimize overall power consumption. The results show the potential of the superstructure to predict PSA cycles with purities as high as 99% for H₂ and 96% for CO₂. Moreover, these cycles can recover more than 92% of CO₂ with a power consumption as low as 46.8 kWh/tonne CO₂ captured. Hence, this chapter demonstrates the versatility of the superstructure approach.

5.1 Introduction and Previous Work

Global energy-related carbon dioxide emissions are increasing by 1.7% every year and have been estimated to reach 41 gigatonnes by 2030 [93]. Power generation accounts for about one-third of CO₂ emissions from fossil fuel use. Carbon dioxide capture and storage is a critical technology to significantly reduce CO₂ emissions, and is most applicable to large, centralized emission sources such as power plants. The purpose of CO₂ capture is to produce a concentrated stream that can be readily transported to a CO₂ storage site. One of the potential capture systems that has gained recent popularity is the pre-combustion capture system. Pre-combustion capture involves partial oxidation (gasification) of coal to produce syngas (or fuel gas) composed mainly of carbon monoxide and hydrogen. The carbon monoxide is reacted in a shift converter to increase carbon dioxide and hydrogen yield. CO₂ is then concentrated from this H₂/CO₂ mixture, resulting in a hydrogen-rich fuel and a CO₂-rich stream available for storage. Compared to post-combustion capture, a pre-combustion system is preferable for CO₂ capture because the fuel gas from the shift converter has a higher CO₂ concentration in the range 30-60%, and is also typically at a higher pressure, thus offering cost-effective means for CO₂ capture [163].

PSA offers significant advantages for pre-combustion CO₂ capture in terms of performance, energy requirements and operating costs. Voss [191] provides an overview of how the PSA units can be integrated in complex flowsheets of power plants and steam reformers for pre-combustion CO₂ capture. Industrial PSA technology to remove CO₂ and other trace components from steam reformer off-gas and fuel gas primarily focuses on producing hydrogen at a high purity, and considers CO₂ as a waste stream [23, 79, 80]. The most frequently used PSA processes in this area, the Polybed process and the Lofin process [81, 130, 171, 205], produce H₂ with more than 99.9999% purity, but consider CO₂ as a by-product and reject it in the tail gas (i.e., the desorbed gas containing H₂O, N₂, CO₂, CO, and H₂) at a much lower purity. The hydrogen recovery in these processes ranges between 60-80%, with the tail gas generally being used as a fuel for the reformer. Over the past few decades, researchers have focused on development,

improvement and optimization of novel PSA cycle configurations for H₂ purification and CO₂ removal. Cen et al. [42] studied a bench-scale 1-bed 4-step PSA process, with activated carbon as the adsorbent, to remove CO₂ from a feed mixture comprising 24.75% CO₂, 24.75% H₂, and 0.0001% H₂S. Whysall and Wagemans [199] increased the H₂ production capacity by extending the purge step in their 16-bed 13-step PSA cycle. Baksh et al. [18, 19] developed a simple 2-bed 12-step process which used layered beds packed with alumina, activated carbon and zeolite. With this configuration they were able to recover 76% H₂ at a very high purity level of 99.996%. Xu et al. [204] developed a 6-bed 16-step PSA process in which only four pressure equalization steps were incorporated. Zhou et al. [213] proposed a 4-bed 13-step PSA cycle and explored the idea of using buffer tanks to carry out pressure equalization during the cycle. Jiang et al. [100] optimized a 5-bed 11-step PSA process, using layered beds of activated carbon and zeolite 5A, and were able to achieve a hydrogen recovery of around 89% with CO impurity as low as 10 ppm in the hydrogen product stream. Jee et al. [98] studied the adsorption characteristics of various permutations of mixtures composed of H₂/CH₄/CO/N₂/CO₂, on a layered bed packed with activated carbon and zeolite 5A, and concluded activated carbon to be a suitable adsorbent for CO₂ extraction. Warmuzinski and coworkers [196] designed a 5-bed 8-step PSA process through rigorous mathematical simulation, for which they obtained a recovery of 74% for H₂, as well as 92% for methane in the tail gas stream. They also verified their results using bench-scale experimentation [184]. Yang et al. [208] studied a 4-bed 9-step cycle experimentally and theoretically using layered beds of activated carbon and zeolite 5A, and recovered 66% of H₂ from syngas at 99.999% purity. Ritter and Ebner [152] provide a comprehensive review on the use of adsorption technologies for H₂ production and CO₂ removal.

In all the PSA cycles developed so far, the weakly adsorbed hydrogen (or the light-product) in the mixture is the desired product, and enriching the strongly adsorbed CO₂ (or the heavy-product) is not a concern. On the other hand, for CO₂ sequestration, it is necessary to concentrate CO₂ to a high purity. The adsorbents designed to date preferentially adsorb

Table 5.1: Activated carbon properties and model parameters [100]

Parameter	Value	
Bed porosity (ϵ_b)	0.37	
Particle diameter (d_p)	0.00149 m	
Adsorbent density (ρ_s)	544.64 kg m ⁻³	
Bulk density (ρ_b)	343.12 kg m ⁻³	
Heat capacity of solid (C_{ps})	711.75 J kg ⁻¹ K ⁻¹	
Heat transfer coefficient (U_A)	0.2839 J m ⁻³ sec ⁻¹ K ⁻¹	
Gas viscosity (μ)	1.2021×10 ⁻⁵ kg m ⁻¹ sec ⁻¹	
Gas constant (R)	8.314 J mol ⁻¹ K ⁻¹	
Mass transfer coefficient (k)	CO ₂ =0.45 sec ⁻¹ H ₂ =1.45 sec ⁻¹	
Heat of adsorption (ΔH^{ads})	CO ₂ =24801 J mole ⁻¹ H ₂ =8420 J mole ⁻¹	
Ambient temperature (T_w)	298 K	
Isotherm parameters		
	CO ₂	H ₂
k_1^1	1.16	1.16
k_1^2	0	0
k_1^3	6.96×10 ⁻¹⁰	1.06×10 ⁻⁹
k_1^4	3259.683	1012.75
k_2^1	8.33	8.33
k_2^2	0	0
k_2^3	1.88×10 ⁻¹⁰	1.06×10 ⁻⁹
k_2^4	2706.279	1012.75

CO₂ from a flue gas or reformer off-gas mixture, consequently making it a heavy-product. The conventional PSA cycles are inappropriate for concentrating heavy-product because the light-product purge step (or the light reflux step) in these cycles uses a portion of the light-product for purge. This necessarily dilutes the heavy component in the heavy-product stream. Therefore, a pure light component is easy to attain from such cycles, but not a pure heavy component. Thus, it is necessary to develop PSA processes specifically targeted to obtain pure strongly adsorbed CO₂. Very few examples of CO₂ purification from a reformer off-gas mixture using a PSA process can be seen in the literature. Sircar et al. developed a 5-bed 5-step PSA process to extract methane and carbon dioxide both at a high purity from a feed mixture having 40-60% CO₂ and CH₄ [174, 165]. A pure CO₂ rinse step was used in the process to

obtain a CO₂ product containing 99.8-99% CO₂. Schell et al. [159] suggested a dual-reflux PSA process with a stripping and a rectifying section to obtain both light and heavy product at high purities. Xiao et al. [202] studied single-stage and dual-stage 2-bed 8-step VSA processes which could recover more than 90% of CO₂, at 95% purity, from a feed mixture having 21.5% CO₂ and 76.8% H₂. Air Products and Chemicals, Inc. have developed the Gemini process to simultaneously produce H₂ and CO₂ at high purities and recoveries [164]. It consists of 6 adsorbers (A beds) to selectively adsorb CO₂, which is then obtained by applying vacuum depressurization, and 3 adsorbers (B beds) to purify hydrogen. Both beds undergo two entirely different sequences of operating steps. However, one A bed and one B bed are connected in series during the adsorption step. Sircar [173] provides more detailed information about the process.

It is clear that novel PSA cycle sequences are anticipated which not only recover H₂ at a high purity, but simultaneously also produce a highly pure CO₂ stream with a reasonably high recovery. In this chapter, we demonstrate the versatility of the superstructure approach by applying it to develop cycles for pre-combustion capture that produce both H₂ and CO₂ at high purity and recovery.

5.2 Case Setup

Here the feed is considered to be a syngas mixture having 55% H₂ and 45% CO₂, arriving at a temperature of 310 K after a single shift conversion in an IGCC [134]. The feed mixture also consists of negligible amounts of CO, CH₄, Ar and N₂, besides H₂ and CO₂. However, hydrogen and carbon dioxide together constitute around 97-99% of the mixture [134]. Therefore, we consider a binary feed mixture for the case studies. We assume that the fuel gas enters at a pressure of 700 kPa, and a maximum velocity (v_{feed}) of 50 cm/sec. Since feed pressure is high, optional inlet compressor doesn't exist in the superstructure for this case. Consequently, work done by inlet compressor is omitted from Equation (3.15a). Since the PSA model doesn't require bed diameter to be specified, we specify superficial feed velocity for the model instead

of a volumetric flow rate. The bed length is fixed and is assumed to be 12 metres. For all the case studies, we also assume an efficiency of 72% for all compressors and the vacuum generator in the superstructure [30]. Activated carbon is chosen as the adsorbent, especially to extract CO₂. Based on the breakthrough tests, Jee et al. [98] recommended activated carbon for high recovery CO₂ separation. The scope of this study is to explore the limits of the performance of the PSA processes for this sorbent. We also note that other sorbents, such as alumina, molecular sieves, zeolite (also in layers), are also applicable and these form the basis for future study with this synthesis technique. The properties and other model parameters for activated carbon are listed in Table 5.1 [100].

Although a large number of spatial and temporal discretization nodes are essential to accurately capture the dynamic movement of the steep adsorption fronts, we consider only 10 spatial finite volumes and 10 temporal finite elements for the NLP to obtain the solution in a reasonable amount of time. Because of such a small number of nodes, accuracy validation of the optimal solution obtained from IPOPT by performing more accurate dynamic simulations in MATLAB at the optimal values is extremely essential. Here we consider two different approaches for accuracy verification. In the first approach, called the step-by-step approach, each operating step of the cycle is simulated in MATLAB for only one cycle, and the purities and recoveries are then compared with AMPL results. The initial condition for each step and the time-dependent fluxes between the beds are taken from the AMPL solution. The number of spatial finite volumes are kept same for both AMPL and MATLAB. Note that, depending on the accuracy, the MATLAB solution may or may not be at CSS after simulating each step in this approach. In the second approach, called the full-cycle approach, entire cycle is simulated in MATLAB multiple times until CSS is achieved. In this approach, we consider more spatial finite volumes for MATLAB simulation. While the step-by-step approach only verifies temporal accuracy, the full-cycle approach validates both spatial and temporal accuracy. Although the full-cycle approach yields more accurate comparison, the step-by-step approach is useful in getting a quick assessment of the validity and physical correctness of the AMPL solution.

To illustrate the generality of the superstructure approach, we consider two different cases. The first case involves superstructure optimization to obtain an optimal PSA configuration which maximizes CO₂ recovery for a given lower bound on both CO₂ and H₂ purity, while the second case involves generating optimal cycle that minimizes overall power consumption for a given lower bound on CO₂ purity and recovery.

5.3 Case Studies and Computational Results

5.3.1 Case I: Cycle synthesis to maximize CO₂ recovery

We solve the optimal control problem (3.1) to obtain an optimal cycle which maximizes CO₂ recovery for a lower bound of 90% on both H₂ and CO₂ purity. Besides this, a lower bound on feed flux Q_{feed} is also imposed. In the absence of this bound, the optimizer may force the feed fraction $\phi(t)$ to zero in order to maximize CO₂ recovery. Large-scale NLP that results after complete discretization of state and control variables in the optimal control problem is shown below. In the following problem, cycle time T_c is also a decision variable. Optimal values of the moving temporal finite elements together with optimal T_c give the optimal step times.

$$\begin{aligned}
 \max \quad & \text{CO}_2 \text{ recovery (from Equation (3.14d))} \\
 \text{s.t.} \quad & c(w) = 0 \text{ (fully discretized Equations (3.2)-(3.13))} & (5.1a) \\
 & \text{H}_2 \text{ purity} \geq 0.9 \text{ (from Equation (3.14a))} & (5.1b) \\
 & \text{CO}_2 \text{ purity} \geq 0.9 \text{ (from Equation (3.14b))} & (5.1c) \\
 & Q_{feed,L} + Q_{feed,H} \geq 35 \text{ kgmol m}^{-2}\text{hr}^{-1} \text{ (from Equation (3.14e))} & (5.1d) \\
 & P_{ads} \geq P_d & (5.1e) \\
 & P_{des} \leq P_{feed} & (5.1f) \\
 & P_a \geq P_{feed} & (5.1g) \\
 & 0 \leq \alpha(t_i), \beta(t_i), \phi(t_i) \leq 1 & \forall t_i & (5.1h) \\
 & 10 \text{ sec} \leq T_c \leq 500 \text{ sec} & & (5.1i) \\
 & 101.32 \text{ kPa} \leq P_{ads}(t_i) \leq 1000 \text{ kPa} & \forall t_i & (5.1j) \\
 & P_{des}(t_i) \geq 50 \text{ kPa} & \forall t_i & (5.1k)
 \end{aligned}$$

Equation (5.1a) is the fully discretized PDAE system with the cyclic steady state condition. Constraint (5.1e) ensures that the pressure always decreases through the valve in the super-

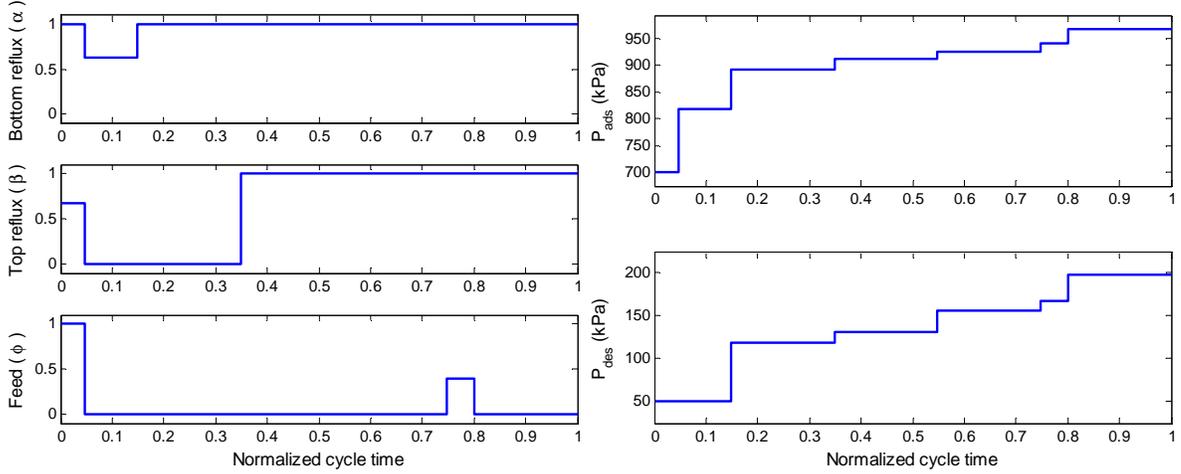


Figure 5.1: Optimal control profiles for case I

structure. Similarly, constraints (5.1f) and (5.1g) ensure that the gas is never expanded by the heavy gas and the feed compressors, respectively. It should be noted that a lower bound of 50 kPa is chosen for the vacuum generated, which is not a substantially high vacuum. Similarly, the chosen upper bound of 1000 kPa for P_{ads} is also reasonably low. Also, it is important to note that a lower bound of $35 \text{ kgmol m}^{-2}\text{hr}^{-1}$ is imposed on the total feed flux which is independent of the bed diameter. Thus, with a bigger bed diameter, it will be possible to handle a much higher feed throughput for the same cycle configuration. Hence, as in the case of post-combustion capture case studies, the focus here is to synthesize industrial-scale PSA systems.

With 10 temporal finite elements and 10 spatial finite volumes, the optimization problem was solved in AMPL using IPOPT. The optimal profiles for the control variables $\alpha(t)$, $\beta(t)$, $\phi(t)$, $P_{ads}(t)$ and $P_{des}(t)$ are shown in Figure 5.1. They are drawn against the cycle time normalized between 0 and 1. These profiles suggest an optimal 2-bed 8-step VSA cycle, illustrated in Figure 5.2, which can be deciphered in the following manner. The cycle starts with $\alpha(t)=1$, $\beta(t)$ close to 0.67, and $\phi(t)=1$. This suggests bottom reflux from CnB to CoB, a 67% top reflux from CoB to CnB and feed being fed to CoB. From the values of $P_{ads}(t)$ and $P_{des}(t)$ for this step, it can be observed that CoB is operating at the feed pressure (700 kPa) while CnB is operating at a vacuum of 50 kPa. Hence, this is an adsorption with a heavy reflux step for

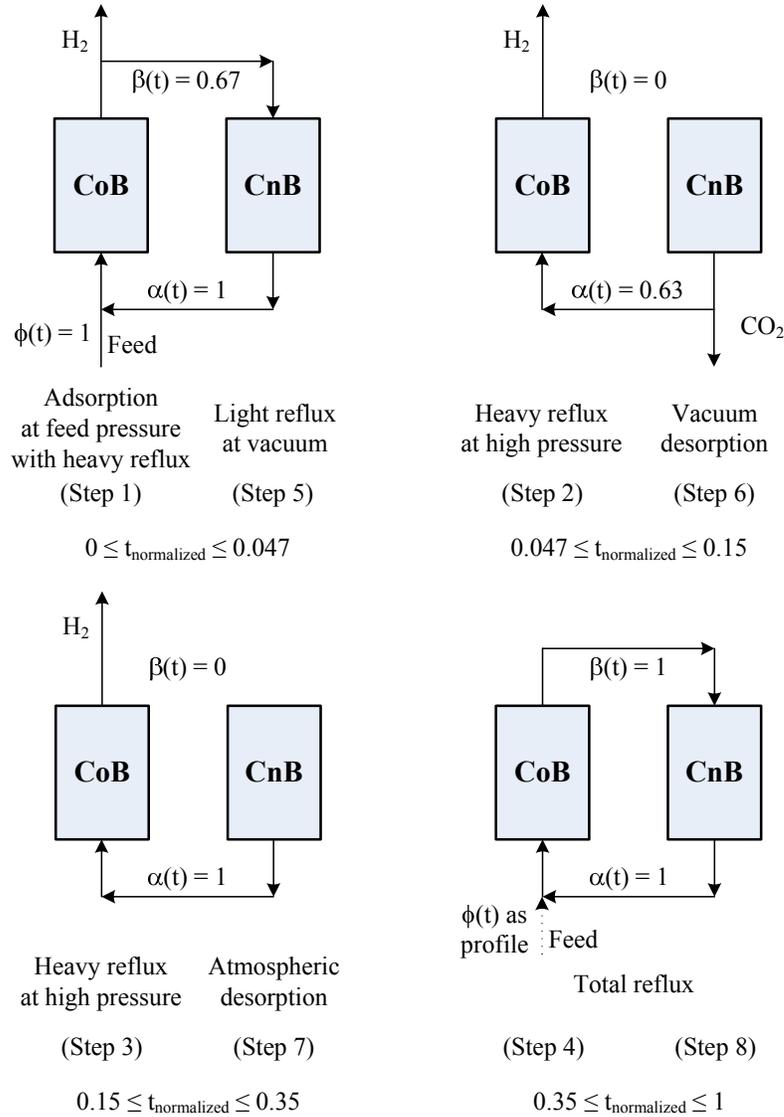


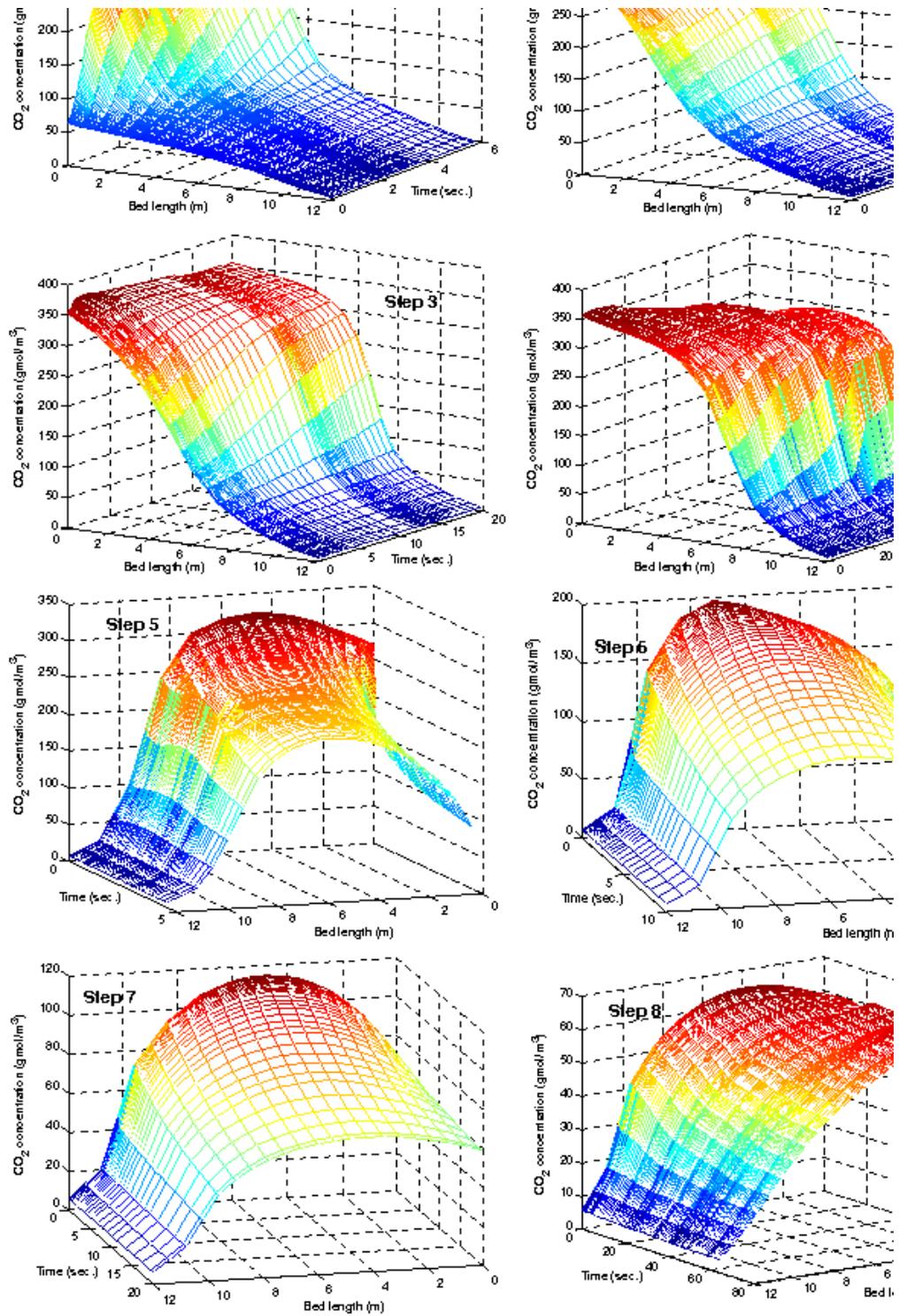
Figure 5.2: Optimal VSA configuration for case I

CoB (step 1) and a light reflux step at vacuum for CnB (step 5). After this, $\alpha(t)$ drops to around 0.63 while both $\beta(t)$ and $\phi(t)$ drop to zero. CnB continues to operate at vacuum while the pressure in CoB rises to around 820 kPa. Thus, with no feed and around 63% bottom reflux, we have a high pressure heavy reflux step for CoB (step 2) with H_2 collection at the light end, and, with no top reflux, we have a vacuum desorption step for CnB (step 6) in which a part of the desorbed CO_2 is collected, while the rest is sent as a heavy reflux to further enrich its solid-phase concentration towards the heavy end of CoB. Next, $\alpha(t)$ goes to 1 while $\beta(t)$ and $\phi(t)$ remain at zero. The pressure further rises to 900 kPa in CoB while the vacuum

is stopped in CnB and it starts operating in the atmospheric range. Therefore, we have an atmospheric desorption step for CnB (step 7) in which desorption occurs at around 120 kPa. The desorbed gas is sent to CoB which undergoes a heavy reflux step at further elevated pressures (step 3). In the final operating step, the values of $\alpha(t)$ and $\beta(t)$ both go to 1, indicating no light or heavy product extraction from the system. The feed enters midway through the step for a short duration, and is otherwise at zero. To reflect this, a dotted line is shown for feed during this step in Figure 5.2. The profiles of $P_{ads}(t)$ and $P_{des}(t)$ show that the pressure rises in both beds. Since the PSA system gets isolated during this step and a recirculation of the components occurs within the system, we call it a total reflux step (step 4 and step 8). After the total reflux step, the co-current bed follows the steps of the counter-current bed and vice-versa. This completes the cycle.

Figure 5.2 together with the gas-phase CO₂ concentration profiles in Figure 5.3 illustrate several unconventional, but key, aspects of the cycle. First, the light reflux step at vacuum (step 5) follows the total reflux step at around 950 kPa (step 4). Such a transition in the bed pressure, although not economical, is essential to improve the purity and recovery of CO₂ in the final product. During the light reflux step, a large amount of CO₂ desorbs in CnB which is then sent to CoB. This is necessary to enrich the adsorbed-phase CO₂ concentration towards the heavy end of CoB. From step 1 in Figure 5.3, it can be observed that the CO₂ front rises significantly towards the heavy end due to this recycle. Such a significant rise is important to achieve the desired CO₂ purity and recovery. Since a large amount of CO₂ is desired for this enrichment, and since the duration of the light reflux step is short, the step operates at vacuum and entire desorbed CO₂ is sent to the other bed.

Second, the light reflux step at vacuum (step 5) precedes the vacuum desorption step (step 6), whereas conventionally it is vice-versa. Since the step duration for step 5 is small, the hydrogen recycle helps in getting more CO₂ desorbed in that interval. This hydrogen reflux is obtained from the feed stream going in CoB. Also, since the hydrogen reflux dilutes the product CO₂, it is collected during the next vacuum desorption step and not during step 5.

Figure 5.3: Gas-phase CO₂ concentration profiles for case I

Therefore, vacuum desorption succeeds the light reflux step. The third key aspect of the cycle is the presence of heavy reflux from CnB to CoB during the entire cycle. From the CO₂ concentration profiles of first four steps in Figure 5.3, it is clear that this CO₂ reflux helps push the CO₂ front towards the light end of the adsorbing bed before we start desorbing and collecting CO₂. Thus, we infer that the heavy reflux step is essential for high purity CO₂ production.

Another aspect of the cycle is the atmospheric desorption step (step 7) after vacuum desorption (step 6). Since CO₂ is not collected as a product during step 7, we observe that the purpose of this step is only to send CO₂ reflux to CoB. The step is carried out at the atmospheric conditions to ensure a controlled CO₂ reflux to CoB such that the CO₂ front doesn't break through CoB's light end. Final aspect of the cycle is the total reflux step (steps 4 and 8). It is a mutual reflux step in which the CO₂ reflux from CnB to CoB helps push hydrogen out of the light end of CoB to the light end of CnB while enhancing adsorbed CO₂ concentration in CoB, while the H₂ reflux enriches its concentration in CnB and helps CO₂ desorb out of the heavy end of CnB. Such a step is important to ensure that both H₂ and CO₂ are collected at a high purity in subsequent steps, and thus is the longest step in the cycle. The feed stream in the middle of the step provides more hydrogen for the light reflux from CoB to CnB.

The optimization results for this case are summarized in Table 5.2. With 10,512 variables and 78 degrees of freedom in the NLP, the optimal solution was obtained in approximately 52 CPU minutes on an Intel Quad core 2.4 GHz machine with 8 GB RAM. At the optimum, the feed flux attained its lower bound of 35 kgmol m⁻² hr⁻¹. For this feed flux, and 72% efficiency for compressors and vacuum generator, a power consumption of 536.16 kWh/tonne CO₂ captured was obtained after optimization. An optimum CO₂ recovery of 98% at a purity of 90% was obtained. Also, a reasonably high hydrogen purity of 98% and a recovery of 91% was obtained simultaneously.

In Table 5.2 we also provide a validation of the optimal results obtained using full discretization approach in AMPL with the method of lines simulations in MATLAB. As discussed in

Table 5.2: Optimization results for case I

No. of variables	10512		
No. of equations	10434		
CPU time	51.59 min.		
Optimal step times			
Step 1 (and 5)	4.69 sec		
Step 2 (and 6)	10.02 sec		
Step 3 (and 7)	19.88 sec		
Step 4 (and 8)	64.81 sec		
Optimal cycle time	198.8 sec		
Feed flux	35 kgmol m ⁻² hr ⁻¹		
Power consumption	536.16 kWh/tonne CO ₂ captured		
Accuracy check			
	Full discretization	MATLAB verification	
		step-by-step	full-cycle
Spatial finite volumes	10	10	40
H ₂ purity	98.20%	99.10%	95.92%
H ₂ recovery	91.09%	91.32%	91.73%
CO ₂ purity	90%	90.32%	90.99%
CO ₂ recovery	97.95%	98.99%	96.03%

section 5.2, AMPL results were validated using both step-by-step and full-cycle approaches in MATLAB. The step-by-step validation was done with the same number of spatial finite volumes as used in AMPL, i.e., 10, while full-cycle validation was done with 40 finite volumes. We observe that the results from AMPL are in reasonable agreement with those from MATLAB for both the approaches. The step-by-step verification is closer to the AMPL solution because the initial conditions for each step and the time-dependent fluxes between the beds are taken from AMPL, and MATLAB only verifies the temporal accuracy of the AMPL solution. On the other hand, the full-cycle approach reflects more accurate comparison as it simulates the entire cycle and verifies both spatial and temporal accuracy. We observe a reasonably good comparison with the full-cycle approach as well. Moreover, we note that as we switch from CoB to CnB or vice-versa during the cycle, it takes a short while for the flow to reverse entirely in the bed. As a result, a flow of components from the heavy end of CoB or the light end of CnB is observed during this short duration. Such a flow is accounted in the purity and re-

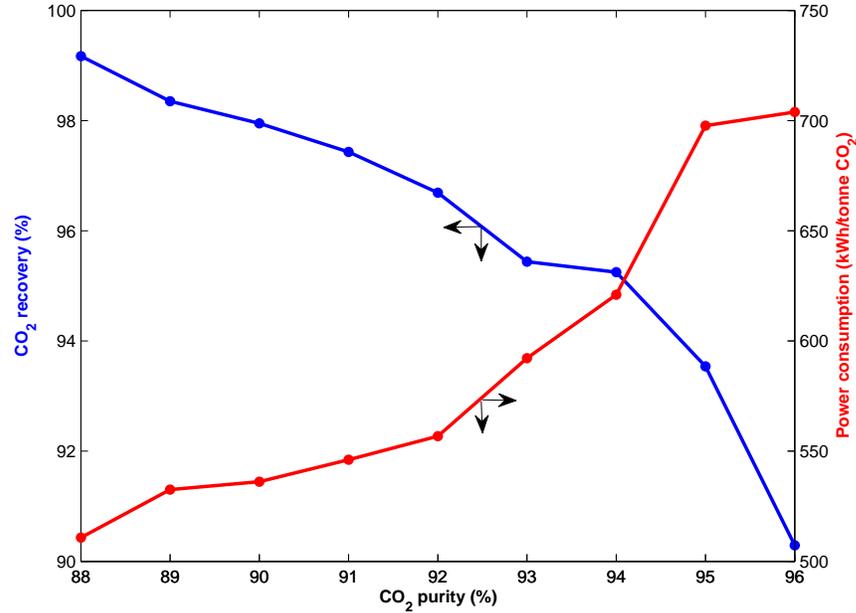


Figure 5.4: Purity-recovery trade-off curve for case I

covery calculations in AMPL and the full-cycle approach, since they simulate the entire cycle, but not in the step-by-step approach. Hence, we observe higher recoveries for H₂ and CO₂ in the step-by-step approach. We register this flow because in our formulation we control the pressures P_{ads} and P_{des} and not the flow rates at the heavy end and the light end of CoB and CnB, respectively. To avoid this, a valve-based superstructure formulation, which can control the flows instead of pressures, will be considered in future extensions of this work.

Figure 5.4 shows a trade-off curve between CO₂ purity and recovery. The curve is constructed by varying the lower bound on CO₂ purity and solving the superstructure NLP repeatedly. As a result, each point plotted on the curve represents an optimal cycle which yields the corresponding optimal CO₂ recovery for the corresponding purity. In other words, it is an optimum purity-recovery trade-off curve for the activated carbon adsorbent and the process conditions assumed in this case study. The feed flux and the cycle time were fixed to their respective optimal values of 35 kgmol m⁻² hr⁻¹ and 198.8 sec for the entire curve. Figure 5.4 also shows the power consumption for the corresponding optimal CO₂ purity-recovery combination. With activated carbon as the sorbent, we are able to obtain a maximum purity of

around 96% with a recovery of 90%, but with a power consumption of around 700 kWh/tonne CO₂ captured. For this system, high CO₂ purity (> 99%) is not possible with activated carbon as the sorbent. The curve shows that if a very high purity CO₂ separation is desired then the recovery falls drastically. A similar trend is observed with the purity when a very high CO₂ recovery is sought. The intermediate section of the curve is a preferable region to operate.

5.3.2 Case II: Cycle synthesis to minimize power consumption

Although we achieved a high purity separation in the previous case, the power consumption was also quite high. Therefore, in this case, we modify the objective function of the optimization problem from maximizing CO₂ recovery to minimizing overall power consumption. A lower bound of 92% is specified for CO₂ recovery, while no lower bounds are specified for hydrogen purity and recovery. The efficiency is kept same as 72% for all compressors and vacuum generator. The rest of the optimization problem remains same as in the previous case, and is as below.

$$\begin{aligned}
 \min \quad & \text{Power (from Equation (3.15b))} \\
 \text{s.t.} \quad & c(w) = 0 \text{ (fully discretized Equations (3.2)-(3.13))} \\
 & \text{CO}_2 \text{ purity} \geq 0.9 \text{ (from Equation (3.14b))} \\
 & \text{CO}_2 \text{ recovery} \geq 0.92 \text{ (from Equation (3.14d))} \\
 & Q_{feed,L} + Q_{feed,H} \geq 35 \text{ kgmol m}^{-2}\text{hr}^{-1} \text{ (from Equation (3.14e))} \\
 & P_{ads} \geq P_d \\
 & P_{des} \leq P_{feed} \\
 & P_a \geq P_{feed} \\
 & 0 \leq \alpha(t_i), \beta(t_i), \phi(t_i) \leq 1 \quad \forall t_i \\
 & 10 \text{ sec} \leq T_c \leq 500 \text{ sec} \\
 & 101.32 \text{ kPa} \leq P_{ads}(t_i) \leq 1000 \text{ kPa} \quad \forall t_i \\
 & P_{des}(t_i) \geq 50 \text{ kPa} \quad \forall t_i
 \end{aligned} \tag{5.2}$$

As in the previous case, 10 temporal finite elements and 10 spatial finite volumes were chosen for complete discretization in AMPL. The optimal control profiles obtained for $\alpha(t)$, $\beta(t)$, $\phi(t)$, $P_{ads}(t)$ and $P_{des}(t)$ are shown in Figure 5.5. These profiles translate in a 2-bed 10-step VSA cycle, illustrated in Figure 5.6, which can be deduced in the following manner. The cycle starts with the first step similar to the first step of the cycle obtained in the previous case. However, the duration of this step is extremely short in this case. With $\alpha(t)=1$, $\phi(t)=1$,

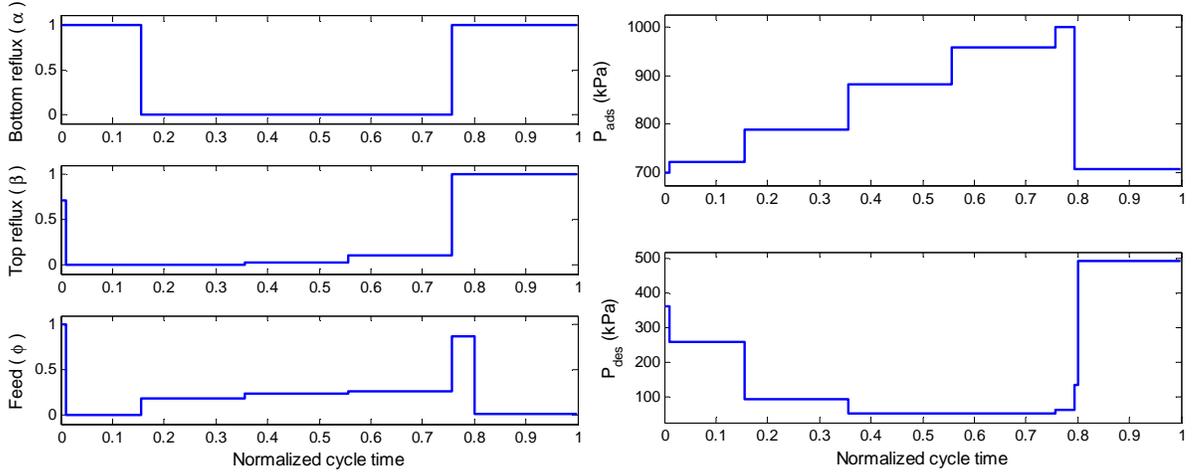


Figure 5.5: Optimal control profiles for case II

and $\beta(t)$ close to 0.72, CoB undergoes an adsorption with a heavy reflux step (step 1) at the feed pressure, while CnB undergoes a light reflux step (step 6) at around 380 kPa. After this, $\alpha(t)$ stays at 1 while $\beta(t)$ and $\phi(t)$ drop to zero. Also, $P_{ads}(t)$ rises slightly while $P_{des}(t)$ drops considerably. Therefore, we observe a high-pressure heavy reflux step for CoB (step 2) and a counter-current depressurization step for CnB (step 7). Then we observe the longest step of the cycle in which $\alpha(t)$ drops down to zero, $\beta(t)$ rises slightly from zero towards the end of the step, and the profile of $\phi(t)$ indicates near constant feed to CoB. From the pressure profiles it can be inferred that the pressure rises steadily in CoB during this step, while CnB first operates at atmospheric pressure and then at a vacuum of 50 kPa. Thus, we have a feed pressurization with adsorption step for CoB (step 3) and a desorption step for CnB (step 8). Both hydrogen and carbon dioxide are collected at high purity during this step. A small value of $\beta(t)$ towards the end of the step suggests a small amount of light reflux from CoB to CnB to increase CO_2 recovery. This small light reflux is shown as a dotted connection between CoB and CnB in Figure 5.6. Next, a short step is observed in which the values of $\alpha(t)$ and $\beta(t)$ both go to 1 and feed rises to a value close to 1. $P_{ads}(t)$ hits the upper bound of 1000 kPa while P_{des} stays in the vacuum range. This suggests a feed pressurization with heavy reflux step for CoB (step 4) and a light reflux step for CnB (step 9). In the final step, both $\alpha(t)$ and $\beta(t)$ stay at 1 while feed goes to zero. In addition CoB pressure drops CnB pressure rises so

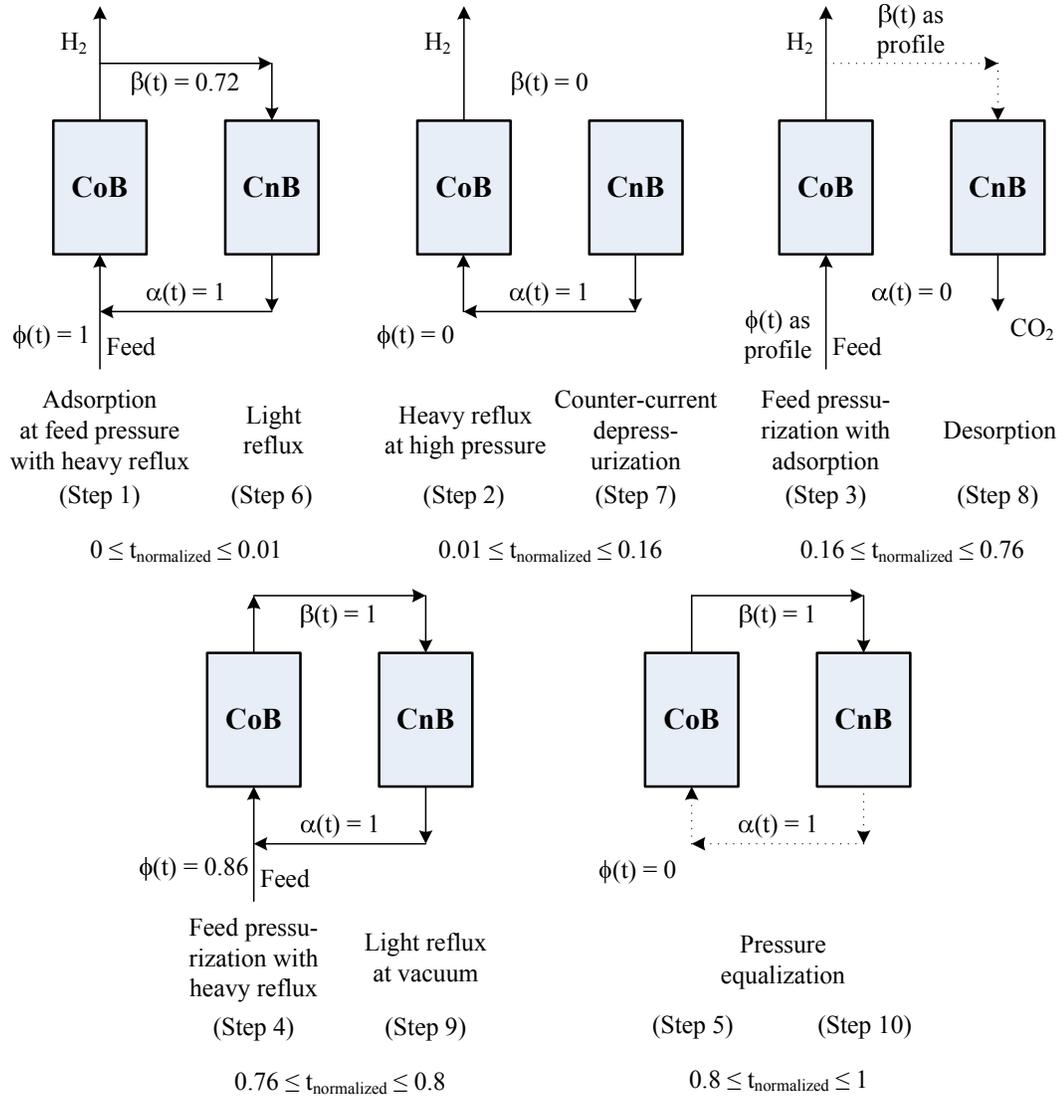
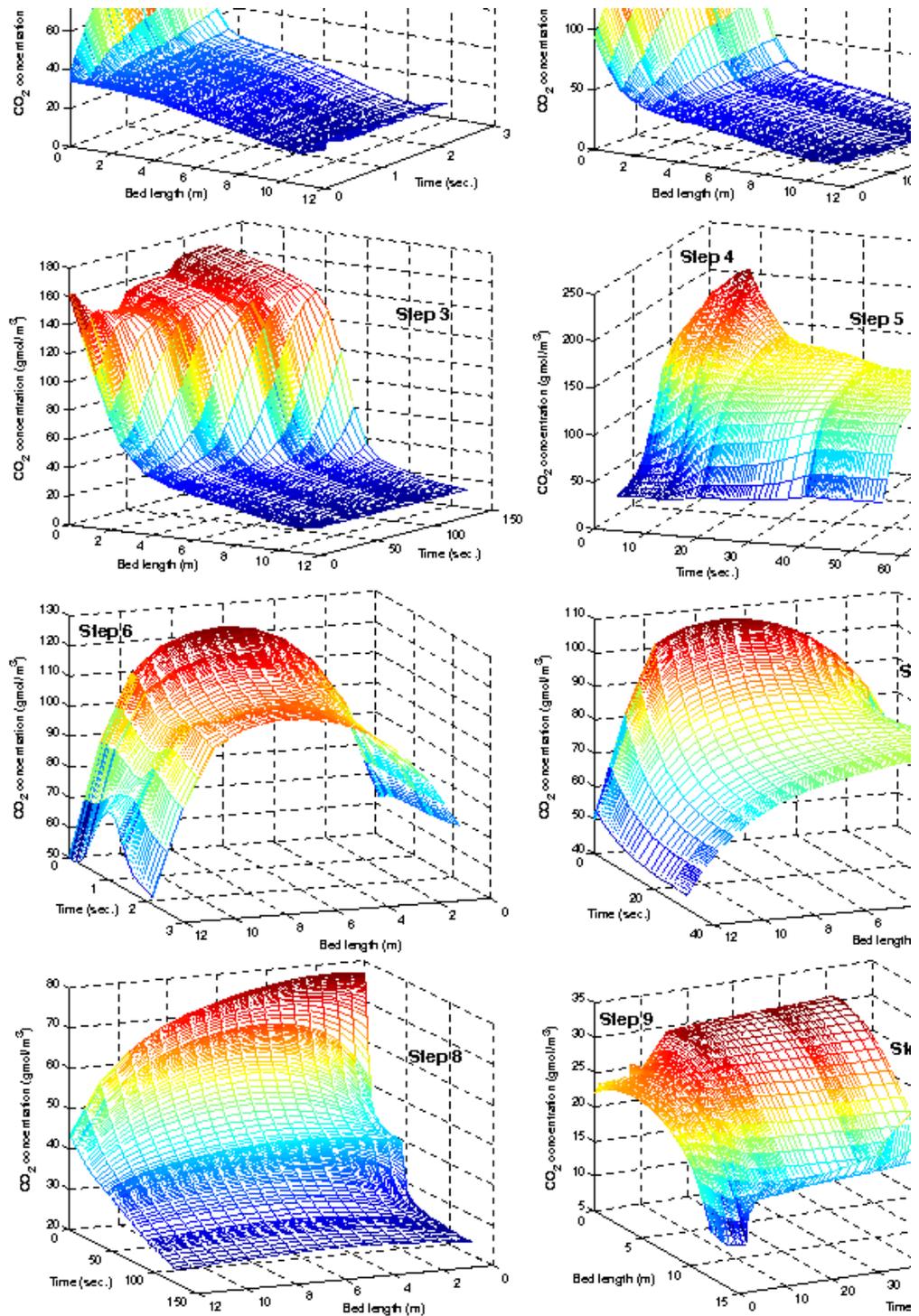


Figure 5.6: Optimal VSA configuration for case II

that $P_a = P_{des}$. This leads to an energy-saving pressure equalization step for both beds (step 5 and step 10). Although $\alpha(t)$ is 1, not much flow is observed from CnB to CoB during this step. Thus, this reflux is shown as a dotted connection in Figure 5.6. Further, the co-current bed follows the steps of the counter-current bed and vice-versa. This completes the cycle.

The optimal VSA cycle obtained in this case incorporates conventional operating steps, and in a conventional order. Figure 5.6 together with the gas-phase CO_2 concentration profiles in Figure 5.7 illustrate several key aspects of the cycle. First we observe that in this cycle heavy reflux is not used as a major step to enrich adsorbed-phase CO_2 concentration towards

Figure 5.7: Gas-phase CO_2 concentration profiles for case II

the heavy end of the bed. The light reflux step (step 5) for CnB, since carried out at a high pressure, doesn't contribute much CO₂ for enrichment for CoB during step 1, as observed from the CO₂ concentration profile for step 1 in Figure 5.7. Similarly, we observe from the concentration profiles for step 2 in Figure 5.7 that the CO₂ reflux from CnB to CoB during step 2 and step 7 marginally pushes the CO₂ adsorption front in CoB. This suggests that the heavy reflux from CnB to CoB during steps 1 and 2 are specifically used to push hydrogen out of the light end of CoB. However, though for a short duration, we do observe the use of heavy reflux to concentrate CO₂ towards the heavy end of CoB during the light reflux step at vacuum (steps 4 and 9). This vindicates the use of vacuum conditions in CnB during this step to provide a large amount of CO₂ for heavy reflux. Moreover, since the duration of these steps is short, the feed stream jumps in CoB to provide enough hydrogen for CnB as a light reflux.

Second aspect of the cycle is that the CO₂ enrichment towards the heavy end of the bed is mostly done with the feed stream. Unlike previous case, this optimal VSA cycle utilizes the fact that the feed stream has a high concentration of CO₂ at a high pressure. From the concentration profile of step 3 in Figure 5.7, it is clear that the feed stream is primarily used to push the CO₂ adsorption front. This not only allows a higher feed throughput and enhanced CO₂ recovery for the process, but also reduces the specific power consumption. Such a step is a conventional way of elevating CO₂ concentration in the bed, and thus makes this VSA cycle more conventional.

The final key aspect of the cycle is the pressure equalization step (steps 5 and 10) which leads to savings in the power consumption. Although this step saves energy, it can be observed from the concentration profiles of step 5 that as the pressure drops in CoB, CO₂ starts diffusing towards the light end of CoB. As a result, a small amount of CO₂ breaks through the light end of CoB and enters the light end of CnB, which is clear from the concentration profiles of step 10 in Figure 5.7. However, the amount is minimal and doesn't lead to a loss in CO₂ recovery.

The optimization results for this case are summarized in Table 5.3. With the same number of variables and degrees of freedom as in the previous case we were able to get the optimal

Table 5.3: Optimization results for case II

No. of variables	10512		
No. of equations	10434		
CPU time	66.45 min.		
Optimal step times			
Step 1 (and 6)	2.16 sec		
Step 2 (and 7)	31.10 sec		
Step 3 (and 8)	127.42 sec		
Step 4 (and 9)	7.77 sec		
Step 5 (and 10)	43.92 sec		
Optimal cycle time	424.74 sec		
Feed flux	96.61 kgmol m ⁻² hr ⁻¹		
Power consumption	46.82 kWh/tonne CO ₂ captured		
Accuracy check			
	Full discretization	MATLAB verification	
		step-by-step	full-cycle
Spatial finite volumes	10	10	40
H ₂ purity	93.33%	94.14%	94.22%
H ₂ recovery	91.64%	93.02%	91.05%
CO ₂ purity	90%	91.59%	89.42%
CO ₂ recovery	92%	92.92%	93.67%

solution in around 1 CPU hour. An optimal power consumption of 46.82 kWh/tonne CO₂ captured was obtained which is an order of magnitude less than the one obtained in the previous case. The low power consumption stems from an optimal feed flux, 96.61 kgmol m⁻² hr⁻¹ that is three times the feed flux of case I, and an optimal cycle time which is more than twice as long as in case I. Since the cycle is handling three times the feed over longer time, the amount of CO₂ recovered increases which leads to a lower work done per tonne of CO₂ captured. Another reason for the savings in power consumption is the pressure equalization step, discovered by the NLP solver.

At the optimum, CO₂ purity and recovery were at their respective lower bounds of 90% and 92%. With this, a reasonable hydrogen purity of 93% and recovery of 91.6% was obtained. Table 5.3 also lists the accuracy verification of the results obtained from the full discretization approach in AMPL. The purities and recoveries obtained from MATLAB using both the step-

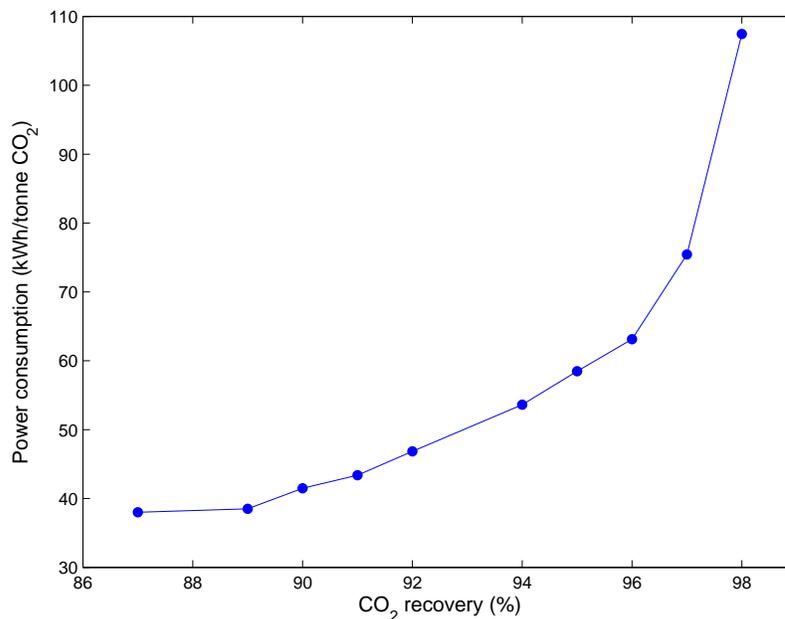


Figure 5.8: Power-recovery trade-off curve at 90% CO₂ purity for case II

by-step approach with 10 spatial finite volumes and the full-cycle approach with 40 finite volumes are reasonably close to the ones obtained from AMPL. As observed in the previous case, the step-by-step approach verification is closer to the AMPL solution. However, the full-cycle verification depicts more accurate comparison since it compares both spatial and temporal accuracy for the entire cycle.

Figure 5.8 shows the trade-off curve between power consumption and CO₂ recovery. As in the previous case, the curve is constructed by varying the lower bound on CO₂ recovery, while keeping the CO₂ purity, feed flux and cycle time fixed to their respective optimal values of 90%, 96.61 kgmol m⁻² hr⁻¹ and 424.74 sec, and solving the superstructure NLP repeatedly. As a result, each point on the curve represents the minimum power consumption that can be obtained for the corresponding CO₂ recovery. As expected, the curve shows that the power requirements increase if a high-recovery separation is desired. However, the increase remains reasonable even for a CO₂ recovery as high as 98%.

5.4 Comparison of Cycles from Case I and II

From the case studies above, we observe that the superstructure optimization can yield entirely different configurations with different objectives. The final configurations obtained match the respective objectives sought in both case studies. The major difference between the optimal cycles is the way they enrich the CO₂ concentration towards the heavy end of the bed. Since the objective of case I is to maximize CO₂ recovery, the optimizer achieves it by minimizing the feed input through the system, thus attaining the specified lower bound for feed flux. As a result, minimal feed is used and the optimal configuration doesn't use the high CO₂ concentration present in the feed to enhance the adsorbed-phase CO₂ concentration. Consequently, we observe the utilization of the heavy reflux step through the entire cycle to achieve desired CO₂ purity. In contrast, the optimal VSA cycle in case II utilizes the feed stream for CO₂ enrichment. Thus, we infer that a heavy reflux step is not an absolute necessity to obtain heavy component at a high purity when the feed to the PSA system is sufficiently rich in the heavy component.

As a result of the CO₂ enrichment through feed, although the lower bound on feed flux is 35 kgmol m⁻² hr⁻¹ for both cases, the optimal feed flux for case II is almost three times this value. Consequently, it also decreases the specific power consumption for the cycle. In contrast, the optimal cycle in case I doesn't incorporate any power saving step due to the lack of any constraint on the power consumption in the problem formulation. Thus, unlike case II, we do not observe a pressure equalization step in case I. In fact we observe an uneconomical pressure drop from 950 kPa to 50 kPa when the cycle transitions from step 4 to step 5. To avoid this, an upper bound on the power consumption can be used for case I in future.

To deduce multibed cycles for a continuous cycle operation from the optimal two-bed solutions, a coordination of step times will be required which will depend upon whether we need a continuous product H₂ collection, or a continuous CO₂ removal or a continuous feed to the system. In both case I and case II, H₂ is collected for a longer period in the cycle and continuous flow can be maintained through product buffer tanks. Thus, the coordination can

be achieved with a small number of beds. However, in case I, feed is given or CO₂ is removed for a short duration in the cycle. Such small step times, without feed and product buffer tanks, can lead to a large number of parallel beds in the continuous operation. On the contrary, the optimal cycle in case II handles a large amount of feed and removes CO₂ for a long duration. Consequently, a continuous cycle operation will require a small number of parallel beds. Thus, the optimal cycle obtained in case II is more practical and implementable. To avoid the kinds of steps obtained in case I, the step times can be constrained to avoid an overly complicated cycle. One way to handle this is to set the step times as integer multiples of each other; this can be enforced with a "slot-based" formulation. Such a formulation will be considered in future extensions of this work.

5.5 Conclusions and Future Work

A major limitation exists with the use of conventional PSA cycles for high purity CO₂ capture because they have been designed to recover H₂ at an extremely high purity, and consider CO₂ as a waste stream. Therefore, it is necessary to develop PSA processes which simultaneously produce H₂ and CO₂ at a high purity. Complex dynamic behavior of PSA processes together with the numerical difficulties of the model governed by PDAEs makes the evaluation of different cycle configurations challenging and computationally expensive. In this work, we propose a systematic optimization-based framework to address this issue. The proposed approach is illustrated for two different case studies of pre-combustion CO₂ capture using only activated carbon as the sorbent. The first case study deals with obtaining optimal PSA cycle which maximizes CO₂ recovery for at least a desired amount of CO₂ and H₂ purity. Superstructure optimization for this case results in a 2-bed 8-step VSA cycle which can produce both H₂ and CO₂ at a substantially high purity of 98% and 90%, respectively. A significantly high CO₂ recovery of 98% is achieved at a high feed flux of 35 kgmol m⁻² hr⁻¹. Changing the objective to minimizing power consumption, in the second case study, yields an entirely different 2-bed 10-step VSA cycle. The cycle can produce CO₂ at a purity of 90% and a recovery of 92%

with a significantly low power consumption of 46.82 kWh/tonne CO₂ captured. With these results it can be inferred that PSA/VSA is a promising technology for pre-combustion capture systems. It can produce highly concentrated CO₂ streams with minimal energy requirements.

Both case studies were solved to optimality within 1 CPU hour in AMPL using IPOPT with a reasonable accuracy. Thus, the proposed superstructure approach, with a complete discretization framework and efficient NLP solvers like IPOPT, is a computationally inexpensive way to obtain optimal cycles. However, as briefly mentioned in section 4.6, to improve upon the accuracy of the approach a sensitivity-based sequential approach, similar to [100], will also be developed to solve the optimal control problem for the superstructure without a separate verification step. Instead, the PDAEs for the PSA system will be decoupled from the optimization problem, and the partially discretized PDAEs, together with the sensitivities of the state variables with respect to decision variables, will be integrated outside the optimization problem using a sophisticated dynamic simulator which is able to capture the state variable profiles with high accuracy. The optimization problem will then be solved for the decisions using these sensitivities.

Finally, as mentioned in section 4.6, our superstructure based methodology, is quite generic and can be extended to many other PSA applications; no assumptions are made on the adsorbent or feedstock, the operating steps that can be predicted, or details of the bed models. This makes the approach fairly general. Moreover, the superstructure can also be used to evaluate different kinds of adsorbents for the same feedstock and process conditions. While the current superstructure involves only two beds, in future we plan to extend the formulation to incorporate more beds with multiple layers of adsorbents, more complex flow patterns and more challenging multi-component mixtures.

Chapter 6

Reduced-order Modeling for Optimization

Synopsis

Reduced-order modeling techniques can be used in order to circumvent computational difficulties that arise due to large-scale state equations related to PDE-constrained optimization problems. Model reduction is one approach to generate cost-efficient low-order models which can be used as surrogate models in the optimization problems. This chapter develops a reduced order modeling framework based on proper orthogonal decomposition (POD), which is a low-dimensional approximation to the dynamic PDE-based model. The proposed method leads to a DAE system of significantly lower order, thus replacing the one obtained from spatial discretization and making the optimization problem computationally efficient. We explain the concept of POD, the methodology to construct reduced-order models (ROMs), and motivate it with an example of Burgers equation. Further, we discuss ROM-based optimization, and describe how ROMs can be utilized to optimize in a trust-region around the point where ROM is constructed. Optimization strategy is illustrated with a hydrogen PSA case study.

6.1 Motivation

It is clear that the mathematical model of pressure swing adsorption processes is described by coupled nonlinear partial differential and algebraic equations distributed in space and time with periodic boundary conditions that connect the processing steps together, and high nonlinearities arising from non-isothermal effects and nonlinear adsorption isotherms. Also, the solution of such convection dominated hyperbolic PDAEs is governed by steep adsorption fronts. Consequently, a large number of spatial finite volumes are generally required to capture dynamic behavior with steep fronts. As a result, optimization of such systems for either design or operation represents a significant computational challenge to the current DAE optimization techniques and nonlinear programming algorithms.

Although sophisticated optimization strategies have been developed and applied to PSA systems with a significant improvement in the performance of the process (such as the complete discretization based approach by Nilchan [138] for optimization of a bench-scale and a rapid PSA process, a mixed-integer nonlinear programming based approach by Smith et al. [177, 178, 179] to minimize number of beds, an SQP-based approach by Ko et al. [110, 111] to optimize PSA and fractionated vacuum PSA processes, an SQP-based approach by Jiang et al. [99] with direct sensitivities to obtain derivatives for the optimization problem, and the complete discretization approach with the interior-point nonlinear solver IPOPT applied in chapter 4 and 5 for the case studies related to the superstructure optimization), even the most efficient of these approaches can usually be quite expensive and prohibitively time-consuming. For instance, we report a CPU time of 12.6 hrs. for case II, and 4.5 hrs. for case III of the post-combustion capture case study in sections 4.4.2 and 4.4.3, respectively. Even for just 10,500 variables in the optimization problem, CPU time was as high as 52 min. for case I, and 66 min. for case II of the pre-combustion capture case study in section 5.3.1 and 5.3.2, respectively. Jiang et al. [100] reported a CPU time of 50-200 hrs. on a 2.4 GHz linux machine for a 5-bed 11-step PSA process optimization to maximize hydrogen recovery. Multiobjective optimization of a simple single bed air drying PSA process by Sankararao et al. [157] took 720 hrs. on a 2.99

GHz Pentium IV machine. This gives a strong motivation to develop cost-efficient and robust optimization strategies for PSA processes. Moreover, for flowsheet optimization, incorporation of dynamic PSA models with other steady-state models in the flowsheet requires much faster approaches for integrated optimization.

Model reduction is a powerful tool that permits systematic generation of cost-efficient representations of large-scale systems that, in particular, result from the discretization of the PDEs. Antoulas et al. [11] provide an overview of numerous model reduction techniques which can be applied to large-scale PDE based systems. In particular, over the past decade, proper orthogonal decomposition (POD) has been developed as a powerful model reduction approach that provides an accurate reduction of the large spatially distributed models to much smaller models, and have resulted in an extensive list of articles. Berkooz et al. [27] provide a detailed list of such articles. Chatterjee [45] explains the concept and applications of POD in a clear and concise manner. POD (also known as Karhunen-Loève approximation) based reduced-order modeling technique is ubiquitous and has been applied for a multitude of applications. It has been extensively used in obtaining low-dimensional models for efficient simulation [116, 117, 128, 131, 153] and control [12, 26, 115, 147] in fluid dynamics. Yuan et al. [210] developed reduced-order models for bubbling fluidized beds. Armaou et al. [13] utilized the concept of POD for a diffusion reaction process, while Theodoropoulou et al. [185] extended the applications of ROM-based modeling to chemical vapor deposition. Park et al. [141] applied Karhunen-Loève approximation to a nonlinear heat conduction equation defined on an irregular domain. Shvartsman et al. [162] utilized POD for generating ROMs for MOVPE reactor. Couplet et al. [56], Favier et al. [68], and Galletti et al. [82] developed calibrated reduced-order modeling techniques using POD for the laminar and turbulent flow problems. Cao et al. [39] developed a POD-based reduced order model for analysis of a detailed model of the upper tropical Pacific Ocean. Gunzburger et al. [89] developed a generic framework to incorporate boundary conditions having multiple parameters in a POD-based ROM of time-dependent PDEs. The framework addresses both homogeneous and nonhomogeneous boundary

conditions. On similar lines, Rambo et al. [146] developed ROMs with parametric conditions specifically for turbulent forced convection problems. Willcox has done extensive research in the field of POD-based reduced-order models (see [200, 201, 15, 34, 35, 36]). In particular, Bui-Thanh et al. [37] provided a goal-oriented framework for generating POD-based ROMs by solving a model-constrained optimization problem.

ROMs are derived from solutions of detailed distributed models through the representation of eigenfunction expansions. Using the singular value decomposition, spatially distributed eigenfunctions can be derived empirically through a set of well-defined minimization problems. ROMs are then formulated through the substitution of the eigenfunction expansion into the PDE model using Galerkin projection. Truncation of those modes that have no significant contribution to the solution profile then leads to a significant reduction in the number of states which eventually leads to a much smaller optimization problem. Numerous studies report use of reduced-order modeling for the purpose of optimization and optimal control. Kunisch et al. [115] used it to control Burgers equation. Armaou et al. [14] applied the concept of model reduction to the time-dependent parabolic PDEs and generated ROMs for optimization of a diffusion-reaction process. Bendersky et al. [22] used it for optimization of the transport-reaction processes, while Theodoropoulou [185] optimized the chemical vapor deposition process using ROMs. Luna-Ortiz et al. [127] developed an input-output based optimization scheme for large-scale systems. Fahl [65] applied a trust region proper orthogonal decomposition algorithm for the optimal boundary control of a cavity flow. Bergmann et al. [25] also applied the same algorithm for optimal control of the circular cylinder wake flow considered in a laminar regime. Balsa-Canto et al. [21] solved a problem related to design and optimization of thermal sterilization in the food industry, particularly conduction-heated foods, using POD-based ROMs. LeGresley et al. [122] performed analysis and design optimization of inviscid airfoils using ROMs for both subsonic and transonic flows. Weickum et al. [198] developed extended ROMs for optimal design problems, in which ROMs are developed for the whole design space before optimization and then a trust region based strategy is used for

globalization. Recently, Varshney et al. [189] utilized ROMs to optimize multiscale systems. However, until now ROM-based optimization has been used only for small-scale optimal control or dynamic optimization problems, and its use for large-scale dynamic optimization problems, especially PSA, has not been explored. For PSA, very few studies involve model reduction or simplification. A compact representation developed by Chlendi et al. [49] using design of experiments, short-cut models suggested by Chung et al. [53], and a model simplification strategy chalked out by Zhang et al. [211], which relies on understanding detailed physics behind each operating step, are the only known articles. Our focus is to develop systematic model reduction strategies for PSA processes using POD.

In subsequent sections we describe the concept of proper orthogonal decomposition and demonstrate construction of reduced-order models using POD. We also illustrate ROM construction with the help of an example of Burgers equation.

6.2 Proper Orthogonal Decomposition (POD)

6.2.1 Concept

Proper orthogonal decomposition (POD), first introduced by Lumley [126] is now used in a wide variety of disciplines such as turbulence, image processing, signal analysis, data compression, oceanography, and process identification and control. The key idea of POD is to compute a set of orthonormal functions, called POD basis functions, such that they can describe the dynamic system under consideration with as few basis functions as possible. The set of basis functions is optimal in the sense that it captures and describes the dynamic behavior of the system with minimum number of basis functions. Given an ensemble of observations defined in a vector space, we seek to find a subspace, much smaller than the original vector space, such that the projection of all the observations on to that is maximal. The attractiveness of the POD technique lies in the fact that the basis functions are derived from the numerical solutions or experimental measurements of the system, thus exhibiting a local characteristic and ensuring

that such a basis set inherently describes the dynamics in the best possible manner by being closely linked to the accurate numerical solution of the system.

In particular, we use *method of snapshots* in this work to generate POD basis functions [175]. POD-based model reduction begins with the collection of snapshot sets which consist of solutions of the PDEs at several time instants during the evolution of the system. These snapshot sets are obtained by solving a rigorous, large-dimensional system obtained after spatial discretization (and temporal also in some cases) of the PDEs. The determination of these sets is crucial to the effectiveness of POD-based reduced-order modeling. Hence, they must contain sufficient information to accurately represent the dynamics of the system. One then uses the set of snapshots to determine a POD basis set which can accurately capture the information contained in the snapshots using a much smaller set of basis functions.

Let the snapshot set (solution of PDEs) be given as

$$\mathcal{Y} = \{y^1, \dots, y^{N_t}\} \quad (6.1)$$

with the fields $y^j = y(x, t_j)$, where N_t is the number of snapshots and N_x is the number of spatial discretization nodes. Here columns $\{\mathcal{Y}_{:,j}\}_{j=1}^{N_t}$, known as snapshots, are the spatial profiles of the state variable evaluated at time t_j . Similarly, rows $\{\mathcal{Y}_{i,:}\}_{i=1}^{N_x}$ are the time trajectories of the state variable evaluated at spatial location x_i . Consequently, $(1/N_t) \sum_{j=1}^{N_t} \mathcal{Y}_{i,j}$ is the time-averaged mean of the trajectory at location x_i . POD procedure computes an orthonormal set of basis functions $\{\phi_1, \dots, \phi_{N_x}\}$ which maximizes projection of each snapshot on to the first $M \leq N_x$ basis functions. In other words, it solves

$$\max_{\phi_1, \dots, \phi_M} \sum_{i=1}^M \sum_{j=1}^{N_t} |(y^j, \phi_i)|^2 \quad \text{s.t.} \quad \|\phi_i\| = 1, \quad (\phi_i, \phi_j)_{i \neq j} = 0 \quad i, j = 1, \dots, M \quad (6.2)$$

where $(v, w) = (v, w)_{L^2}$ denotes the L^2 -inner product with the corresponding norm $\|v\| = \|v\|_{L^2}$. Here (y^j, ϕ_i) is the projection of j^{th} snapshot on i^{th} basis function ϕ_i . Instead of maximizing a convex problem, it is reformulated such that the sum of the error between each

snapshot and its projection represented with a truncated first $M \leq N_x$ basis functions is minimized.

$$\min_{\phi_1, \dots, \phi_M} \quad \varepsilon^{POD}(M) = \sum_{j=1}^{N_t} \left\| y^j - \sum_{i=1}^M (y^j, \phi_i) \phi_i \right\|^2 = \sum_{j=1}^{N_t} \left\| \sum_{i=M+1}^{N_x} (y^j, \phi_i) \phi_i \right\|^2 \quad (6.3a)$$

$$\text{s.t.} \quad \|\phi_i\| = 1, \quad (\phi_i, \phi_j)_{i \neq j} = 0 \quad i, j = 1, \dots, M \quad (6.3b)$$

where each snapshot y^j can be represented in terms of the new basis set using its projection (y^j, ϕ_i) in the direction of ϕ_i

$$y^j = \sum_{i=1}^{N_x} (y^j, \phi_i) \phi_i, \quad j \in \{1, \dots, N_t\} \quad (6.4)$$

We choose M based on a subspace selection strategy (discussed in section 6.2.4). Usually, a significant model reduction is achieved since the value of M is much smaller compared to the value of N_x . Moreover, even with such less M , generally the POD error in projection $\varepsilon^{POD}(M)$ is significantly small. Thus, eventually we obtain a much smaller subspace spanned by a very few basis functions.

6.2.2 Optimality Property

It can be shown that POD basis functions provide an optimal basis set for the representation of the dynamics under consideration [65, 115]. Here, from a physical point of view, these first M basis functions capture more “energy” of the snapshot field than any other set of M orthonormal spatial basis functions. In other words, if we desire to represent each snapshot with exactly M orthonormal basis functions, M POD basis functions will provide the best representation and least error in projection, i.e.,

$$\varepsilon^{POD}(M) = \sum_{j=1}^{N_t} \left\| y^j - \sum_{i=1}^M (y^j, \phi_i) \phi_i \right\|^2 \leq \sum_{j=1}^{N_t} \left\| y^j - \sum_{i=1}^M (y^j, \psi_i) \psi_i \right\|^2 \quad (6.5)$$

for any other orthonormal basis $\{\psi_i\}_{i=1}^{N_x}$ and for all $M \leq N_x$.

6.2.3 POD and Singular Value Decomposition

Computation of POD basis functions is closely linked with calculating the singular value decomposition (SVD) of the snapshot matrix $\mathcal{Y} \in \mathbb{R}^{N_x \times N_t}$. Singular-value decomposition guarantees the existence of real numbers $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{N_x} \geq 0$ and orthogonal matrices

$$\begin{aligned} U &\in \mathbb{R}^{N_x \times N_x}, & \text{with columns } \{u_i\}_{i=1}^{N_x}, \\ V &\in \mathbb{R}^{N_t \times N_t}, & \text{with columns } \{v_j\}_{j=1}^{N_t}, \end{aligned}$$

such that $U^T \mathcal{Y} V = D$, where $D \in \mathbb{R}^{N_x \times N_t}$ is a diagonal matrix with diagonal entries being $\sigma_1, \dots, \sigma_{N_x}$, called singular values of \mathcal{Y} . Here, vectors $\{u_i\}_{i=1}^{N_x}$ and $\{v_j\}_{j=1}^{N_t}$ are the eigenvectors of $\mathcal{Y} \mathcal{Y}^T$ and $\mathcal{Y}^T \mathcal{Y}$, respectively, while $\sigma_1^2, \dots, \sigma_{N_x}^2$ represent the eigenvalues of $\mathcal{Y} \mathcal{Y}^T$ (or $\mathcal{Y}^T \mathcal{Y}$, as both have same eigenvalues). As a consequence, we obtain the singular value decomposition of the snapshot matrix as

$$\mathcal{Y} = U D V^T \tag{6.6}$$

since U and V are orthogonal matrices.

SVD can be used to obtain a complete set of POD basis functions. To obtain POD basis set, we solve Problem (6.2) for a given value of M . A necessary optimality condition for Problem (6.2) is given by the following eigenvalue problem

$$\mathcal{Y} \mathcal{Y}^T u = \sigma^2 u \tag{6.7}$$

From SVD, it is clear that the first M vectors $\{u_i\}_{i=1}^M$, $M \leq N_x$ of orthogonal matrix U solve (6.7), and thus solve Problem (6.2) as well. Hence $\{u_i\}_{i=1}^M$ represent the desired set of POD basis functions (or basis vectors in our case of finite spatial dimension). The amount of projection of all the snapshots $\{y^j\}_{j=1}^{N_t}$ captured by each basis function $\{u_i\}_{i=1}^M$ is given by corresponding singular values $\{\sigma_i^2\}_{i=1}^M$. Since $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_M$, the first POD basis function captures maximum projection of \mathcal{Y} on the new reduced basis set, second POD basis

is second best in capturing projection of \mathcal{Y} , and so on. We note that Equation (6.6) can also be written as $\mathcal{Y} = \widehat{U}B$, with $\widehat{U} \in \mathbb{R}^{N_x \times M}$ being the reduced set of eigenvectors, and $B = DV^T \in \mathbb{R}^{M \times N_t}$. Consequently, each snapshot $\{y^j\}_{j=1}^{N_t}$ of \mathcal{Y} can be represented in terms of M linearly independent columns of \widehat{U}

$$y^j = \sum_{i=1}^M b_{i,j} u_i, \quad j = 1, \dots, N_t \quad (6.8)$$

where $b_{i,j}$ are the elements of B . Hence, it is clear that the columns of U indeed represent a set of basis functions, and eventually turn out to be the POD basis functions based on the definition. We also note that Equation (6.6) can also be written as $\widehat{U}\widehat{D} = \mathcal{Y}V$, with $\widehat{D} \in \mathbb{R}^{M \times N_t}$ and \widehat{U} as defined above. It follows that the POD basis vectors are linear combinations of the snapshots, thus ensuring that such a basis set inherently describes the dynamics of the system by being closely linked to its actual numerical solution.

6.2.4 Subspace Selection

Choice of the subspace dimension M controls the overall error in projection $\varepsilon^{POD}(M)$ in Problem (6.3). Nevertheless, the choice of M is an important and critical task, since it determines the interrelation between accuracy and dimension of the POD based reduced order models. Since the singular values of the snapshot matrix \mathcal{Y} convey the amount of projection captured, we utilize a criterion based on these singular values to choose M . Using SVD, the error in projection in Problem (6.3) is given as

$$\varepsilon^{POD}(M) = \sum_{i=M+1}^{N_x} \sigma_i^2 \quad (6.9)$$

We define *error tolerance* λ as

$$\lambda = 1 - \frac{\sum_{i=1}^M \sigma_i^2}{\sum_{i=1}^{N_x} \sigma_i^2} = \frac{\sum_{i=M+1}^{N_x} \sigma_i^2}{\sum_{i=1}^{N_x} \sigma_i^2} = \frac{\varepsilon^{POD}(M)}{\sum_{i=1}^{N_x} \sigma_i^2} = \varepsilon_{norm}^{POD}(M) \quad (6.10)$$

where $\varepsilon_{norm}^{POD}(M)$ is the normalized error in projection. A value of M is chosen such that $\varepsilon_{norm}^{POD}(M) \leq \lambda^*$ for a desired threshold error tolerance λ^* . In other words, the normalized projection error should be less than the tolerance level λ^* [175, 65, 27]. It is also known as an M -rank approximation since the rank of the matrix U after truncation, and that of the solution matrix obtained after solving the POD-based reduced-order model, is always $M \leq N_x$.

It is commonly observed that the first few singular values are significantly larger than the subsequent ones, thus representing most of the captured projection of the system. Therefore, based on the aforementioned criterion, basis functions corresponding to those smaller singular values are dropped, which eventually leads to a much smaller subspace spanned by a very few basis functions. Hence, a significant model reduction is achieved since generally the value of M is much smaller compared to the value of N_x . For instance, M can be less than 10, whereas N_x can be on the order of 100s, and the error $\varepsilon_{norm}^{POD}(M)$ can still be only 1-3%.

6.3 Reduced-order Modeling

6.3.1 Methodology

After computing POD basis functions, a reduced-order model (ROM) is derived by projecting the underlying PDEs of the system onto the corresponding POD subspace. We use a Galerkin-type projection scheme to project our set of PDEs in this work.

Let the set of hyperbolic/parabolic PDEs be given as

$$\frac{\partial y}{\partial t} = f\left(y, \frac{\partial y}{\partial x}, \frac{\partial^2 y}{\partial x^2}\right) \quad (6.11)$$

In terms of the new set of POD basis functions, the state variable $y(x, t)$ is written as

$$y(x, t) = \sum_{i=1}^M a_i(t) \phi_i(x) \quad (6.12)$$

where $\{a_i\}_{i=1}^M$ are the unknown temporal coefficients in the expansion. We solve this system of

PDEs using *method of weighted residuals* in which the inner product of the residual of PDEs with an orthonormal set of basis functions $\{\omega_i\}_{i=1}^P$ is set to zero, i.e.,

$$\int \left[\frac{\partial y}{\partial t} - f \left(y, \frac{\partial y}{\partial x}, \frac{\partial^2 y}{\partial x^2} \right) \right] \omega_i dx = 0, \quad i = 1, \dots, P \quad (6.13)$$

In particular, for *Galerkin projection*, we choose such a basis set $\{\omega_i\}_{i=1}^P$ same as the set of basis functions in terms of which the state variable is defined, i.e., POD basis functions in this case, with $P = M$. Thus, Equation (6.13) becomes

$$\int \left[\frac{\partial y}{\partial t} - f \left(y, \frac{\partial y}{\partial x}, \frac{\partial^2 y}{\partial x^2} \right) \right] \phi_i dx = 0, \quad i = 1, \dots, M \quad (6.14)$$

Substituting Equation (6.12) in the above expression, we obtain

$$\int \left[\sum_{j=1}^M \phi_j(x) \frac{da_j}{dt} - f \left(\sum_{j=1}^M a_j(t) \phi_j(x), \sum_{j=1}^M a_j(t) \frac{d\phi_j}{dx}, \sum_{j=1}^M a_j(t) \frac{d^2\phi_j}{dx^2} \right) \right] \phi_i dx = 0, \quad i = 1, \dots, M \quad (6.15)$$

Since the POD basis functions are orthonormal, we finally obtain our reduced-order model

$$\frac{da_i}{dt} = \int f \left(\sum_{j=1}^M a_j(t) \phi_j(x), \sum_{j=1}^M a_j(t) \frac{d\phi_j}{dx}, \sum_{j=1}^M a_j(t) \frac{d^2\phi_j}{dx^2} \right) \phi_i dx, \quad i = 1, \dots, M \quad (6.16)$$

In case of a finite dimensional problem, integral inner product is replaced by an L^2 inner product. It should be noted that in the final reduced-order model we obtain only M ordinary differential equations (ODEs) compared to N_x ODEs that we usually obtain after applying spatial discretization techniques such as finite difference, finite element, or finite volume. Since M is significantly less compared to N_x , we obtain a significantly low-order model compared to the one obtained after conventional spatial discretization. More importantly, in a PDE-constrained optimization problem, replacing the set of ODEs obtained after spatial discretization with the smaller set of ODEs of the reduced-order model yields a much smaller and computationally efficient optimization problem which can be solved cheaply.

6.3.2 Handling Boundary Conditions

One of the key issues in reduced-order modeling is the incorporation of boundary conditions. If the boundary conditions are homogeneous, no changes are required in the aforementioned methodologies of obtaining POD basis functions and final reduced-order model. However, incorporating non-homogeneous Dirichlet and Neumann boundary conditions is non-trivial in general. Gunzburger et al. [89] have developed a generic framework to incorporate boundary conditions in the POD-based ROMs.

For non-homogeneous Dirichlet boundary conditions, we utilize the idea of computing POD basis elements for fluctuations around the mean value of the snapshots. Given N_t snapshots, first we compute the mean value of the snapshots $\bar{y} = (1/N_t) \sum_{j=1}^{N_t} y^j$. Next, the snapshot matrix is modified as $\mathcal{Y} = \{y^1 - \bar{y}, \dots, y^{N_t} - \bar{y}\}$, and POD basis functions are computed using this modified input ensemble. This helps in projecting out the boundary condition to the mean value of the snapshot and allows POD basis functions to follow a homogeneous boundary condition. For ROM purposes, a state variable is now expressed as

$$y(x, t) = \bar{y}(x) + \sum_{i=1}^M a_i(t) \phi_i(x) \quad (6.17)$$

Let the boundary conditions be $y(0, t) = A$, $y(L, t) = B$. Since all the snapshots satisfy this, $\bar{y}(x)$ will also satisfy this, i.e., $\bar{y}(0) = A$ and $\bar{y}(L) = B$. Hence, $\phi_i(0) = \phi_i(L) = 0$, $i = 1, \dots, M$, which helps in ensuring that the boundary conditions are always satisfied by the solution obtained after integrating ROM. In some cases, especially when a boundary condition appears as a decision variable in an optimization problem, we desire explicit occurrence of the boundary condition in the expansion (6.17). Let such a boundary condition be y^b . In this case, first we subtract y^b from all the snapshots at all spatial points $\{x_i\}_{i=1}^{N_x}$. To ensure consistency for the other boundary condition, mean value of the snapshots is then computed for these modified snapshots, i.e., $\bar{y}_b = (1/N_t) \sum_{j=1}^{N_t} (y^j - y^b)$. Consequently, the snapshot matrix used to compute POD basis set becomes $\mathcal{Y} = \{y^1 - y^b - \bar{y}_b, \dots, y^{N_t} - y^b - \bar{y}_b\}$. As a result, expansion

(6.17) gets modified in the following fashion

$$y(x, t) = \bar{y}_b + y^b + \sum_{i=1}^M a_i(t) \phi_i(x) \quad (6.18)$$

To incorporate Neumann boundary conditions (homogeneous and non-homogeneous) in this work, we construct a ROM only for the interior spatial domain and boundaries are excluded. In other words, ROMs do not directly determine the solution at the boundaries. We approximate the derivative at boundaries with any finite difference scheme. Based on this approximation together with the boundary condition and the interior solution obtained after integrating ROM, solution at the boundary nodes is determined.

6.3.3 Example - Burgers Equation

We illustrate the methodology of constructing a reduced-order model with the help of an example of Burgers equation. Burgers equation represents a wave moving in time with a constant velocity. The wave doesn't retain its shape and diffuses out because of the diffusion present in the system. Thus, it closely represents the dynamic behavior observed in a PSA system. Mathematically, Burgers equation is similar to the component mass balance equation of a PSA process, and is written as [115]

$$\begin{aligned} \frac{\partial y}{\partial t} + y \frac{\partial y}{\partial x} &= \mu \frac{\partial^2 y}{\partial x^2}, \quad \mu = 0.01 \\ y(0, t) = 0, \quad y(1, t) = 0, \quad y(x, 0) &= \begin{cases} 0.5 & \text{for } 0 < x \leq 0.5 \\ 0 & \text{for } 0.5 < x < 1 \end{cases} \end{aligned} \quad (6.19)$$

While the boundary conditions are homogeneous, this peculiar initial condition represents a square wave. To construct a POD-based reduced-order model, we express $y = \sum_{i=1}^M a_i(t) \phi_i(x)$, where $\{\phi_i\}_{i=1}^M$ are the POD basis functions, and apply Galerkin projection to obtain

$$\left(\sum_{j=1}^M \phi_j \frac{da_j}{dt} + \left(\sum_{j=1}^M a_j \phi_j \right) \left(\sum_{j=1}^M a_j \frac{d\phi_j}{dx} \right) - \mu \sum_{j=1}^M a_j \frac{d^2 \phi_j}{dx^2}, \phi_i \right) = 0, \quad i = 1, \dots, M \quad (6.20)$$

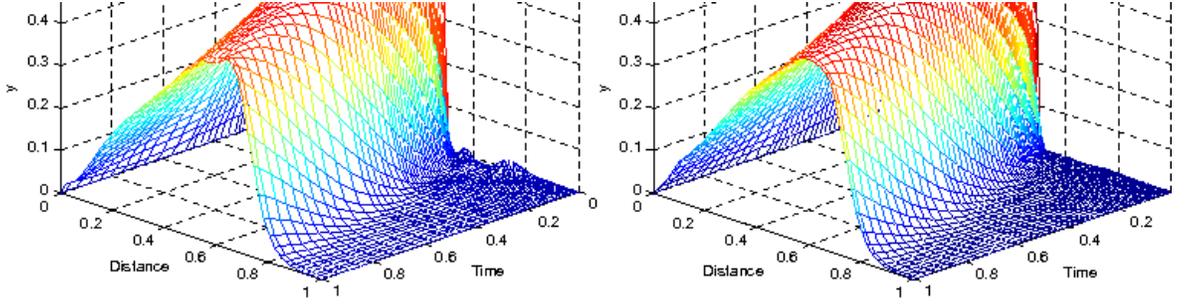


Figure 6.1: Comparison of original profile and ROM profiles for Burgers equation for varying subspace dimension

Here we use an L^2 inner product since the snapshots are obtained after spatial discretization of the PDE which lead to POD basis vectors (not basis functions). After simplification and applying orthonormality of the basis functions, we obtain our reduced-order model which is given as

$$\frac{da_i}{dt} + \left(\left(\sum_{j=1}^M a_j \phi_j \right) \left(\sum_{j=1}^M a_j \frac{d\phi_j}{dx} \right), \phi_i \right) - \mu \sum_{j=1}^M a_j \left(\frac{d^2 \phi_j}{dx^2}, \phi_i \right) = 0, \quad i = 1, \dots, M \quad (6.21)$$

Thus, the reduced-order model comprises only M ODEs.

Equation (6.19) was first discretized in space using a simple finite difference scheme with 50 spatial nodes ($N_x = 50$). The resulting set of ODEs was then integrated in MATLAB using

ode15s to obtain snapshots. With 100 time snapshots in the snapshot matrix, POD basis set was computed using SVD. Finally, ROM was constructed as in Equation (6.21) and analyzed for different values of subspace dimension M . Figure 6.1 compares the original solution profile of the Burgers equation with that obtained from ROM for different values of M . It also reports the normalized error in projection $\varepsilon_{norm}^{POD}(M)$ for corresponding M . We observe that for just 7 POD basis functions, error in projection is as low as 0.7%, and we also obtain a substantial model reduction (almost $(1/7)^{th}$ of the model obtained after spatial discretization with 50 nodes). Moreover, the solution of ROM with 7-rank approximation is almost identical to the original solution. Also, the solutions obtained with 3-rank and 5-rank approximation are fairly accurate with a little high error in projection. Hence, this illustrates the power of reduced-order modeling to predict dynamic behavior of the system with significant accuracy and with the help of a model quite small in order.

6.4 ROM-based Optimization

As mentioned before, reduced-order modeling offers computational advantages for PDE-constrained optimization problems. Since the number of DAEs in a ROM is much smaller compared to the number of DAEs obtained after spatial discretization of PDEs, replacing the latter with the former yields a much smaller and computationally-efficient optimization problem which can be solved cheaply. Mathematically, if the original optimization problem is given as

$$\min \quad \Phi(y(t_f), z(t_f), p) \quad (6.22a)$$

$$\text{s.t.} \quad \frac{dy_i}{dt} = f(y(t), z(t), p) \quad i = 1, \dots, N_x \quad (6.22b)$$

$$s(y(t), z(t), p) = 0 \quad (6.22c)$$

$$g(y(t), z(t), p) \leq 0 \quad (6.22d)$$

$$b_L \leq p \leq b_U \quad (6.22e)$$

with differential variables $y(t)$, algebraic variables $z(t)$, decision variables p with lower and upper bounds b_L and b_U , respectively, and N_x ODEs obtained after spatial discretization, the

corresponding ROM-based optimization problem can be stated as

$$\min \quad \Phi(a(t_f), p) \quad (6.23a)$$

$$\text{s.t.} \quad \frac{da_i}{dt} = f(a(t), p) \quad i = 1, \dots, M \quad (6.23b)$$

$$s(a(t), p) = 0 \quad (6.23c)$$

$$g(a(t), p) \leq 0 \quad (6.23d)$$

$$b_L \leq p \leq b_U \quad (6.23e)$$

Here $a_i(t)$ are the unknown temporal coefficients from Equation (6.12). Equation (6.23b) replaces Equation (6.22b), thus yielding a smaller optimization problem.

To solve Problem (6.23), we discretize DAEs in time and convert it into a standard nonlinear programming problem (NLP) which can be solved using state-of-the-art NLP solvers such as IPOPT. With the superstructure optimization case studies in chapters 4 and 5, we observed that such a strategy of converting PDE-constrained optimization problem to a standard NLP leads to a very large set of algebraic equations and prohibitively expensive optimization problem due to a large number of spatially discretized nodes required to capture steep adsorption fronts. Thus, we considered fewer spatial finite volumes to solve the NLP in a reasonable amount of time, and compromised on the accuracy. However, Problem (6.23) doesn't present such an issue since the DAE set is obtained after projecting PDEs onto the POD subspace, and thus is quite small in size. Moreover, even after considering many temporal finite elements to ensure satisfactory temporal accuracy, the size of the resulting NLP remains manageable. Although a large number of spatial finite volumes are required to obtain snapshots using method of lines to obtain POD subspace, such computation is done just once and remains outside the optimization problem (6.23).

A major issue with ROM-based optimization and using a ROM for Problem (6.22) is that although a ROM is substantially accurate for the values of the decision variables at which it is constructed (we call it "root-point"), it loses its accuracy at a different point in the decision variable space since the snapshots at the root-point do not capture the spatial behavior and dynamics of the system at any other point in the decision variable space. Moreover, the error

in the solution given by the ROM increases as we go further away from the root-point. Therefore, it is invalid to use a particular ROM for the optimization problem (6.22), i.e., Problem (6.23) cannot be defined for the entire decision variable space. However, we assume a reasonable accuracy for the ROM in a confidence region (or “trust-region”) around root-point, and write Problem (6.23) only for that trust-region to benefit from the computational advantage offered by ROMs. Hence, we define tighter bounds on decision variables in Problem (6.23) to ensure algorithm remains close to the root-point. Computationally-efficient optimization is then performed using ROM and the optimal solution obtained becomes the new root-point where ROM can be updated with new snapshots. Problem (6.23) is then solved again with a new trust-region around this new root-point. A systematic adaptive scheme based on such a repetitive strategy will be developed in the next chapter. In the subsequent sections, we illustrate ROM-based optimization within a neighborhood of the root-point with the help of an example of the hydrogen PSA process. Since this forms a key step in the adaptive optimization algorithm, we explore it in detail.

6.5 Case Study - Hydrogen PSA

6.5.1 PSA Process and Model Equations

We consider a 2-bed 4-step hydrogen PSA process which extracts hydrogen from a feed mixture comprising 30% hydrogen and 70% methane. In particular, the process is a Skarstrom cycle shown in Figure 2.3, and described in detail in section 2.4.1. The target process for this case study is bench-scale as described in Ko et al. [110, 111]. Design specifications and simulation conditions are listed in Table 6.1. We make following assumptions to develop a mathematical model for this process

1. All of the gases follow the ideal gas law.
2. There are no radial variations in temperature, pressure and concentrations of the gases in the solid and the gas phase.

Table 6.1: Physical properties and basic simulation parameters

parameter	value
Bed Length (L)	1 m
Bed porosity (ϵ_b)	0.404
Bed radius (R_b)	0.25 m
Particle radius (R_p)	5.41×10^{-3} m
Particle porosity (ϵ_p)	0.546
Diffusivity (D_x)	1.3×10^{-5} m ² /sec
Particle density (ρ_p)	716.3 kg/m ³
Bed density (ρ_b)	426.7 kg/m ³
Thermal diffusivity (K_L)	1.2×10^{-6} J/m/sec/K
Heat capacity of solid (C_{ps})	1046.7 J/kg/K
Heat transfer coefficient (h)	60 J/m ² /sec/K
Lumped mass transfer coefficient (k)	(0.136,0.259)(CH ₄ ,H ₂) 1/s
Heat of adsorption (ΔH)	(24124,8420)(CH ₄ ,H ₂) J/mole
Gas viscosity (μ)	3.73×10^{-8} kg/m/sec
R	8.314 J/mol/K
Ambient temperature (T_w)	300 K
Feed temperature (T_{feed})	310 K
Feed composition	(0.7,0.3)(CH ₄ ,H ₂)
Feed pressure (P_{feed})	600 kPa
Purge pressure (P_{purge})	150 kPa
Pressurization time (t_p)	5 s
Adsorption time (t_a)	50 s

3. The gas and the solid phases are in thermal equilibrium and bulk density of the solid phase remains constant.
4. Pressure drop along the bed is calculated by the Ergun equation.
5. The adsorption behaviors are described by the single-site Langmuir isotherm.
6. The adsorption rate is approximated by the linear driving force (LDF) expression.
7. A linear profile is assumed for superficial gas velocity for all the steps. Cruz et al. [58] suggested that this kind of an assumption is valid for bench-scale PSA processes and an overall mass balance is not required.

Based on the above assumptions, the mathematical model for the PSA process is listed in Table 6.2. Here we use a lumped mass transfer coefficient for the LDF model. The temperature

Table 6.2: Model equations of hydrogen PSA

Component mass balance

$$-D_L \left\{ \frac{\partial^2 y_1}{\partial x^2} - 2 \left(\frac{1}{T} \right) \left(\frac{\partial y_1}{\partial x} \right) \left(\frac{\partial T}{\partial x} \right) + 2 \left(\frac{1}{P} \right) \left(\frac{\partial y_1}{\partial x} \right) \left(\frac{\partial P}{\partial x} \right) \right\} + \frac{\partial y_1}{\partial t} + u \frac{\partial y_1}{\partial x} + \frac{RT}{P} \frac{1 - \epsilon_b}{\epsilon_b} \rho_s \left(\frac{\partial q_1}{\partial t} - y_1 \sum_{i=1}^2 \frac{\partial q_i}{\partial t} \right) = 0 \quad (6.24)$$

$$y_2 = 1 - y_1 \quad 1 = CH_4 \quad 2 = H_2 \quad (6.25)$$

Ergun equation

$$-\frac{\partial P}{\partial x} = \frac{150\mu (1 - \epsilon_b)^2}{4R_p^2 \epsilon_b^3} u + \frac{1.75\rho_g}{2R_p} \left(\frac{1 - \epsilon_b}{\epsilon_b} \right) u^2 \quad (6.26)$$

LDF equation

$$\frac{\partial q_i}{\partial t} = k_i (q_i^* - q_i) \quad i = 1, 2 \quad (6.27)$$

Energy balance

$$(\epsilon_b \rho_g C_{pg} + \rho_b C_{ps}) \frac{\partial T}{\partial t} + \rho_g C_{pg} \epsilon_b u \frac{\partial T}{\partial x} - K_L \frac{\partial^2 T}{\partial x^2} - \rho_b \sum_{i=1}^2 \left(\Delta H_i^{ads} \frac{\partial q_i}{\partial t} \right) + \frac{4h_w}{D} (T - T_w) = 0 \quad (6.28)$$

$$\rho_g = \frac{P}{RT} \sum_{i=1}^2 y_i M_w^i$$

$$C_{pg} = \sum_{i=1}^2 y_i C_{pg}^i$$

$$C_{pg}^i = a_c^i + b_c^i T + c_c^i T^2 + d_c^i T^3 \quad i = 1, 2$$

Langmuir isotherm

$$q_i^* = \frac{a_i y_i P}{1 + \sum_{i=1}^2 b_i y_i P} \quad a_i = \alpha_{1i} e^{\alpha_{2i} T} \quad b_i = \beta_{1i} e^{\beta_{2i} T} \quad i = 1, 2 \quad (6.29)$$

Linear velocity profile

$$u = u_L (x/L) + u_0 (L - x)/L \quad (6.30)$$

Cyclic steady state

$$z(t_0) = z(t_{cycle}) \quad z : y_i, q_i, T \quad \forall i \quad (6.31)$$

dependent adsorption isotherm parameters for hydrogen and methane on activated carbon (α_1, α_2 and β_1, β_2) are listed in Table 6.3 [108]. As suggested by Equations (6.24) and (6.25), PDE for the component mass balance is solved only for methane and mole fraction of hydrogen is evaluated by ensuring that the mole fractions sum up to one. We enforce this summation because the overall mass balance, which implicitly ensures such a summation, is not taken into account for superficial velocity calculation. We denote this model in Table 6.2 as the rigorous model for which we develop a reduced-order model.

Tables 6.4 and 6.5 show the equations for molar flux variables, to calculate purities and recoveries, and the boundary conditions for each operating step, respectively. Based on the molar flux variables, purities and recoveries of hydrogen and methane are given by

$$\text{purity}_{H_2} = \frac{\int O_{H_2}(t) dt}{\int O_{H_2}(t) + O_{CH_4} dt} \quad (6.32)$$

$$\text{purity}_{CH_4} = \frac{\int Hp_{CH_4}(t) dt}{\int Hp_{H_2}(t) + Hp_{CH_4} dt} \quad (6.33)$$

$$\text{recovery}_{H_2} = \frac{\int O_{H_2}(t) dt - \int Pg_{H_2}(t) dt}{\int F_{H_2}(t) dt} \quad (6.34)$$

$$\text{recovery}_{CH_4} = \frac{\int Hp_{CH_4}(t) dt}{\int F_{CH_4}(t) dt} \quad (6.35)$$

6.5.2 Reduced-order Model

We use the method of lines approach to convert PDAEs in Table 6.2 into a set of DAEs after spatial discretization, which is then simulated to generate snapshots for POD basis functions. For spatial discretization, we use an upwind-based finite volume scheme with Van Leer flux limiter for mole fraction and temperature in order to introduce additional numerical dispersion around steep adsorption fronts (*cf.* section 2.5.1). We utilize the *Multibed* approach and simulate both beds simultaneously for half the cycle (*cf.* section 2.4.3).

Table 6.3: Isotherm parameters for H₂ and CH₄ on activated carbon [108]

	α_1	α_2	β_1	β_2
Methane	0.0086	-0.2155	0.0004066	-0.010604
Hydrogen	-0.0000379	-0.01815	2.2998	-0.05993

Table 6.4: Molar flux variables for each operating step

pressurization	adsorption	depressurization	desorption
$F_j(t) = \frac{uPy_j}{RT} \Big _{x=0}$	$F_j(t) = \frac{uPy_j}{RT} \Big _{x=0}$	$F_j(t) = 0$	$F_j(t) = 0$
$O_j(t) = 0$	$O_j(t) = \frac{uPy_j}{RT} \Big _{x=L}$	$O_j(t) = 0$	$O_j(t) = 0$
$Hp_j(t) = 0$	$Hp_j(t) = 0$	$Hp_j(t) = -\frac{uPy_j}{RT} \Big _{x=0}$	$Hp_j(t) = -\frac{uPy_j}{RT} \Big _{x=0}$
$Pg_j(t) = 0$	$Pg_j(t) = 0$	$Pg_j(t) = 0$	$Pg_j(t) = -\frac{uPy_j}{RT} \Big _{x=L}$

Table 6.5: Boundary conditions for each operating step

pressurization	adsorption	depressurization	desorption
$y_j _{x=0} = y_{f,j}$	$y_j _{x=0} = y_{f,j}$	$\frac{\partial y_j}{\partial x} \Big _{x=0} = 0$	$\frac{\partial y_j}{\partial x} \Big _{x=0} = 0$
$\frac{\partial y_j}{\partial x} \Big _{x=L} = 0$	$\frac{\partial y_j}{\partial x} \Big _{x=L} = 0$	$\frac{\partial y_j}{\partial x} \Big _{x=L} = 0$	$y_j _{x=L} = y_j _{x=L}$ of other bed
$P _{x=0} = P_{high}$	$P _{x=0} = P_{high}$	$P _{x=0} = P_{low}$	$P _{x=0} = P_{low}$
$T _{x=0} = T_{feed}$	$T _{x=0} = T_{feed}$	$\frac{\partial T}{\partial x} \Big _{x=0} = 0$	$\frac{\partial T}{\partial x} \Big _{x=0} = 0$
$\frac{\partial T}{\partial x} \Big _{x=L} = 0$			
$u _{x=0} = u_{feed}$	$u _{x=0} = u_{feed}$	$u _{x=0} = -u_{reg}$	$u _{x=0} = -u_{reg}$
$u _{x=L} = 0$	$u _{x=L} = 0.125u_{feed}$	$u _{x=L} = 0$	$u _{x=L} = -0.6u_{reg}$

The snapshots are generated only after the 2-bed system attains cyclic steady state. To achieve CSS, we use a successive substitution method in which first the PSA cycle is simulated with random initial conditions, and then a series of simulations are performed with initial conditions of each new cycle taken from the final condition of the previous cycle. This is executed successively until the bed conditions do not change from cycle to cycle. Since a bench-scale PSA model is considered in this case, we achieve cyclic steady state after 100-120 cycles.

After obtaining snapshots, separate POD basis functions are generated for pressurization, adsorption, depressurization and desorption steps. Moreover, we derive separate POD basis functions for gas phase mole fractions, solid phase loadings, temperature, and pressure. Similar to Equation (6.17), these state variables are then expressed in terms of the corresponding POD basis as below

$$\begin{aligned} y_i^R(x, t) &= y_{0i}(x) + \sum_{j=1}^M a_{y_{ij}}(t) \phi_{y_{ij}}(x) & q_i^R(x, t) &= q_{0i}(x) + \sum_{j=1}^M a_{q_{ij}}(t) \phi_{q_{ij}}(x) \\ T^R(x, t) &= T_0(x) + \sum_{j=1}^M a_{T_j}(t) \phi_{T_j}(x) & P^R(x, t) &= P_0(x) + \sum_{j=1}^M a_{P_j}(t) \phi_{P_j}(x) \end{aligned}$$

Here $y_{0i}(x)$, $q_{0i}(x)$, $T_0(x)$, and $P_0(x)$ are snapshot averages for mole fraction, solid loading, temperature, and pressure, respectively. Gas density, specific heat and equilibrium solid concentrations are calculated explicitly in terms of y_i^R , T^R , and P^R . Table 6.6 shows DAEs of the reduced-order model obtained after Galerkin projection of the model in Table 6.2 on to these POD basis functions.

Since the snapshots are obtained after the rigorous model achieves cyclic steady state, snapshots for both beds for the corresponding steps are identical. For instance, snapshots collected during pressurization step for bed 1 are identical to the snapshots of the pressurization step for bed 2. Therefore, we construct reduced-order model for only one bed and the other bed is ignored. Coupling for the adsorption/desorption steps of the two beds is ensured by the adsorption and desorption steps of the same bed. Hence, a greater model reduction is achieved

Table 6.6: ROM for the hydrogen PSA process

Component mass balance

$$\begin{aligned}
& -D_L \left(\left(\frac{d^2 y_{0i}}{dx^2} + \sum_{j=1}^M a_{yij} \frac{d^2 \phi_{yij}}{dx^2}, \phi_{yik} \right) + \left(\frac{d}{dx} \left(\frac{2}{TR} \right) \left(\frac{dy_{0i}}{dx} + \sum_{j=1}^M a_{yij} \frac{d\phi_{yij}}{dx} \right), \phi_{yik} \right) \right) \\
& + R\rho_s \left(\frac{1 - \epsilon_b}{\epsilon_b} \right) \left(\sum_{j=1}^M \left(\frac{TR \phi_{qij}}{PR}, \phi_{yik} \right) \frac{da_{qij}}{dt} - \sum_{r=1}^2 \sum_{j=1}^M \left(\frac{TR y_i^R \phi_{qrj}}{PR}, \phi_{yik} \right) \frac{da_{qrj}}{dt} \right) \\
& + \frac{da_{yik}}{dt} + \left(u \left(\frac{dy_{0i}}{dx} + \sum_{j=1}^M a_{yij} \frac{d\phi_{yij}}{dx} \right), \phi_{yik} \right) = 0 \quad \forall k \in [1, M]
\end{aligned} \tag{6.36}$$

Ergun equation

$$\begin{aligned}
& \frac{150\mu (1 - \epsilon_b)^2}{4R_p^2 \epsilon_b^3} (u, \phi_{Pk}) + \frac{1.75\rho_g}{2R_p} \left(\frac{1 - \epsilon_b}{\epsilon_b} \right) (u^2, \phi_{Pk}) \\
& + \left(\frac{dP_0}{dx} + \sum_{j=1}^M a_{Pj} \frac{d\phi_{Pj}}{dx}, \phi_{Pk} \right) = 0 \quad \forall k \in [1, M]
\end{aligned} \tag{6.37}$$

LDF equation

$$\frac{da_{qik}}{dt} = k_i \left(\sum_{j=1}^M (q_i^{R*}, \phi_{qik}) - a_{qik} - (q_{0i}, \phi_{qik}) \right) \quad \forall k \in [1, M] \tag{6.38}$$

Energy balance

$$\begin{aligned}
& \epsilon_b \sum_{j=1}^M \rho_g^R C_{pg}^R (\phi_{Tk}, \phi_{Tj}) \frac{dT_j}{dt} + \left(\epsilon_b u \rho_g^R C_{pg}^R \left(\frac{dT_0}{dx} + \sum_{j=1}^M a_{Tj} \frac{d\phi_{Tj}}{dx} \right), \phi_{Tk} \right) \\
& + \rho_b C_{ps} \frac{dT_k}{dt} - \rho_b \sum_{r=1}^2 \left(\Delta H_r^{ads} \sum_{j=1}^M (\phi_{qrj}, \phi_{Tk}) \frac{da_{qrj}}{dt} \right) \\
& - K_L \left(\frac{d^2 T_0}{dx^2} + \sum_{j=1}^M a_{Tj} \frac{d^2 \phi_{Tj}}{dx^2}, \phi_{Tk} \right) + \frac{4h_w}{D} (T^R - T_w, \phi_{Tk}) = 0 \quad \forall k \in [1, M]
\end{aligned} \tag{6.39}$$

Langmuir isotherm

$$q_i^{R*} = \frac{a_i^R y_i^R PR}{1 + \sum_{i=1}^2 b_i^R y_i^R PR} \quad \begin{aligned} a_i^R &= \alpha_{1i} e^{\alpha_{2i} T^R} \\ b_i^R &= \beta_{1i} e^{\beta_{2i} T^R} \end{aligned} \tag{6.40}$$

Cyclic steady state

$$a_{z,k}(0) \Big|_{pres} = \left(\sum_{j=1}^M a_{z,j}(t_{cycle}) \Big|_{des} \phi_{z,j} \Big|_{des}, \phi_{z,k} \Big|_{pres} \right) \quad \forall k \in [1, M], z : y_i, q_i, T \tag{6.41}$$

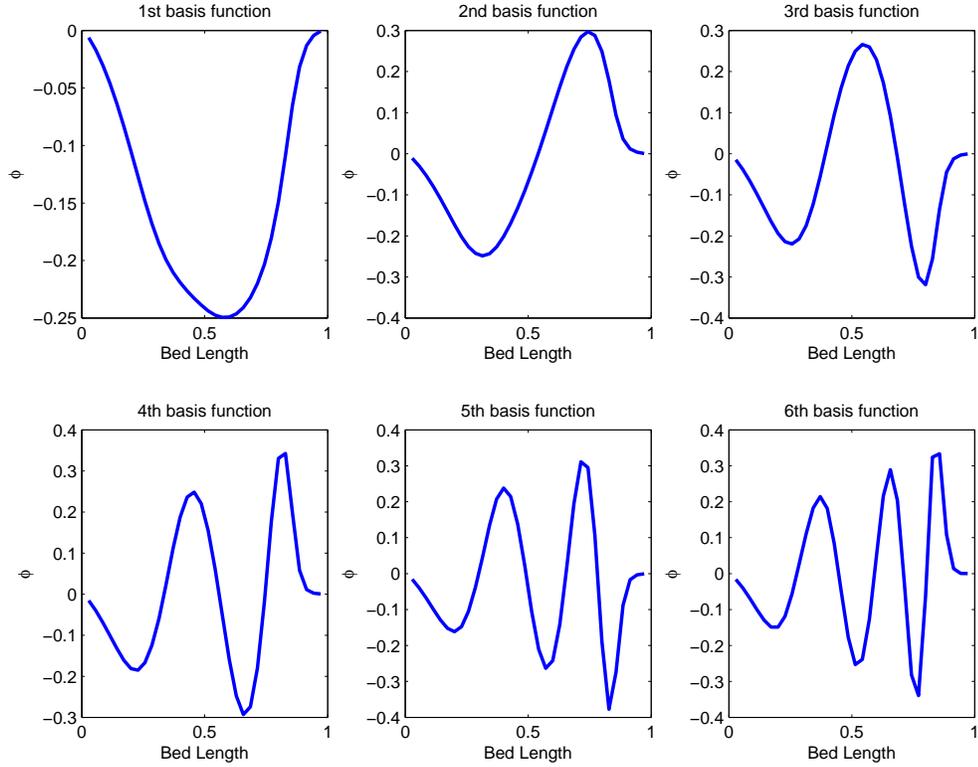


Figure 6.2: First six POD basis functions of methane mole fraction for adsorption

since ROM comprises DAEs of only one bed, while the rigorous model is simulated for both beds simultaneously using the Multibed approach.

6.5.3 Comparison of Rigorous Model and ROM at CSS

With 35 spatial finite volumes, we convert PDAEs in Table 6.2 to DAEs and integrate using *ode15s* in MATLAB. Cyclic steady state is achieved up to the desired tolerance after simulating the model repeatedly for 120 cycles, and snapshots are collected to generate empirical POD basis. Figure 6.2 shows POD basis functions of the gas-phase methane mole fraction for the adsorption step. Since these functions are empirical, their shapes are different for all four operating steps. Figure 6.3 shows first 10 singular values for the gas-phase methane mole fraction and temperature for all operating steps. Slopes of the curves show that the singular values decay sharply and the first 5-6 values can accurately capture the dynamic behavior with ε_{norm}^{POD} as low as 0.1%. Thus, we infer that we can represent all the state variables for all the

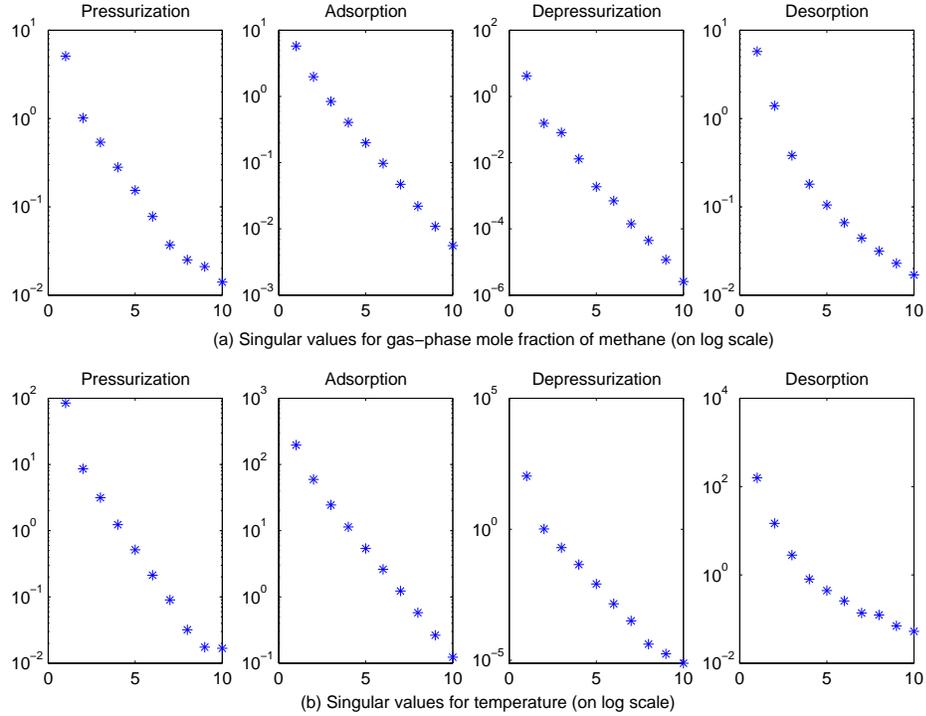


Figure 6.3: Singular values of gas-phase mole fraction of methane and temperature

operating steps with only 5-6 spatial POD basis functions, instead of 35 spatial volumes. With 35 spatial volumes, the rigorous model comprises a total of 2800 DAEs (including both beds and all four operating steps), while with 5 basis functions the ROM obtained after Galerkin projection comprises a mere 200 DAEs which is $1/14^{th}$ of the rigorous system. Hence, we observe a significant model reduction.

We discretize DAEs of the reduced-order model for all the operating steps in time with 30 finite elements, and the resulting algebraic equations are solved simultaneously in AMPL using IPOPT. Instead of solving the discretized ROM repeatedly for CSS, we consider initial conditions as decision variables and reduced CSS conditions (shown in Table 6.6) are solved simultaneously with the model equations in AMPL. Figure 6.4 compares profiles of gas-phase methane mole fraction obtained for the rigorous model and ROM with 5 POD basis functions. We observe that 5-rank approximation captures the dynamics of the problem very well and profiles are nearly identical. Temperature profiles in Figure 6.5 also depict such similarity. We note that all the profiles plotted are at CSS. We also observe that the mole fraction profiles

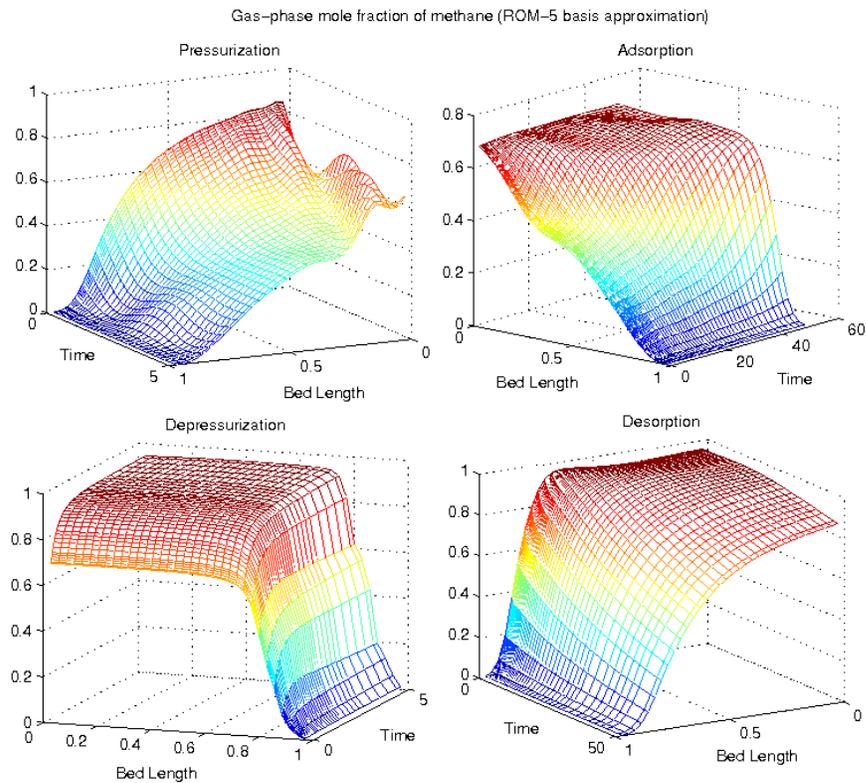
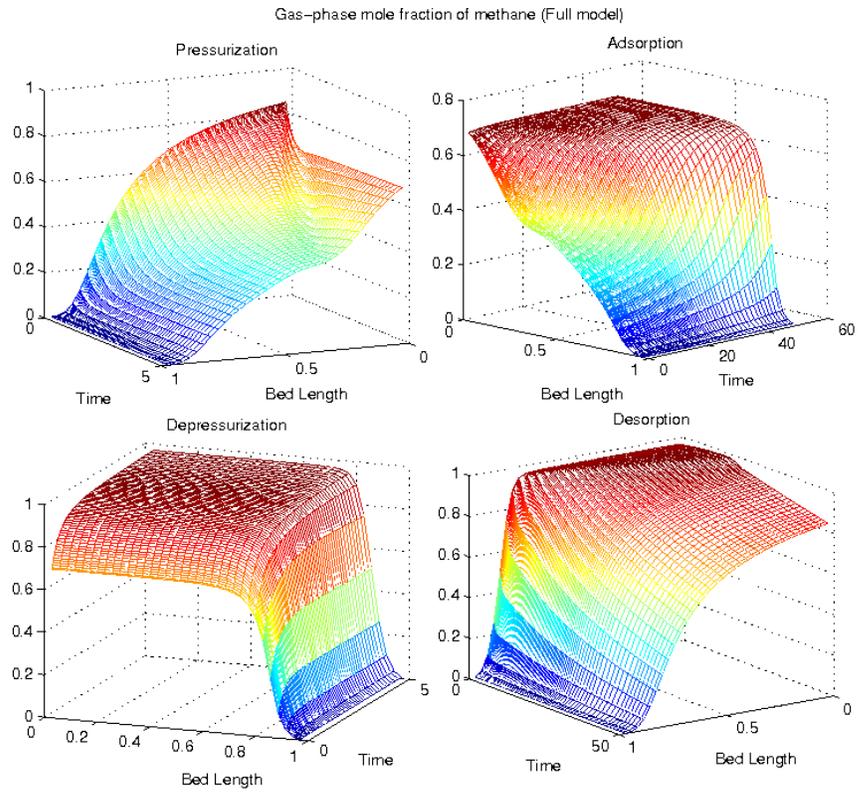
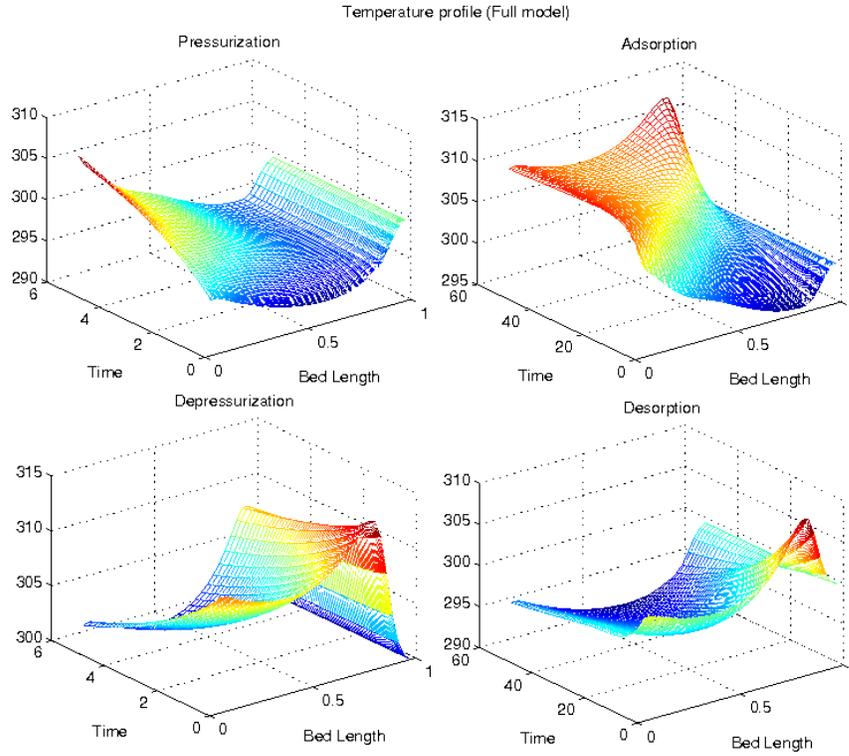
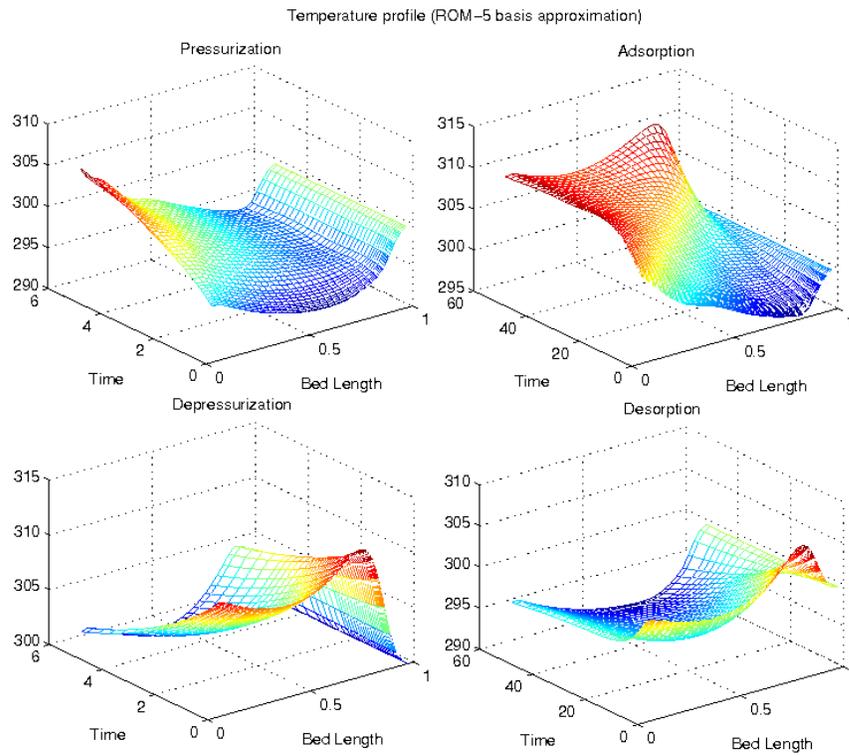


Figure 6.4: Comparison of methane mole fraction for all the steps after CSS



(a) Rigorous model simulation in MATLAB



(b) ROM simulation in AMPL

Figure 6.5: Comparison of temperature profiles for all the steps after CSS

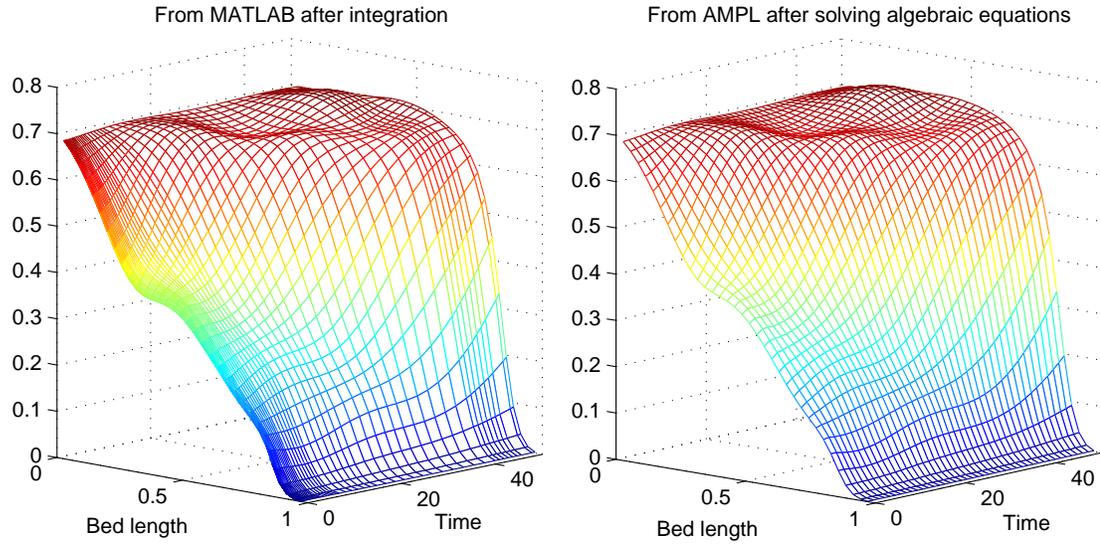


Figure 6.6: Comparison of methane mole fraction profile for adsorption step obtained after integrating in MATLAB and solving simultaneously in AMPL

Table 6.7: Comparison of rigorous model and ROM based on the performance variables

Performance variables	Rigorous model	ROM
H ₂ purity	0.9987	0.9987
H ₂ recovery	0.1095	0.1094
CH ₄ purity	0.9421	0.9425
CH ₄ recovery	0.2091	0.2094

are quite steep in the spatial dimension, especially for the adsorption and depressurization steps, and ROM effectively captures such steep behavior, besides adequately capturing system's behavior in the temporal dimension.

To check for the accuracy of the temporal discretization in AMPL, we integrate the DAE system of the ROM in MATLAB as well as solve it in AMPL without the CSS conditions. Figure 6.6 compares the gas-phase methane mole fraction profile for the adsorption step for both approaches. Clearly, the profiles compare very well and show that the additional errors introduced by a pre-determined temporal discretization in AMPL are negligible for this system of equations.

Table 6.7 compares the purity and recovery of hydrogen and methane obtained from the rigorous model as well as the reduced-order model at CSS. This can be seen as another basis

to compare the accuracy of ROM. Values obtained from ROM developed with the 5-rank approximation are accurate up to 3 decimal places and can be considered identical to the true values for all practical purposes.

6.5.4 ROM-based Optimization within a Trust-region

In this section, we demonstrate how a ROM can be used to perform computationally efficient optimization within a trust-region around the root-point where ROM is constructed. In particular, we utilize the reduced-order model in Table 6.6 to maximize hydrogen recovery subject to a purity constraint and tight bounds on the decision variables which form the trust-region. As discussed before, such tight bounds are essential as ROM loses its accuracy as we go further away from the root-point.

Table 6.8 shows decision variables considered for the optimization problem and their values at which the ROM is constructed. We consider two separate cases for optimization. In the first case, we do not consider feed and regeneration velocities as decision variables, while they are included as decision variables in the second case. Optimization results for both cases are discussed further. We also perform optimization without a trust-region and demonstrate why such a bounded region is essential when ROM is used for optimization.

6.5.4.1 Case I: Optimization without velocities

With feed and regeneration velocities held fixed at their root-point values, we solve the following ROM-based optimization problem.

$$\begin{aligned}
 \max \quad & \text{H}_2 \text{ recovery (from Equation (6.34))} \\
 \text{s.t.} \quad & \text{ROM Equations (6.36)-(6.41)} \\
 & \text{H}_2 \text{ purity} \geq 0.998 \text{ (from Equation (6.32))} \\
 & 480 \text{ kPa} \leq P_H \leq 520 \text{ kPa} \\
 & 130 \text{ kPa} \leq P_L \leq 170 \text{ kPa} \\
 & 3 \leq t_p \leq 7 \\
 & 47 \leq t_a \leq 53
 \end{aligned} \tag{6.42}$$

Here bounds on decision variables define the trust-region. We discretize ROM in the temporal dimension and the resulting large-scale NLP is solved in AMPL using IPOPT. Optimal values

Table 6.8: Decision variables and the root-point at which ROM is built

Variable	Value
High operating bed pressure (P_H)	500 kPa
Low operating bed pressure (P_L)	150 kPa
Pressurization step time (t_p)	5 sec
Adsorption step time (t_a)	50 sec
Feed velocity (u_{feed})	0.1 m/sec
Regeneration velocity (u_{reg})	-0.05 m/sec

Table 6.9: Optimization results for Case I

Problem size and computational time		
No. of variables	42760	
No. of constraints	42756	
Total no. of iterations	15	
Total CPU sec.	195.44	
Optimal parameters		
High operating bed pressure (P_H)	520 kPa	
Low operating bed pressure (P_L)	130 kPa	
Pressurization step time (t_p)	3 sec	
Adsorption step time (t_a)	53 sec	
Comparison of performance variables		
	ROM (AMPL)	Rigorous model (MATLAB)
H ₂ purity	0.9988	0.9991
H ₂ recovery	0.1628	0.1629
CH ₄ purity	0.9541	0.9491
CH ₄ recovery	0.1771	0.1769

of decision variables together with the CPU time are listed in Table 6.9. Within the bounded region, we observe an increase in the recovery of hydrogen up to 16.3%. The optimum point is achieved in only 195 CPU seconds as ROM based on 5-rank approximation (200 DAEs) is used for optimization. Moreover, optimization is performed cheaply since few iterations are needed to achieve the optimum.

To validate accuracy of optimization results, we simulate the rigorous model using the method of lines approach in MATLAB at the optimal values of decision variables. Purities and recoveries obtained from the rigorous model simulation is also listed in Table 6.9. We observe that these values are reasonably close to the ones obtained after the ROM-based optimization

in AMPL. Thus, we infer that the ROM is behaving reasonably accurately at the optimum too. Since increase in the hydrogen recovery is same, ROM is not depicting much error at the optimum. Moreover, optimum is feasible since the hydrogen purity constraint is also satisfied by the rigorous model.

To further consolidate our inference, we plot gas-phase mole fraction profiles of methane in Figure 6.7 for all steps. Mole fraction profiles are generated from the rigorous model simulation at the optimum and compared with profiles obtained from the ROM after optimization in AMPL. The profiles match reasonably accurately and show that the behavior of the ROM is close to the rigorous model at the optimum.

It is worth noting that all decision variables are at their bounds after optimization. Therefore, to verify that (locally) optimal values for the ROM are also optimal for the rigorous model, we evaluate KKT conditions of the rigorous model at this point, x^* . A general rigorous model based optimization problem with its KKT conditions is given by:

$$\begin{array}{ll} \max & f(x) \\ \text{s.t.} & c(x) = 0 \\ & a \leq x \leq b \end{array} \quad \left| \quad \begin{array}{l} \nabla f(x^*) + \nabla c(x)\lambda + \mu_l - \mu_u = 0 \\ c(x^*) = 0, \quad \mu_l, \mu_u \geq 0 \\ \mu_l(x^* - a) = 0 \\ \mu_u(x^* - b) = 0 \end{array} \right.$$

By shifting the decision variables to the beginning of vector x and defining a null space basis matrix that satisfies $Z^T[I \ 0]^T = I$ and $Z^T \nabla c(x^*) = 0$, allows KKT conditions to be written as:

$$\begin{aligned} Z^T \nabla f(x^*) - \mu_u &= 0 && : \text{for decision variables at upper bound } (\mu_l = 0) \\ Z^T \nabla f(x^*) + \mu_l &= 0 && : \text{for decision variables at lower bound } (\mu_u = 0) \end{aligned}$$

Since $\mu_u, \mu_l \geq 0$, KKT conditions simplify to $(Z^T \nabla f(x^*))_i \leq 0$ if a decision variable i is at its lower bound, and $(Z^T \nabla f(x^*))_i \geq 0$ if a decision variable i is at its upper bound. We indeed

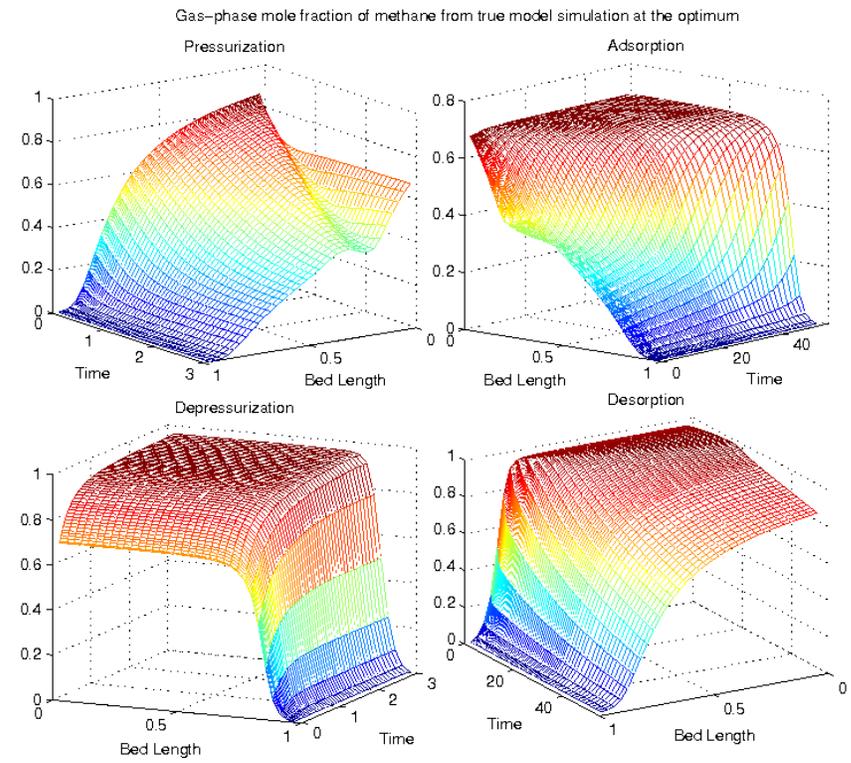
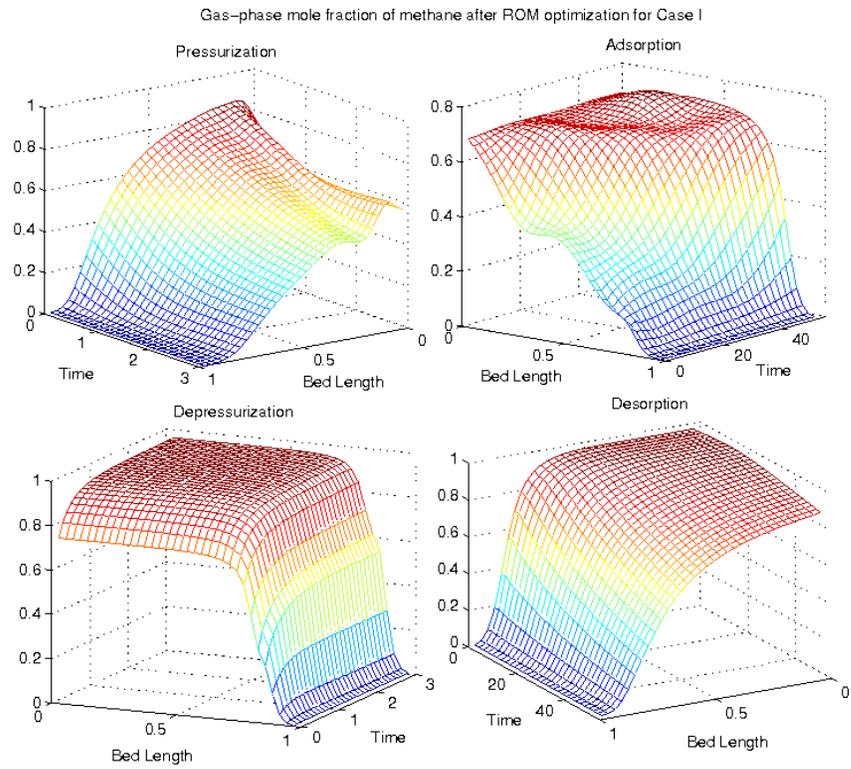


Figure 6.7: Comparison of mole fraction of methane after optimizing Case I

Table 6.10: Perturbation results

Perturbed variable	P_H	P_L	t_p	t_a
Optimal value	520	130	3	53
Perturbed value	530	125	2.5	55
H ₂ recovery(after perturbation)*	0.1675	0.1723	0.1643	0.1632

*Optimal hydrogen recovery = 0.1628

Table 6.11: Comparison of hydrogen performance for relaxed optimization

Performance variable	ROM	Rigorous Model
H ₂ recovery	0.3482	0.2763
H ₂ purity	0.9981	0.8032

verify these properties by perturbing decision variables from their optimal values. We provide a positive perturbation to variables at upper bound and negative to the ones at lower bound, and record the change in the objective function. The results are shown in Table 6.10. We observe an increase in the hydrogen recovery for all perturbations which proves optimality.

Although successful results are obtained by imposing tight bounds on decision variables in the ROM-based optimization problem, we desire to verify if such a strategy is indeed necessary. Thus, we solve Problem (6.42) with the following relaxed bounds on decision variables:

$$\begin{aligned} 300 \text{ kPa} \leq P_H \leq 1300 \text{ kPa} & \quad 0.5 \leq t_p \leq 10 \\ 100 \text{ kPa} \leq P_L \leq 250 \text{ kPa} & \quad 30 \leq t_a \leq 80 \end{aligned} \quad (6.43)$$

At the optimum, we obtain a hydrogen recovery of 34.8%. However, solution profiles obtained from the ROM are oscillatory and physically unrealistic. Figure 6.8 illustrates methane gas-phase mole fraction profiles obtained after ROM optimization with relaxed bounds. For adsorption, the oscillations are quite big and they tend to increase as step time increases. In case of depressurization, there is a jump in the profile before steep decrease along spatial dimension. Large oscillations in the profiles thus show large error in the reduced-order model at the optimum. Moreover, Table 6.11 shows that when the rigorous model is simulated to CSS at the optimal values, hydrogen purity dips to 80%, compared to 99.8% given by ROM optimization. This vindicates the use of a trust-region and the claim that tighter restrictions

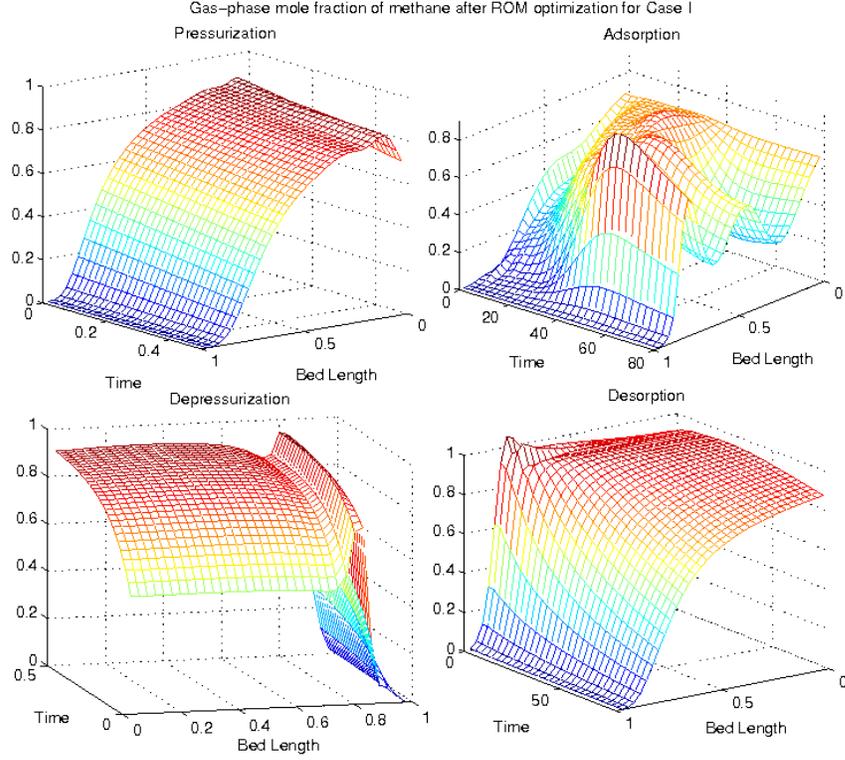


Figure 6.8: Gas-phase methane mole fraction profiles for ROM for relaxed bounds in Case I

are required on decision variables for ROM-based optimization.

6.5.4.2 Case II: Optimization with velocities

In this case, besides operating pressures and step times, we also consider feed and regeneration velocities as decision variables and solve the following ROM-based optimization problem which maximizes hydrogen recovery within a trust-region.

$$\begin{aligned}
 & \max \quad \text{H}_2 \text{ recovery (from Equation (6.34))} \\
 & \text{s.t.} \quad \text{ROM Equations (6.36)-(6.41)} \\
 & \quad \text{H}_2 \text{ purity} \geq 0.998 \text{ (from Equation (6.32))} \\
 & \quad 490 \text{ kPa} \leq P_H \leq 510 \text{ kPa} \\
 & \quad 140 \text{ kPa} \leq P_L \leq 160 \text{ kPa} \\
 & \quad 4 \leq t_p \leq 6 \\
 & \quad 49 \leq t_a \leq 51 \\
 & \quad 0.09 \leq u_{feed} \leq 0.11 \\
 & \quad -0.0505 \leq u_{reg} \leq -0.0495
 \end{aligned} \tag{6.44}$$

As a convention, flow is considered negative for the depressurization and desorption steps, and

Table 6.12: Optimization results for Case II

Problem size and computational time		
No. of variables	43040	
No. of constraints	43034	
Total no. of iterations	13	
Total CPU sec.	184.493	
Optimal parameters		
High operating bed pressure (P_H)	510 kPa	
Low operating bed pressure (P_L)	140 kPa	
Pressurization step time (t_p)	4 sec	
Adsorption step time (t_a)	51 sec	
Feed velocity (u_{feed})	0.11 m/s	
Regeneration velocity (u_{reg})	-0.0495 m/s	
Comparison of performance variables		
	ROM (AMPL)	Rigorous model (MATLAB)
H ₂ purity	0.9986	0.9974
H ₂ recovery	0.1511	0.1509
CH ₄ purity	0.9507	0.9495
CH ₄ recovery	0.1827	0.1826

thus a minus sign is used for u_{reg} . We solve the optimization problem in AMPL using IPOPT after discretizing ROM in the temporal dimension. Optimization results along with the CPU time is shown in Table 6.12.

Within the bounded region, an optimum hydrogen recovery of 15.11% is obtained. As in the previous case, optimization is computationally efficient and the problem gets solved in only 184 CPU seconds. The rigorous model is also simulated at the optimal values of the decision variables to validate optimization results from AMPL. Purities and recoveries obtained from the rigorous model simulation are also listed in Table 6.12. We observe that these values are quite close to the ones obtained after ROM-based optimization, indicating that the ROM is predicting the dynamic behavior reasonably accurately at the optimum as well. As in the previous case, we also compare gas-phase methane mole fraction profiles for the rigorous model simulation and ROM optimization in Figure 6.9. Barring small oscillations in the adsorption step, profiles match reasonably accurately, thus showing that ROM's behavior is fairly close to the dynamic behavior predicted by rigorous model at the optimum.

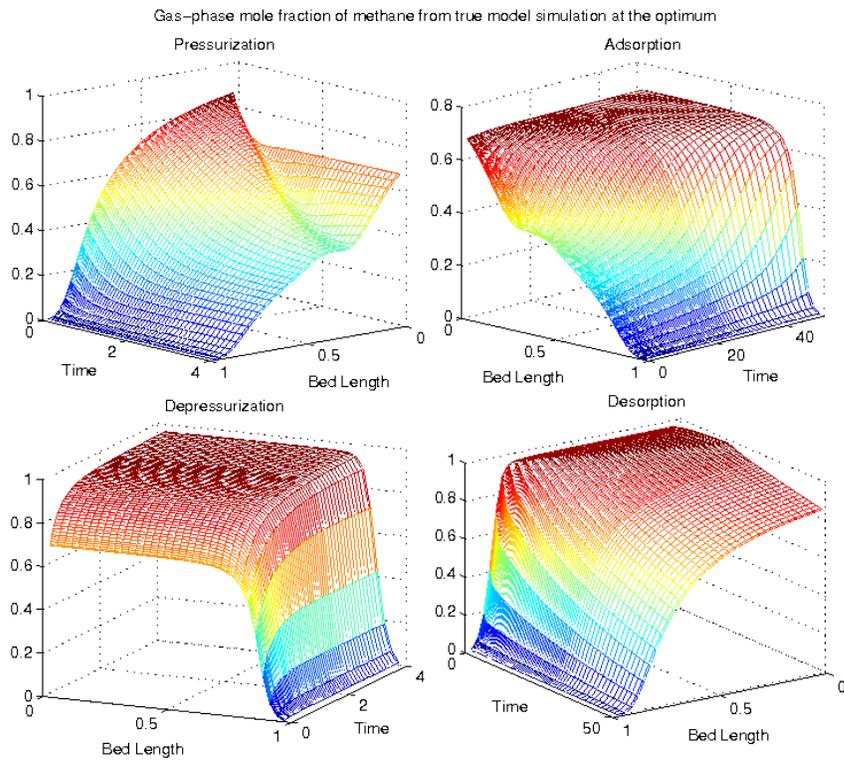
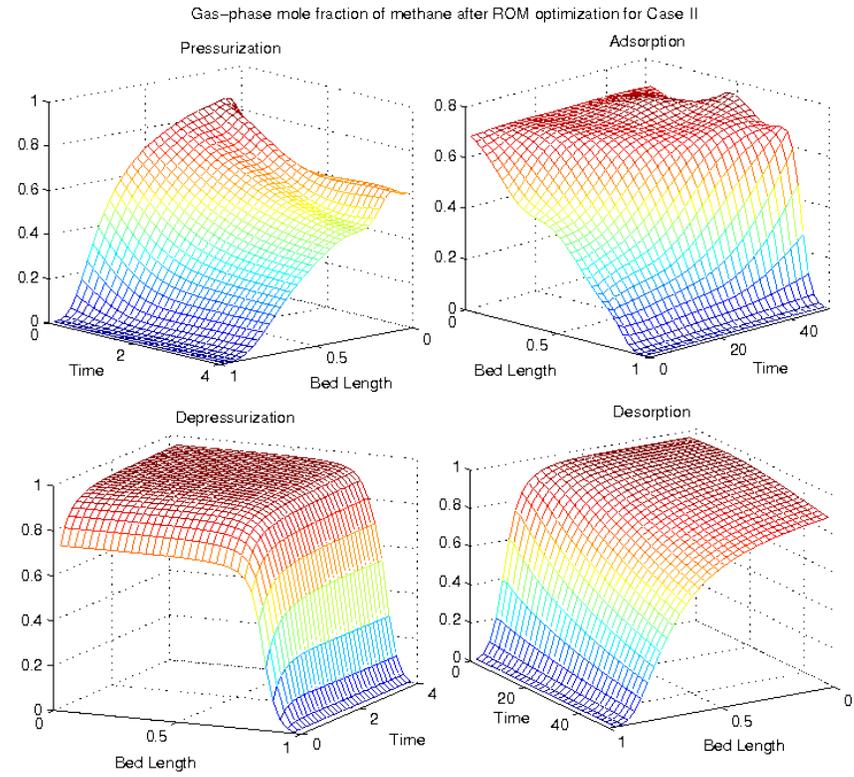


Figure 6.9: Comparison of mole fraction of methane after optimizing Case II

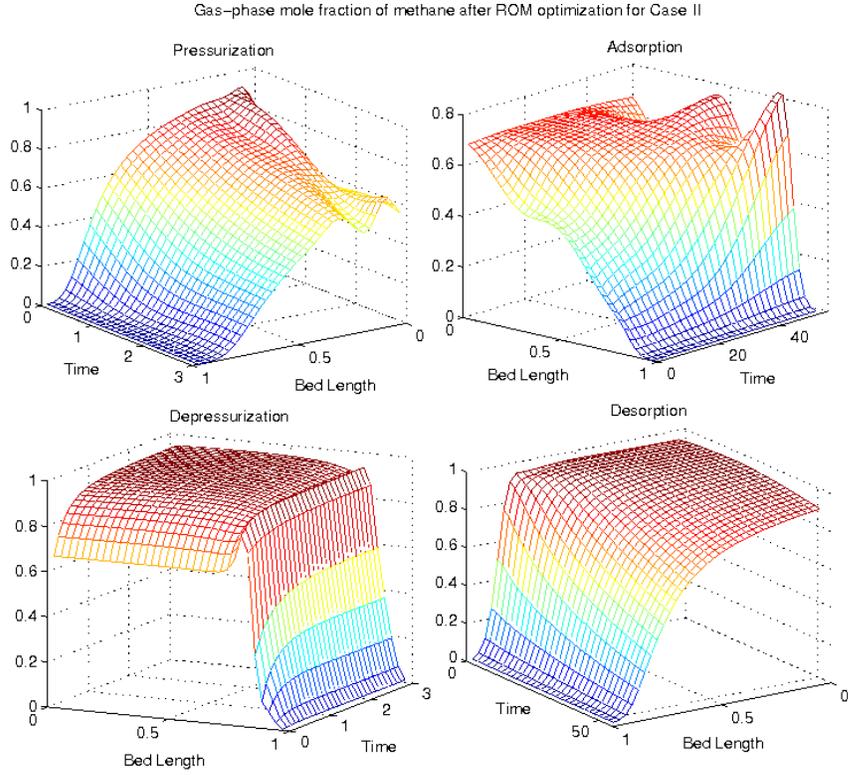


Figure 6.10: Methane mole fraction profiles for ROM for relaxed bounds in Case II

Unlike previous case, we observe that in this case the hydrogen purity constraint is violated slightly by the rigorous model at the optimum. Rigorous model gives a hydrogen purity of 99.74% which is slightly less than the desired lower bound 99.8%. Hence, we infer that attaining feasibility in a ROM-based optimization problem cannot guarantee feasibility for the original optimization problem, even with a small trust-region. A more systematic algorithm to ensure feasibility in both problems will be developed in the next chapter.

We also solve Problem (6.44) with slightly relaxed bounds as shown below:

$$\begin{aligned}
 480 \text{ kPa} \leq P_H \leq 520 \text{ kPa} & \quad 3 \leq t_p \leq 7 & \quad 0.08 \leq u_{feed} \leq 0.12 \\
 130 \text{ kPa} \leq P_L \leq 170 \text{ kPa} & \quad 47 \leq t_a \leq 53 & \quad -0.053 \leq u_{reg} \leq -0.047
 \end{aligned} \tag{6.45}$$

Within this new trust-region, hydrogen recovery could be increased to 21.6%. However, obtaining this extra increment in recovery is marred by oscillatory solution profiles in the ROM. Figure 6.10 shows the profiles of the gas-phase mole fraction of methane obtained after ROM optimization in AMPL. It can be observed that profiles are oscillatory for the adsorption and

depressurization steps, which is unphysical indicating inadequate representation of the dynamics by the ROM at the optimum. Hence, these results also suggest that tight restrictions on decision variables and an adaptive trust-region based strategy with appropriate ROM updation is required for the ROM-based optimization.

6.6 Conclusions

Beginning with a review of the previous work on model reduction, we describe how proper orthogonal decomposition can be used to develop low-order approximations to PDE-based models. In particular, with the method of snapshots, singular-value decomposition, and Galerkin's framework, POD can be successfully used to construct reduced-order models which can be orders of magnitude smaller than the original model without losing accuracy. Methodology to construct ROMs is illustrated for a Skarstrom PSA process to separate hydrogen and methane. We not only show that the reduced-order modeling technique can be successfully used for large-scale models as well, but on the basis of the comparison made between the rigorous model and the ROM, we also conclude that such ROMs can provide significant model reduction and can mimic the dynamic behavior quite accurately at the same time.

We also discuss that ROMs can be used to optimize in a confidence-region in the vicinity of the root-point assuming it is reasonably accurate in that region. Such ROM-based optimization in a trust-region is successfully demonstrated for two separate case studies of the hydrogen PSA process. ROMs accurately predict the descent direction in the trust-region and an improvement in the objective is obtained for both cases. Moreover, based on the CPU times observed, we conclude that ROMs enable fast and cheap optimization. Current results indicate that the proposed ROM-based methodology is a promising surrogate modeling technique for cost-effective optimization purposes. We vindicate that a trust-region is essential for ROM-based optimization, and conclude that an adaptive strategy with appropriate ROM updation is required. We develop such a strategy in the next chapter.

Chapter 7

Trust-region Framework for ROM-based Optimization

Synopsis

Trust-region methods provide an excellent adaptive framework for ROM-based optimization since they not only restrict the step around the root-point, but also synchronize ROM update with the information obtained during the course of optimization, thus providing a robust and globally convergent framework. In this chapter, we first develop an exact penalty-based trust-region algorithm, and develop correction schemes for the objective and the constraints to ensure global convergence with ROM-based approximate models. Algorithm with corrections is demonstrated for a two-bed four-step PSA case study for post-combustion capture. Next, highlighting drawbacks of the penalty approach and benefits of a filter, we develop a hybrid filter trust-region algorithm for constrained ROM-based optimization. Finally, the filter algorithm is illustrated with the PSA case study. In particular, we observe that both algorithms converge to a local optimum of the original optimization problem within reasonable trust-region iterations and optimization CPU time.

7.1 Introduction

From the previous chapter it is clear that a single POD-based ROM is generally reliable in a restricted zone around the point it is constructed (root-point), and it needs to be updated as the optimization proceeds from the root-point to other points in the decision space. To converge to the optimum we can a) solve the ROM-based optimization problem within tight bounds, b) take a step and construct a new ROM by generating new snapshots at this new point, and c) repeat the process until the optimum is achieved. Construction and updation of ROM should be coupled with the progress in the optimization process. Here, the computational advantage comes from the fact that the ROM is used for optimization instead of the detailed rigorous model. In order to guarantee convergence of the above process, we additionally embed it in a trust-region framework.

Trust-region methods [54, 61] are suitable and quite appropriate for ROM-based optimization since they not only ensure that the step computed by the optimizer using ROM stays reasonably close to the root-point (as demonstrated in section 6.5.4.1), but also allow ROM updation decisions based on the information obtained during the optimization procedure, and provide a robust and globally convergent framework with ROMs. With such an adaptive framework we can regulate the amount of optimization done with a ROM before we return to the detailed model to update it. By comparing the improvement predicted by the ROM to the improvement realized for the true system being optimized, we not only deduce how well ROM is predicting the behavior of the system, but also decide if it should be updated or re-used during the course of the optimization.

The idea of using a trust-region framework to manage approximate models in optimization was first introduced by Alexandrov et al. [8]. However, they developed their generic framework only for unconstrained optimization problems. They use a basic trust-region algorithm [54] which involves solving the ROM-based problem in a trust-region, taking a step if the reduction in the original objective function is reasonable compared to the one predicted by ROM-based problem, and repeating this until convergence. In order to ensure convergence to the correct

optimum, they assume *first-order consistency conditions* to be true. One of their major contributions is the introduction of scaled (or corrected) objective functions and constraints in the trust-region subproblem to enforce these consistency conditions. Later, Fahl [65] developed the TRPOD algorithm based on the Alexandrov’s approach with few modifications. In TRPOD, they relax the consistency conditions and use an inexact gradient based formulation suggested by Carter [41]. Moreover, they solve the trust-region subproblem approximately and utilize Toint’s algorithm [186] for step computation. However, Alexandrov’s correction for objective functions and constraints is essentially a part of the TRPOD. Bergmann et al. [25] also applied the TRPOD algorithm for optimal control of the circular cylinder wake flow considered in the laminar regime. Kragel [112] developed a streamline diffusion POD methodology to construct ROMs which are tuned to a high-order Navier-Stokes solver, and applied a recursive multilevel trust-region algorithm [87, 88] for an optimal flow control problem. For optimal design problems, Weickum et al. [198] developed extended ROMs for the whole design space and developed a simple trust region strategy for globalization.

The focus of all the aforementioned studies is only unconstrained optimization, and the strategies developed do not involve any discussion about the techniques to manage equality and inequality constraints. We demonstrated in section 6.5.4.2 that attaining feasibility in the ROM-based optimization problem cannot guarantee feasibility for the original optimization problem, and thus a systematic formulation to ensure feasibility is desired. Eldred et al. [63] briefly discuss few ways to handle infeasibility in a trust-region based methodology for optimization problems involving ROMs, but do not provide any systematic rigorous framework. Alexandrov et al. extended their previous work on unconstrained optimization with approximate models, and included equality and inequality constraints as well in the original optimization problem [4, 5, 6, 7, 9, 10]. In particular, they develop three distinct algorithms for constrained optimization problems involving ROMs; Augmented Lagrangian AMMO based on an augmented Lagrangian formulation, MAESTRO-AMMO which involves an ℓ_2 penalty function as a merit function, and SQP-AMMO which utilizes an exact ℓ_1 penalty function as

a merit function and a trust-region SQP formulation. However, they use squared slack variables for inequalities in the optimization problem which can be easily shown to fail the linear independence constraint qualification (LICQ), thus making the corresponding KKT system inconsistent.

In this chapter, we develop a trust-region framework for solving constrained optimization problems using reduced-order models. In particular, we explore penalty and filter based approaches to manage infeasibility and utilize a few concepts from Fahl's TRPOD algorithm, Alexandrov's scaling (correction) scheme for the objective function and the constraints, and MAESTRO-AMMO algorithm. The algorithms developed are demonstrated for a case study of a two-bed four step isothermal PSA process for post-combustion CO₂ capture.

7.2 Optimization Problem

7.2.1 Trust-region Subproblem

In this work, the original optimization problem is represented by a nonlinear program of the following form

$$\begin{aligned}
 \min_x \quad & f(x) \\
 \text{s.t.} \quad & c_{\mathcal{E}}(x) = 0 \\
 & c_{\mathcal{I}}(x) \leq 0 \\
 & x_L \leq x \leq x_U
 \end{aligned} \tag{7.1}$$

where $x \in \mathbb{R}^n$ are the decision variables bounded between lower and upper bounds x_L and x_U , respectively, and the objective function $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$, equality constraints $c_{\mathcal{E}}(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{N_E}$, and inequality constraints $c_{\mathcal{I}}(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{N_I}$ are assumed to be sufficiently smooth and at least twice differentiable functions. We note that Problem (7.1) is written in the reduced-space of the original DAE-constrained optimization problem. DAEs are integrated outside Problem (7.1), and the solution profiles are then used to compute the objective function and the constraints.

At k^{th} iteration during the course of optimization, the ROM constructed for a particular x_k is used to build the model function for the trust-region subproblem. We define a ROM-based

trust-region subproblem for k^{th} iteration as

$$\begin{aligned}
& \min_s && f_k^{\mathcal{R}}(x_k + s) \\
& \text{s.t.} && c_{\mathcal{E},k}^{\mathcal{R}}(x_k + s) = 0 \\
& && c_{\mathcal{I},k}^{\mathcal{R}}(x_k + s) \leq 0 \\
& && x_L \leq x_k + s \leq x_U \\
& && \|s\|_{\infty} \leq \Delta_k
\end{aligned} \tag{7.2}$$

where $f_k^{\mathcal{R}}(x_k + s)$ is the objective function and $c_{\mathcal{E},k}^{\mathcal{R}}(x_k + s)$, and $c_{\mathcal{I},k}^{\mathcal{R}}(x_k + s)$ are the equality and inequality constraints, respectively, computed from the reduced set of state variables of the reduced-order model. For this subproblem also, DAEs of the ROM are solved outside Problem (7.2) and the solution of the unknown temporal coefficients in the POD expansion is then used to obtain $f_k^{\mathcal{R}}(x_k + s)$, $c_{\mathcal{E},k}^{\mathcal{R}}(x_k + s)$, and $c_{\mathcal{I},k}^{\mathcal{R}}(x_k + s)$. The last inequality in Problem (7.2) is the trust-region constraint which limits the step size within the current trust-region radius Δ_k . In this work, we prefer to use an infinity norm for the trust-region constraint, i.e., we use a box-type (ℓ_{∞}) trust-region to restrict the step size of the decision variables.

It should be noted that the dimension of x , $f(x)$, $c_{\mathcal{E}}(x)$, and $c_{\mathcal{I}}(x)$ remains same for both original optimization problem as well as ROM-based trust-region subproblem. In other words, the number of decision variables and constraints remain same for both problems. Computational advantage is achieved in terms of the smaller number of DAEs of the reduced-order model which leads to cheap calculation of the gradients of the objective function and the constraints in the trust-region subproblem (7.2).

7.2.2 Correction (scaling) for Objective and Constraints

To develop a robust and globally convergent trust-region algorithm involving approximate models, the following assumptions should hold [54]

(AF1) Functions $f(x)$, $c_{\mathcal{E}}(x)$, and $c_{\mathcal{I}}(x)$ are twice-continuously differentiable on \mathbb{R}^n .

(AF2) The function $f(x)$ is bounded below for all $x \in \mathbb{R}^n$.

(AF3) The second derivatives of $f(x)$, $c_{\mathcal{E}}(x)$, and $c_{\mathcal{I}}(x)$ are uniformly bounded for all $x \in \mathbb{R}^n$.

(A1) For every iteration k , $f_k^{\mathcal{R}}$ is twice differentiable on \mathcal{B}_k , where

$$\mathcal{B}_k = \{x \in \mathbb{R}^n \mid \|x - x_k\|_{\infty} \leq \Delta_k\}, \quad \Delta_k > 0$$

(A2) The values of the objective and the constraints for the original optimization problem and the trust-region subproblem coincide at the current iterate, i.e., for all k ,

$$f(x_k) = f_k^{\mathcal{R}}(x_k) \quad c_{\mathcal{E}}(x_k) = c_{\mathcal{E},k}^{\mathcal{R}}(x_k) \quad c_{\mathcal{I}}(x_k) = c_{\mathcal{I},k}^{\mathcal{R}}(x_k) \quad (7.3)$$

(A3) The gradient of the objective and the Jacobian of the constraints for both the problems coincide, i.e., for all k ,

$$\nabla_x f(x_k) = \nabla_x f_k^{\mathcal{R}}(x_k) \quad \nabla_x c_i(x_k) = \nabla_x c_{i,k}^{\mathcal{R}}(x_k) \quad i \in \{\mathcal{E}, \mathcal{I}\} \quad (7.4)$$

(A4) The second derivatives of $f_k^{\mathcal{R}}(x_k)$, $c_{\mathcal{E},k}^{\mathcal{R}}(x_k)$, and $c_{\mathcal{I},k}^{\mathcal{R}}(x_k)$ remain bounded within the trust-region \mathcal{B}_k for all k .

In this work, it is assumed that **(AF1)**–**(AF3)** are true, and assumptions **(A1)** and **(A4)** hold by construction of the ROM. However, it cannot be guaranteed if assumptions **(A2)** and **(A3)** (also called *first-order consistency conditions*) will be true in general. Depending on the accuracy of the reduced-order model, values of the objective and the constraints for the original problem and the ROM-based trust-region subproblem may reasonably match; however, their gradients will differ since the POD basis set is obtained from the snapshots containing state information but no gradient information. One way to ensure reasonable gradient matching is to develop a separate ROM for the sensitivity equations of the original system, and solve it with the ROM for the state variables. Fahl et al. [66] applied such an approach with the adjoint sensitivity equations of the system and generated a separate ROM for the adjoint variables, different from the ROM for the state variables. However, they reported that such an approach can lead to inconsistent gradients in Problem (7.2), and thus to an algorithm which is not globally convergent.

To circumvent the dependency on the accuracy of ROM, and to ensure globally-convergent behavior of the trust-region algorithm, Alexandrov et al. [8, 10], Eldred et al. [64], and Giunta

et al. [83] propose enforcing assumptions **(A2)** and **(A3)** by using scaled/corrected objective and constraints in the trust-region subproblem which can be derived by using local corrections corresponding to the current iterate k . However, these corrections require computation of the gradient of the original objective function and the Jacobian of original constraints. In particular, they propose two types of local corrections, additive and multiplicative, which can be defined in the following manner

$$\text{Additive: } \tilde{\Phi}_k^{\mathcal{R}}(x) = \Phi_k^{\mathcal{R}}(x) + (\Phi(x_k) - \Phi_k^{\mathcal{R}}(x_k)) + (\nabla\Phi(x_k) - \nabla\Phi_k^{\mathcal{R}}(x_k))^T(x - x_k) \quad (7.5)$$

Multiplicative:

$$\tilde{\Phi}_k^{\mathcal{R}}(x) = \Phi_k^{\mathcal{R}}(x) \left(\frac{\Phi(x_k)}{\Phi_k^{\mathcal{R}}(x_k)} + \left(\frac{\Phi_k^{\mathcal{R}}(x_k)\nabla\Phi(x_k) - \Phi(x_k)\nabla\Phi_k^{\mathcal{R}}(x_k)}{\Phi_k^{\mathcal{R}}(x_k)^2} \right)^T (x - x_k) \right) \quad (7.6)$$

$$\text{where } \Phi(x) : f(x), c_i(x) \quad \Phi_k^{\mathcal{R}}(x) : f_k^{\mathcal{R}}(x), c_{i,k}^{\mathcal{R}}(x) \quad i \in \{\mathcal{E}, \mathcal{I}\}$$

Multiplicative correction can become ill-conditioned and may require additional modification when $\Phi_k^{\mathcal{R}}(x_k)$ gets close to zero, especially for the equality constraints and active inequality constraints. Hence, we prefer to use the additive correction for our work. For both corrections, it is obvious that at the current iterate x_k

$$\begin{aligned} \tilde{f}_k^{\mathcal{R}}(x_k) &= f(x_k), & \tilde{c}_{i,k}^{\mathcal{R}}(x_k) &= c_i(x_k), & \text{and} \\ \nabla_x \tilde{f}_k^{\mathcal{R}}(x_k) &= \nabla_x f(x_k), & \nabla_x \tilde{c}_{i,k}^{\mathcal{R}}(x_k) &= \nabla_x c_i(x_k), & i \in \{\mathcal{E}, \mathcal{I}\} \end{aligned}$$

Therefore, we re-define the trust-region subproblem (7.2) in terms of the corrected objective and constraints as below

$$\begin{aligned} \min_s \quad & \tilde{f}_k^{\mathcal{R}}(x_k + s) \\ \text{s.t.} \quad & \tilde{c}_{\mathcal{E},k}^{\mathcal{R}}(x_k + s) = 0 \\ & \tilde{c}_{\mathcal{I},k}^{\mathcal{R}}(x_k + s) \leq 0 \\ & x_L \leq x_k + s \leq x_U \\ & \|s\|_{\infty} \leq \Delta_k \end{aligned} \quad (7.7)$$

It is worth noting that in these correction schemes, the gradient of the original objective function and the Jacobian of the constraints is computed only once at the center of the trust-region for a single trust-region iteration. Optimization within a trust-region is performed using the cheap gradient of the objective and Jacobian of the constraints of the reduced-order model, thus offering computational advantage. However, $\nabla f(x_k)$, $\nabla c_{\mathcal{E},k}(x_k)$, and $\nabla c_{\mathcal{I},k}(x_k)$ will have to be evaluated for each trust-region iteration if corrected functions $\tilde{f}_k^{\mathcal{R}}(x)$, $\tilde{c}_{\mathcal{E},k}^{\mathcal{R}}(x)$, and $\tilde{c}_{\mathcal{I},k}^{\mathcal{R}}(x)$ are used. In this work, we define two kinds of additive correction schemes

Zero-order Correction (ZOC)

$$\tilde{\Phi}_k^{\mathcal{R}}(x) = \Phi_k^{\mathcal{R}}(x) + (\Phi(x_k) - \Phi_k^{\mathcal{R}}(x_k)) \quad (7.8)$$

First-order Correction (FOC)

$$\tilde{\Phi}_k^{\mathcal{R}}(x) = \Phi_k^{\mathcal{R}}(x) + (\Phi(x_k) - \Phi_k^{\mathcal{R}}(x_k)) + (\nabla\Phi(x_k) - \nabla\Phi_k^{\mathcal{R}}(x_k))^T(x - x_k) \quad (7.9)$$

where $\Phi(x) : f(x), c_i(x)$ $\Phi_k^{\mathcal{R}}(x) : f_k^{\mathcal{R}}(x), c_{i,k}^{\mathcal{R}}(x)$ $i \in \{\mathcal{E}, \mathcal{I}\}$

We can obtain Zero-order Correction cheaply as it doesn't require gradient evaluation for the original objective and the constraints. However, with ZOC, only assumption **(A2)** is satisfied while assumption **(A3)** is not guaranteed. On the other hand, First-order Correction ensures both assumption **(A2)** and **(A3)** are satisfied, although it is expensive to construct.

7.3 Penalty-based Trust-region Algorithm

7.3.1 Penalty Formulation

Note that if Problem (7.7) is constructed at an infeasible point, the trust-region box may be too small to satisfy the constraints in (7.7). Thus, handling feasibility needs special treatment in a trust-region framework. Few trust-region algorithms are designed to deal with general equality and inequality constraints. To attain feasibility, first we explore a penalty-based formulation,

and utilize the following ℓ_1 exact penalty function [71, 140]

$$\tilde{\psi}_k^{\mathcal{R}}(x_k + s) = \tilde{f}_k^{\mathcal{R}}(x_k + s) + \mu \sum_{i \in \mathcal{E}} |\tilde{c}_{i,k}^{\mathcal{R}}(x_k + s)| + \mu \sum_{i \in \mathcal{I}} \max(0, \tilde{c}_{i,k}^{\mathcal{R}}(x_k + s)) \quad (7.10)$$

to reformulate the trust-region subproblem (7.7) into the following bound-constrained problem

$$\begin{aligned} \min_s \quad & \tilde{\psi}_k^{\mathcal{R}}(x_k + s) \\ \text{s.t.} \quad & x_L \leq x_k + s \leq x_U \\ & \|s\|_{\infty} \leq \Delta_k \end{aligned} \quad (7.11)$$

Here μ is the penalty parameter, which must be chosen sufficiently large. Note that the bound constraints $x_L \leq x_k + s \leq x_U$ are either ignored if the box trust-region lies completely within the polytope defined by them, or help to redefine the box trust-region if it intersects with the polytope.

A penalty based formulation enables us to minimize the objective function while controlling constraint violations by penalizing them. The penalty function is exact in the sense that for a sufficiently high μ , the local solution of (7.7) is equivalent to the local minimizer of (7.11). To evaluate the actual reduction obtained in the original objective function in (7.1), we define the corresponding exact penalty function as

$$\psi(x_k + s) = f(x_k + s) + \mu \sum_{i \in \mathcal{E}} |c_i(x_k + s)| + \mu \sum_{i \in \mathcal{I}} \max(0, c_i(x_k + s)) \quad (7.12)$$

Since the penalty functions are non-differentiable, we adopt the following smoothing approximation [20]. A value of 0.01 is used for ϵ in the following equations.

$$\max(0, f(x)) = 0.5 \left(f(x) + \sqrt{f(x)^2 + \epsilon^2} \right) \quad (7.13a)$$

$$|f(x)| = \max(0, f(x)) + \max(0, -f(x)) = \sqrt{f(x)^2 + \epsilon^2} \quad (7.13b)$$

One of the main issues with penalty functions is to find a reasonable value for the penalty

parameter μ . Since the Lagrange multipliers, and thus the lower bound on μ are not known a priori, choice of a value for μ is not intuitive. Too high a value for μ can cause the algorithm to suffer from poor performance and ill-conditioning. Usually μ is adjusted by some update criterion as the algorithm proceeds, and an acceptable step is decided thereafter. Such an approach can provide considerable flexibility in choosing larger steps. However, in this work, we do not propose any update mechanism for μ and work with a constant value which is decided before the algorithm begins.

7.3.2 Trust-region Algorithm

Algorithm I: Exact penalty trust-region algorithm

Choose $0 < \eta_1 \leq \eta_2 < 1 \leq \eta_3$, $0 < \gamma_1 \leq \gamma_2 < 1 < \gamma_3$, penalty $\mu > 0$, an initial trust-region radius Δ_0 , minimum radius Δ_{min} , and an initial iterate x_0 . Compute $\psi(x_0)$ and set $k = 0$.

1. Compute POD basis functions and construct a reduced-order model using the snapshots obtained for current iterate x_k .
2. Compute a step s_k from (7.11). Problem (7.11) can also be solved “approximately” such that a sufficient decrease condition (7.14) or (7.15) is satisfied.
3. Compute $\psi(x_k + s_k)$ and define the ratio

$$\rho_k = \frac{ared_k}{pred_k} = \frac{\psi(x_k) - \psi(x_k + s_k)}{\tilde{\psi}_k^{\mathcal{R}}(x_k) - \tilde{\psi}_k^{\mathcal{R}}(x_k + s_k)}$$

If $\rho_k < \eta_1$, set $x_{k+1} = x_k$ and $\Delta_{k+1} = \gamma_1 \Delta_k$. If $\Delta_{k+1} \leq \Delta_{min}$, STOP, otherwise increment k by 1 and go to Step 2.

4. Set $x_{k+1} = x_k + s_k$,

$$\Delta_{k+1} = \begin{cases} \gamma_2 \Delta_k & \text{if } \rho_k \in [\eta_1, \eta_2), \\ \Delta_k & \text{if } \rho_k \in [\eta_2, \eta_3), \\ \gamma_3 \Delta_k & \text{if } \rho_k \geq \eta_3 \end{cases}$$

Increment k by 1 and go to Step 1.

The algorithm repeatedly solves the ROM-based trust region subproblem (7.11) with the Zero-order Correction (ZOC) or the First-order Correction (FOC) for the objective and the constraints until the trust-region radius shrinks to a value less than Δ_{min} , and no further improvement is obtained. In order to estimate the quality of the next iterate, we compare the actual reduction in the true objective $ared_k$, to the predicted reduction $pred_k$. This requires computation of a new solution for the DAEs of the original problem in order to evaluate $\psi(x_k + s_k)$. If the trial step s_k yields a satisfactory decrease, and if $\rho_k \geq \eta_1$, it is accepted and we update the ROM for next iteration with the help of these new snapshots just calculated. Otherwise, the size of the trust-region is reduced and Step 2 is repeated with a smaller trust-region. In Step 2, “approximately” means that a solution s_k can be obtained in any manner suitable to the application as long as it satisfies the following sufficient decrease condition, also known as the fraction of Cauchy decrease condition [54]

$$\tilde{\psi}_k^{\mathcal{R}}(x_k) - \tilde{\psi}_k^{\mathcal{R}}(x_k + s_k) \geq \kappa_p \|\nabla\psi(x_k)\| \min \left[\Delta_k, \frac{\|\nabla\psi(x_k)\|}{\beta_k} \right] \quad (7.14)$$

where $0 < \kappa_p < 1$, while $1 < \beta_k < \infty$ is any bounded sequence of numbers (note that $\nabla\psi(x_k)$ can be computed because of the smoothing approximation (7.13)). However, at the beginning of the algorithm, when not close to the optimum, equation (7.14) can be replaced by the following gradient free sufficient decrease condition [186]

$$\tilde{\psi}_k^{\mathcal{R}}(x_k) - \tilde{\psi}_k^{\mathcal{R}}(x_k + s_k) \geq \kappa_p \min[\nu_1, \nu_2 \Delta_k] \quad (7.15)$$

for some $0 < \kappa_p < 1$, $0 < \nu_2 \leq 1$, $\nu_1 > 0$. Although a step s_k can be computed approximately, in this work we find an exact local optimum for Problem (7.11) using IPOPT for each iteration.

One of the key features of this algorithm is that once we achieve feasibility during the course of the algorithm, we stop using the penalty formulation, constraint relaxation is removed and they are transferred back to the trust-region subproblem. In other words, Problem (7.11) is converted back to Problem (7.7). This is especially important when FOC is used for objective

and constraints as the penalty parameter can put a lot of weight on constraint violation and its gradient, thus allowing tiny reduction in the objective with each trust-region iteration. Moreover, with feasible equalities and active inequalities, smoothing parameter ϵ (see Equation (7.13)) in the penalty function can substantially skew the results and can make the algorithm terminate prematurely.

In this work we use the following values for the constants in the algorithm

$$\eta_1 = 0.05, \eta_2 = 0.5, \eta_3 = 1, \gamma_1 = 0.25, \gamma_2 = 0.5, \gamma_3 = 1.5$$

Note that the value for η_1 is very close to zero. This implies that we take a step even if the reduction in $\psi(x)$ is quite small. This is essential for POD ROM-based optimization because computation of $\psi(x_k + s_k)$ in ρ_k involves evaluation of new snapshots from the original DAEs. Hence, it is always beneficial to take a step, even if ρ_k is small, and use these new snapshots to update our ROM which we expect to perform better in the next iteration. Also, the choice of $\eta_2 = 0.5$ and $\eta_3 = 1$ shows that most of the time during the algorithm we wish to keep the trust-region radius constant instead of increasing it frequently. With POD ROM-based optimization we prefer not to be greedy and limit the growth of the trust-region for longer duration because of the oscillatory behavior of the ROM for large trust-regions, as observed in the previous chapter. With oscillations that result from large extrapolation, ROM can take the algorithm in a direction which may not be a descent direction, which can cause ρ_k to become negative and lead to drastic reductions in Δ_k for subsequent iterations. The oscillatory behavior can be monitored by checking whether the state variables are within the defined bounds or not. Such bounds may be used as safeguards to avoid oscillatory behavior.

7.3.3 Convergence and Optimality

Since Algorithm I becomes a basic trust-region algorithm because of the smoothing approximation (7.13), it enjoys the following convergence property.

Theorem 7.3.1. (See Theorem 6.4.6 in [54]) Suppose that **(AF1)**–**(AF3)**, **(A1)**–**(A4)**, and

(7.14) hold. Then one has that

$$\lim_{k \rightarrow \infty} \|\nabla \psi(x_k)\| = 0 \quad (7.16)$$

In other words, all limit points for Algorithm I converge to x^* , a first-order optimal point for Problem (7.1), when all the assumptions **(AF1)**–**(AF3)**, and **(A1)**–**(A4)** hold, and a sufficient decrease (7.14) can be ensured. As mentioned before, **(AF1)**–**(AF3)**, **(A1)** and **(A4)** are assumed to be true in this work. With Zero-order Correction, we can satisfy **(A2)**, while First-order Correction can ensure both **(A2)** and **(A3)** are satisfied. With assumptions **(A2)** and **(A3)** being true, it can be shown that as the trust-region gets small enough, the linear part of $\tilde{\psi}_k^{\mathcal{R}}(x_k + s)$ dominates and we can always compute a step s_k which lies in the steepest descent direction of $\tilde{\psi}_k^{\mathcal{R}}(x_k + s)$ (Cauchy step) and satisfies the fraction of Cauchy decrease condition (7.14). Moreover, with FOC, the Cauchy step obtained for Problem (7.11) coincides with the exact Cauchy step of Problem (7.11) with $\tilde{\psi}_k^{\mathcal{R}}(x_k + s)$ replaced with $\psi(x_k + s)$. Hence, Algorithm I is globally convergent.

Assumptions **(A2)**, and **(A3)** should necessarily be satisfied at the optimum to ensure that the Algorithm I converges to a solution that corresponds to the optimum of the original optimization problem (7.1) [29, 77]. Since we satisfy **(A2)** and **(A3)** with FOC for all the trust-region iterations, it is ensured that the algorithm will converge to the true optimum. We follow the approach presented in section 6.5.4.1 for our case studies, in order to verify if Algorithm I converges to the true optimum.

7.4 PSA Case Study - Post Combustion CO₂ Capture

7.4.1 PSA Process and Model Equations

We consider a 2-bed 4-step isothermal PSA process with an 85%-15% N₂-CO₂ feed mixture which is a typical composition of a post-combustion flue gas stream. In particular, the process is a Skarstrom cycle shown in Figure 2.3, and described in detail in section 2.4.1. Zeolite 13X is chosen as the adsorbent to separate CO₂. We make following assumptions to develop a

Table 7.1: Model equations for isothermal PSA

Component mass balance	
$\epsilon_b \frac{\partial y_i}{\partial t} + \frac{\partial(vy_i)}{\partial x} + \frac{RT}{P}(1 - \epsilon_b)\rho_s \frac{\partial q_i}{\partial t} = 0 \quad i = 1$	(7.17)
Overall mass balance	
$\frac{\partial v}{\partial x} + \frac{RT}{P}(1 - \epsilon_b)\rho_s \sum_{i=1}^2 \frac{\partial q_i}{\partial t} = 0$	(7.18)
LDF equation	
$\frac{\partial q_i}{\partial t} = k_i(q_i^* - q_i) \quad i = 1, 2$	(7.19)
Dual-site Langmuir Isotherm	
$q_i^* = \frac{q_{1i}^s b_{1i} y_i P}{1 + \sum_j b_{1j} y_j P} + \frac{q_{2i}^s b_{2i} y_i P}{1 + \sum_j b_{2j} y_j P} \quad i = 1, 2$	(7.20)
Cyclic steady state	
$z(t_0) = z(t_{cycle}) \quad z : y_i, q_i \quad \forall i$	(7.21)

mathematical model for this process

1. All of the gases follow the ideal gas law.
2. There are no radial variations for concentrations in the solid and the gas phase.
3. The process is isothermal with a fixed temperature for the entire cycle.
4. Pressure drop along the bed is negligible.
5. The adsorption behaviors are described by the dual-site Langmuir isotherm.
6. The adsorption rate is approximated by the linear driving force (LDF) expression.

Based on the above assumptions, the mathematical model for the PSA process is listed in Table 7.1. Here we assume no axial dispersion and use a lumped mass transfer coefficient for the LDF model. The adsorbent properties for 13X and other model parameters are listed in Table 7.2 [111]. Since we also have an overall mass balance in the model to solve for velocity

Table 7.2: Zeolite 13X properties and model parameters [111]

Parameter	Value	
Bed Length (L)	1 m	
Bulk porosity (ϵ_b)	0.34	
Adsorbent density (ρ_s)	1870 kg m ⁻³	
Mass transfer coefficient (k)	CO ₂ =0.1631 sec ⁻¹ N ₂ =0.2044 sec ⁻¹	
Process temperature (T)	310 K	
Isotherm parameters		
	CO ₂	N ₂
q_1^s	2.708769	1.819949
q_2^s	2.436388	1.819949
b_1	1.23×10^{-5}	6.17×10^{-7}
b_2	4.78×10^{-4}	6.17×10^{-7}

Table 7.3: Boundary conditions for each operating step

pressurization	adsorption	depressurization	desorption
$y_j _{x=0} = y_{f,j}$	$y_j _{x=0} = y_{f,j}$	—	$y_j _{x=L} = y_j _{x=L}$ of other bed
$P = P_h$	$P = P_h$	$P = P_l$	$P = P_l$
$u _{x=L} = 0$	$u _{x=0} = u_a$	$u _{x=L} = 0$	$u _{x=L} = 0.4u _{x=L}$ of other bed

along the bed length, we solve component mass balance for only one component. Moreover, we don't have to ensure that the mole fractions sum up to one as it implicitly happens because of the overall mass balance. We denote this model in Table 7.1 as the rigorous model for which we develop a reduced-order model.

Boundary conditions for each step are shown in Table 7.3. We note that for the depressurization step a boundary condition for mole fraction is not needed since $u|_{x=L} = 0$ automatically sets the inlet flux to be zero for the component mass balance. However, boundary conditions are needed for both velocity and mole fraction for the pressurization step due to the nature of the upwind-based spatial discretization scheme. Also, we note that the purge fraction of the outlet from $x = L$ which goes from one bed to the other during the adsorption step is chosen to be 0.4, which appears in the boundary condition of velocity for the desorption step.

Tables 6.4 in the previous chapter shows the equations for molar flux variables of the 2-bed 4-step process which is used to calculate purities and recoveries of nitrogen and CO₂ in the following manner

$$\text{purity}_{N_2} = \frac{\int O_{N_2}(t) dt}{\int O_{N_2}(t) + O_{CO_2} dt} \quad (7.22)$$

$$\text{purity}_{CO_2} = \frac{\int H_{pCO_2}(t) dt}{\int H_{pN_2}(t) + H_{pCO_2} dt} \quad (7.23)$$

$$\text{recovery}_{N_2} = \frac{\int O_{N_2}(t) dt - \int P_{gN_2}(t) dt}{\int F_{N_2}(t) dt} \quad (7.24)$$

$$\text{recovery}_{CO_2} = \frac{\int H_{pCO_2}(t) dt}{\int F_{CO_2}(t) dt} \quad (7.25)$$

7.4.2 Reduced-order Model

We use the method of lines approach to convert PDAEs in Table 7.1 to a set of DAEs after spatial discretization, which is then simulated to generate snapshots for POD basis functions. For spatial discretization, we use an upwind-based finite volume scheme with the Van Leer flux limiter for mole fraction to introduce additional numerical dispersion around steep adsorption fronts (*cf.* section 2.5.1). Moreover, we utilize *Unibed* approach and simulate only one bed for the entire cycle with the help of storage buffers to handle boundary matching for the two beds during adsorption and desorption steps (*cf.* section 2.4.3).

It is important to note that the snapshots are generated only after the 2-bed system attains cyclic steady state. To achieve CSS, we use a successive substitution method in which a series of simulations are performed with initial conditions of each new cycle taken from the final condition of the previous cycle until the bed conditions do not change from cycle to cycle. In this case, we achieve cyclic steady state after 50-60 cycles.

After obtaining snapshots, separate POD basis functions are generated for pressurization,

Table 7.4: ROM for the isothermal PSA process

Component mass balance

$$\epsilon_b \frac{da_{yik}}{dt} + \left(u^R \left(\frac{dy_{0i}}{dx} + \sum_{j=1}^M a_{yij} \frac{d\phi_{yij}}{dx} \right), \phi_{yik} \right) + \left(y_i^R \left(\frac{du_0}{dx} + \sum_{j=1}^M a_{uj} \frac{d\phi_{uj}}{dx} \right), \phi_{yik} \right) + \frac{RT}{P} (1 - \epsilon_b) \rho_s \sum_{j=1}^M (\phi_{yik}, \phi_{qij}) \frac{da_{qij}}{dt} = 0 \quad \forall k \in [1, M], \quad i = 1 \quad (7.26)$$

Overall mass balance

$$\left(\left(\frac{du_0}{dx} + \sum_{j=1}^M a_{uj} \frac{d\phi_{uj}}{dx} \right), \phi_{uk} \right) + \frac{RT}{P} (1 - \epsilon_b) \rho_s \sum_{i=1}^2 \sum_{j=1}^M (\phi_{uk}, \phi_{qij}) \frac{da_{qij}}{dt} = 0 \quad (7.27)$$

$\forall k \in [1, M]$

LDF equation

$$\frac{da_{qik}}{dt} = k_i \left(\sum_{j=1}^M (q_i^{R*}, \phi_{qik}) - a_{qik} - (q_{0i}, \phi_{qik}) \right) \quad \forall k \in [1, M] \quad (7.28)$$

Langmuir isotherm

$$q_i^{R*} = \frac{q_{1i}^s b_{1i} y_i^R P}{1 + \sum_{j=1}^2 b_{1j} y_j^R P} + \frac{q_{2i}^s b_{2i} y_i^R P}{1 + \sum_{j=1}^2 b_{2j} y_j^R P} \quad i = 1, 2 \quad (7.29)$$

Cyclic steady state

$$a_{z,k}(0) \Big|_{pres} = \left(\sum_{j=1}^M a_{z,j}(t_{cycle}) \Big|_{des} \phi_{z,j} \Big|_{des}, \phi_{z,k} \Big|_{pres} \right) \quad \forall k \in [1, M], \quad z : y_i, q_i \quad (7.30)$$

adsorption, depressurization and desorption steps. Moreover, we derive separate POD basis functions for gas phase mole fractions, solid phase loadings, and velocity. State variables are then expressed in terms of the corresponding POD basis as below

$$y_i^R(x, t) = y_{0i}(x) + \sum_{j=1}^M a_{yij}(t) \phi_{yij}(x) \quad \text{for } i = 1$$

$$q_i^R(x, t) = q_{0i}(x) + \sum_{j=1}^M a_{qij}(t) \phi_{qij}(x) \quad u^R(x, t) = u_0(x) + \sum_{j=1}^M a_{uj}(t) \phi_{uj}(x)$$

Here $y_{0i}(x)$, $q_{0i}(x)$, and $u_0(x)$ are snapshot averages for mole fraction, solid loading, and velocity respectively. It is noteworthy to mention that for the adsorption step, we use the

following POD basis representation for velocity to make the adsorption feed velocity u_a visible during optimization, as we consider it a decision variable:

$$u^R(x, t) = u_a + \sum_{j=1}^M a_{uj}(t) \phi_{uj}(x) \quad \text{for adsorption step.}$$

Table 7.4 shows DAEs of the reduced-order model obtained after Galerkin projection of the model in Table 7.1 on to these POD basis functions.

7.4.3 Optimization using Algorithm I

In this section we apply the exact penalty trust-region algorithm on a ROM-based optimization problem which maximizes CO₂ recovery subject to a constraint on CO₂ purity. For optimization, we consider five decision variables, high pressure P_h up to which the bed is pressurized and at which the adsorption step takes place, low pressure P_l for the depressurization and desorption steps, step time for pressurization and depressurization t_p , and that for adsorption and desorption t_a , and finally, feed velocity during the adsorption step u_a . The DAE-constrained optimization problem is described as below

$$\begin{aligned} \max \quad & \text{CO}_2 \text{ recovery (from Equation (7.25))} \\ \text{s.t.} \quad & \text{CO}_2 \text{ purity} \geq 0.5 \text{ (from Equation (7.23))} \\ & 101.32 \text{ kPa} \leq P_h \leq 300 \text{ kPa} \\ & 40 \text{ kPa} \leq P_l \leq 101.32 \text{ kPa} \\ & 35 \text{ sec} \leq t_p \leq 150 \text{ sec} \\ & 50 \text{ sec} \leq t_a \leq 400 \text{ sec} \\ & 10 \text{ cm/sec} \leq u_a \leq 30 \text{ cm/sec} \end{aligned} \tag{7.31}$$

Here, a lower bound of 50% is reasonable for CO₂ purity since we are using Skarstrom cycle to extract CO₂, which lacks any kind of a step that enriches CO₂ concentration in the bed. Thus, we cannot achieve high CO₂ purity with this 2-bed 4-step cycle. However, our focus here is to illustrate the exact penalty trust-region algorithm for optimization using ROMs with the help of this case study. We also note that in order to improve CO₂ purity and recovery, we allow vacuum depressurization and desorption steps as the bounds for P_l lie in the vacuum range.

Problem (7.31) is solved by using the ROM-based trust-region subproblem repeatedly. We discretize the DAEs of the ROM in the temporal dimension and convert it into a standard

Table 7.5: Initial guess for optimization problem (7.31)

Decision variable	Guess value
Adsorption pressure (P_h)	150 kPa
Desorption pressure (P_l)	50 kPa
Pressurization step time (t_p)	50 sec
Adsorption step time (t_a)	150 sec
Adsorption feed flow (u_a)	20 cm/sec

NLP which is solved in AMPL using IPOPT. We note that to construct a ROM, snapshots are obtained only after CSS is achieved by the rigorous model. Therefore, it is ensured that CSS is satisfied by ROM as well for every trust-region iteration. In other words, CSS is solved internally and is not a part of the infeasibility measure in Problem (7.31).

In the subsequent sections, we first compare the accuracy of the reduced-order model at the starting guess x_0 for the optimization problem. Next, we solve the optimization problem (7.31) with ZOC for the objective and the purity constraint and monitor if Algorithm I converges to an optimum even when assumption **(A3)** is not true. Finally, we solve problem (7.31) with FOC. The quality of the dynamic behavior predicted by ROM is compared at the optimum as well.

7.4.3.1 Comparison of Rigorous Model and ROM at the Starting Guess

In this section, we validate the accuracy of the ROM and verify how accurately ROM is predicting the dynamic behavior of the original isothermal PSA process at the initial guess x_0 . Table 7.5 lists the starting guess for our optimization problem (7.31).

With 50 spatial finite volumes, we convert the PDAEs in Table 7.1 into DAEs and integrate using *ode15s* in MATLAB. Cyclic steady state is achieved up to the desired tolerance after simulating the model repeatedly for 50 cycles, and snapshots are collected to generate empirical POD basis. Figure 7.1 shows first six POD basis functions of gas-phase CO₂ mole fraction for the adsorption step. The shapes of these functions are quite different since they are empirical in nature. Figure 7.2 shows first 10 singular values for gas-phase CO₂ mole fraction and

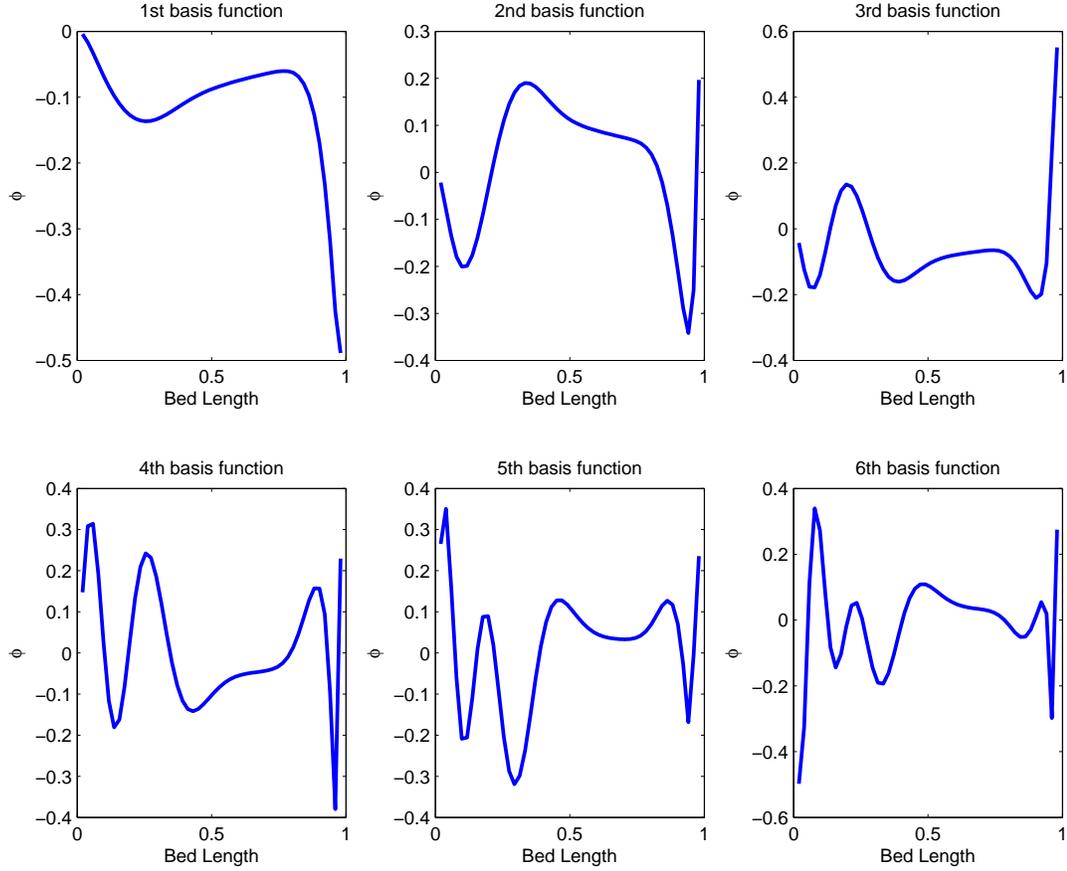


Figure 7.1: First six POD basis functions of CO₂ mole fraction for adsorption

superficial gas velocity for all the operating steps. From the slopes of the curves, it is clear that the singular values decay fairly sharply. For this case study, we choose a threshold error tolerance λ^* of 0.05 (*cf.* (6.10) and section 6.2.4). For $\varepsilon_{norm}^{POD}(M) \leq \lambda^*$ to hold true, we obtain $M = 2, 4, 1,$ and 3 for pressurization, adsorption, depressurization, and desorption steps, respectively. In other words, with such few basis functions error in projection can be at most 5%. We purposely choose a slightly higher value of 0.05 for the threshold tolerance since it is observed that with a low value of λ^* (say 0.01 or 0.001), ROM incorporates those POD basis functions which do not contribute much towards predicting the dynamics, thus causing the DAE system of the ROM to become ill-conditioned. Moreover, with these values of M , ROM comprises a mere 70 DAEs, while the rigorous model contains a total of 1400 DAEs. Hence, we obtain a significant model reduction as ROM is $1/20^{th}$ of the rigorous model with this choice

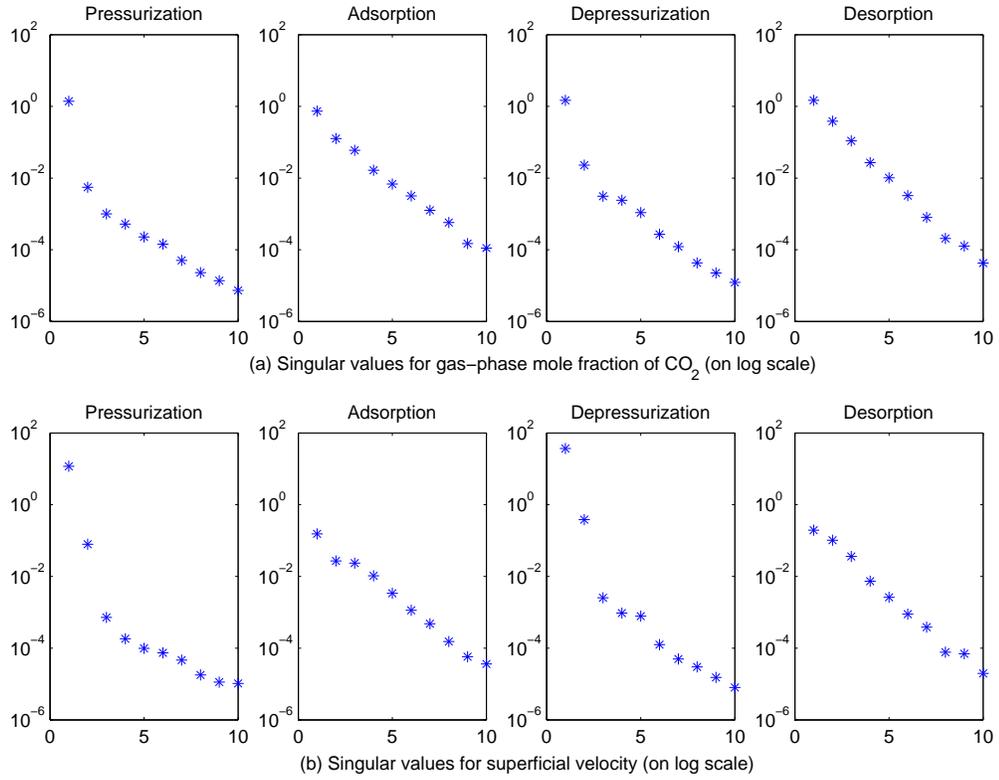
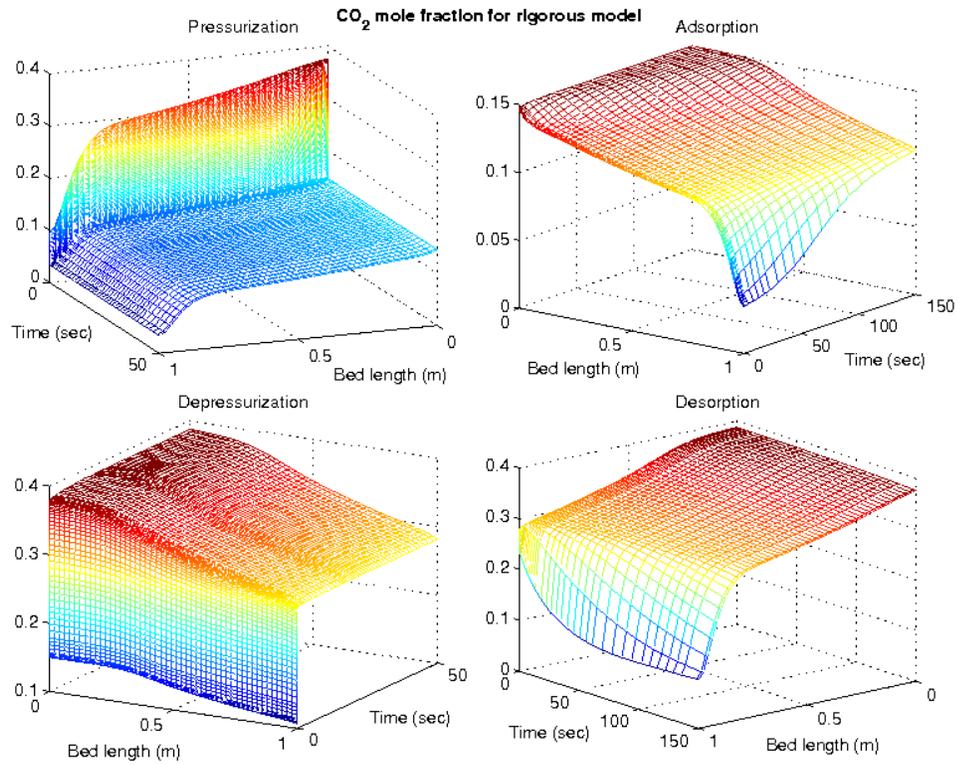


Figure 7.2: Singular values of mole fraction of CO₂ and superficial gas velocity

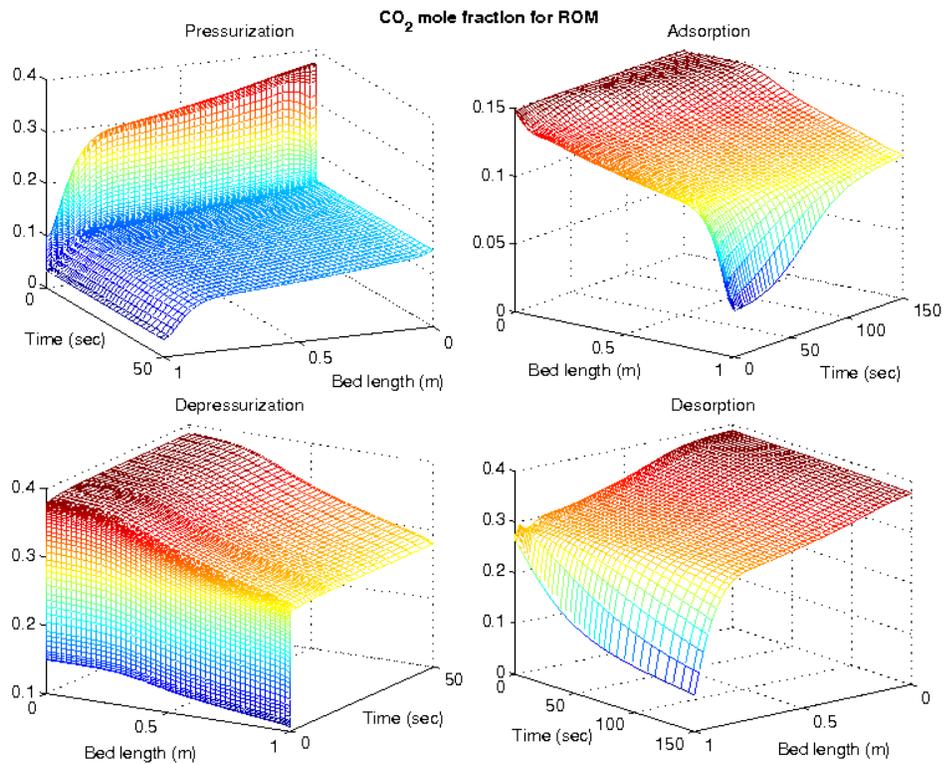
of λ^* .

We discretize the DAEs of the reduced-order model in time with 20 finite elements and 3 collocation points for all four operating steps, and the resulting algebraic equations are solved with IPOPT. The initial conditions for the process are taken as decision variables and reduced CSS conditions (shown in Table 7.4) are solved simultaneously with the model equations in AMPL. Figure 7.3 compares the profiles of gas-phase CO₂ mole fraction obtained after integrating the rigorous model till CSS is achieved, and after solving the algebraic equations of ROM with reduced CSS conditions. We observe a significant match between the profiles, thus indicating that ROM is predicting the dynamics quite precisely.

As another basis to verify ROM's accuracy, we compare purities and recoveries of nitrogen and CO₂ for both rigorous model and ROM at CSS for this starting guess. Table 7.6 lists such a comparison. It can be observed that the values obtained from ROM are fairly close to the ones obtained after integrating the rigorous model even with relatively large $\lambda^* = 0.05$.



(a) Rigorous model simulation in MATLAB



(b) ROM simulation in AMPL

Figure 7.3: Comparison of CO₂ mole fraction for all the steps at the starting guess

Table 7.6: Comparison of rigorous model and ROM based on the performance variables

Performance variables	Rigorous model	ROM
N ₂ purity	92.57%	92.51%
N ₂ recovery	80.21%	80.71%
CO ₂ purity	37.76%	38.29%
CO ₂ recovery	66.27%	67.44%

Since the comparison is reasonable, we solve (7.31) with the POD-based ROM in Table 7.4 constructed and updated with every trust-region iteration with $\lambda^* = 0.05$.

7.4.3.2 Optimization with Zero-order Correction

We first solve (7.31) with the Zero-order Correction (7.8) applied for the objective function and the purity constraint. As mentioned before, ZOC can be computed cheaply as it doesn't involve evaluation of the original objective and the constraint gradients. However, it doesn't satisfy assumption **(A3)**, although it ensures **(A2)** holds. Therefore, we cannot ensure that ZOC satisfies the fraction of Cauchy decrease (7.14). Our focus in this section is to observe if the POD-based ROM, besides accurately predicting the dynamics, can also predict the descent direction accurately without involving the actual gradients from the original optimization problem. In other words, since POD-based ROMs are reasonably accurate at any x_k , we are inquisitive about whether they can also implicitly satisfy **(A3)** by their very construction, or whether a first-order correction (7.9) with accurate gradients is vital for ROM-based optimization for convergence to an optimum.

As mentioned before, we use a box trust-region in this work. For problem (7.31), with a current trust-region radius Δ_k , the shape of the box is defined as below

$$\begin{aligned}
 |P_h - P_{h,k}| &\leq \Delta_k, & |P_l - P_{l,k}| &\leq \Delta_k/4, & \text{(in kPa)} \\
 |t_p - t_{p,k}| &\leq \Delta_k/2, & |t_a - t_{a,k}| &\leq \Delta_k/2, & \text{(in sec)} \\
 |u_a - u_{a,k}| &\leq \Delta_k/4 & & & \text{(in cm/sec)}
 \end{aligned} \tag{7.32}$$

With penalty parameter $\mu = 1000$, we apply the exact penalty trust-region algorithm. Table B.1 in Appendix B lists all the trust-region iterations together with the values of the decision variables after the iteration ends (i.e., $x_k + s_k$), CO₂ purity p_{CO_2} , and recovery r_{CO_2} at $x_k + s_k$,

Table 7.7: Optimization results with ZOC

Problem size and computational time		
No. of variables	52247	
Total trust-region iterations	13	
Optimization CPU time	35.7 min	
Optimal parameters		
Adsorption pressure (P_h)	203.28 kPa	
Desorption pressure (P_l)	40 kPa	
Pressurization step time (t_p)	55.43 sec	
Adsorption step time (t_a)	161.08 sec	
Adsorption feed flow (u_a)	20.8 cm/s	
Comparison of performance variables		
	ROM (AMPL)	Rigorous model (MATLAB)
N ₂ purity	95.81%	95.99%
N ₂ recovery	85.85%	85.26%
CO ₂ purity	51.41%	50.01%
CO ₂ recovery	85.11%	81.74%

ROM-based objective function at the center of the trust-region $\tilde{\psi}_k^{\mathcal{R}}(0)$ and at the end of the iteration $\tilde{\psi}_k^{\mathcal{R}}(s_k)$ (see Equation (7.10)), the true objective function at the end of the iteration $\psi(s_k)$ (see Equation (7.12)), ratio ρ_k defined in Algorithm I, and the trust-region radius Δ_k . We observe that because of the penalty, the focus of the first few iterations of the algorithm is to gain feasibility, although additionally CO₂ recovery also improves. Feasibility is attained in 7th iteration ($k=6$), after which the penalty formulation is dropped and the purity constraint is moved from the objective function to the constraint section of the trust-region subproblem. Eventually, the algorithm terminates after 13th iteration ($k=12$) when Δ_{12} is shrunk further from 0.059 to 0.015, as it becomes less than the Δ_{min} of 0.02.

Optimal values of the decision variables together with the optimization CPU time are listed in Table 7.7. CPU time for optimization only accounts for the time taken for all the trust-region iterations, and doesn't include the time required to calculate snapshots and construct ROM. With 52,247 algebraic variables, algorithm terminated with 13 trust-region iterations and within a reasonable CPU time of 35.7 min. We notice that after algorithm termination, only P_l is at its lower bound, while other decision variables are well within their limits. We

Table 7.8: Perturbation results with ZOC

	Optimal value	Perturbed value	CO ₂ purity*	CO ₂ recovery*
P_h (kPa)	203.28	206.28	50.28%	82.01%
P_h (kPa)	203.28	200.28	49.72%	81.47%
P_l (kPa) [†]	40	39	50.44%	82.71%
t_p (sec)	55.43	57.43	49.97%	81.68%
t_p (sec)	55.43	53.43	50.04%	81.81%
t_a (sec)	161.08	164.08	50.14%	81.39%
t_a (sec)	161.08	158.08	49.87%	82.09%
u_a (cm/s)	20.8	21.8	50.32%	80.77%
u_a (cm/s)	20.8	19.8	49.65%	82.77%

*Optimal CO₂ recovery: 81.74%, CO₂ purity: 50.01%, [†]at bound

also report the purities and recoveries of nitrogen and CO₂ as obtained from AMPL after final optimization iteration, and from the rigorous model MATLAB simulation at the optimum. We recall that the ROM is constructed with a threshold tolerance $\lambda^* = 0.05$. With this λ^* , ROM compares fairly well with the rigorous model.

In order to verify if the algorithm terminated at an optimal point, we follow the analysis presented in section 6.5.4.1 and perturb decision variables from their optimal values. In particular, we provide a positive perturbation to variables at their upper bounds and negative to the ones at their lower bounds, and record the change in CO₂ purity and recovery. For decision variables not at their bounds, we provide both positive and negative perturbations. Since the CO₂ purity constraint is active at the termination, the termination point can be proven optimal if, with such perturbations, we either improve CO₂ recovery and simultaneously decrease its purity, or vice-versa is true for all the decision variables. Moreover, if we improve both CO₂ purity and recovery for the parameters at their bounds, termination point is optimal. However, if CO₂ purity and recovery both improve for a decision variable not at a bound, or if they both decrease for a variable at either of its bounds, the termination point is not optimal.

Perturbation results for this case are shown in Table 7.8. It can be observed that the CO₂ purity and recovery both improve when P_h is perturbed from 203.28 kPa to 206.28 kPa, and t_p is perturbed from 55.43 sec to 53.43 sec. This shows that the point at which algorithm

terminated is not optimal, which implies that ROM is not predicting the correct descent direction at the termination point. The reason for this is that the assumption **(A3)** is not true at the termination point, i.e., the objective function gradient and the constraint Jacobian obtained from ROM do not match the actual ones. Hence, this clearly shows that although we can construct substantially accurate ROMs based on the snapshot information of the state variables, such ROMs in general cannot always ensure that the gradient information is also reasonably accurate. Moreover, we infer that to ensure convergence to an optimal point, assumption **(A3)** is essential and accurate gradients should be incorporated in the ROM-based optimization problem. This can be accomplished with a First-order Correction (FOC), as illustrated in the next section.

7.4.3.3 Optimization with First-order Correction

We solve (7.31) with the First-order Correction (7.9) applied for the objective function and the purity constraint. For FOC, we need to evaluate gradients of the objective and the constraints of the original optimization problem with the rigorous model before each trust-region iteration starts. However, this is computed just once and optimization within a trust-region is carried out using ROM. For this case, we evaluate gradients using perturbation.

Table B.2 in Appendix B lists all the trust-region iterations with the penalty parameter $\mu=1000$. As in the previous case, we observe that because of the penalty parameter, the algorithm focuses on satisfying the CO₂ purity constraint for the first few iterations. In fact, within 5 iterations ($k=4$) feasibility is attained after which we drop the penalty parameter and move the purity constraint into the trust-region subproblem. Because of the exact gradient information, algorithm goes beyond the optimal CO₂ recovery of 81.74% obtained in the previous case, up to a recovery of 97.19%. However, the key observation is that the algorithm takes tiny steps to improve CO₂ recovery after achieving feasibility, and thus eventually takes 92 iterations to get to the optimum, which is considerably large. After 92nd iteration ($k=91$), Δ_{91} gets reduced from 0.021 to 0.005, thus going below $\Delta_{min} = 0.02$, and the algorithm terminates.

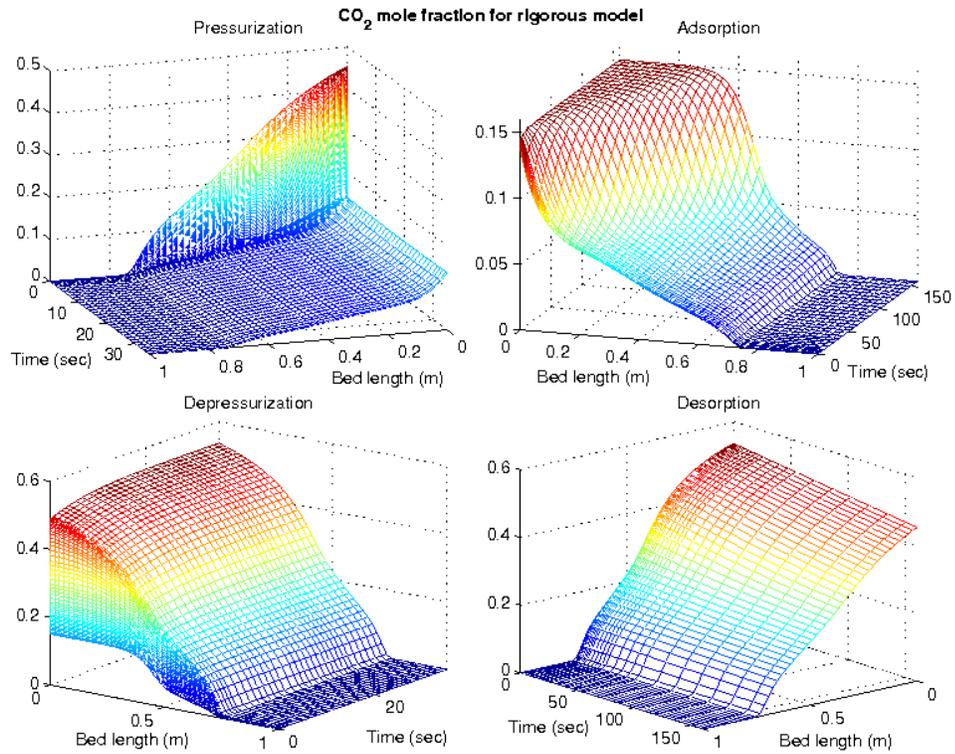
Table 7.9: Optimization results with FOC

Problem size and computational time		
No. of variables	52247	
Total trust-region iterations	92	
Optimization CPU time	1.88 hrs.	
Optimal parameters		
Adsorption pressure (P_h)	300 kPa	
Desorption pressure (P_l)	40 kPa	
Pressurization step time (t_p)	35 sec	
Adsorption step time (t_a)	160.97 sec	
Adsorption feed flow (u_a)	14.87 cm/s	
Comparison of performance variables		
	ROM (AMPL)	Rigorous model (MATLAB)
N ₂ purity	99.99%	99.99%
N ₂ recovery	82.36%	83.01%
CO ₂ purity	50.54%	50.01%
CO ₂ recovery	99.04%	97.19%

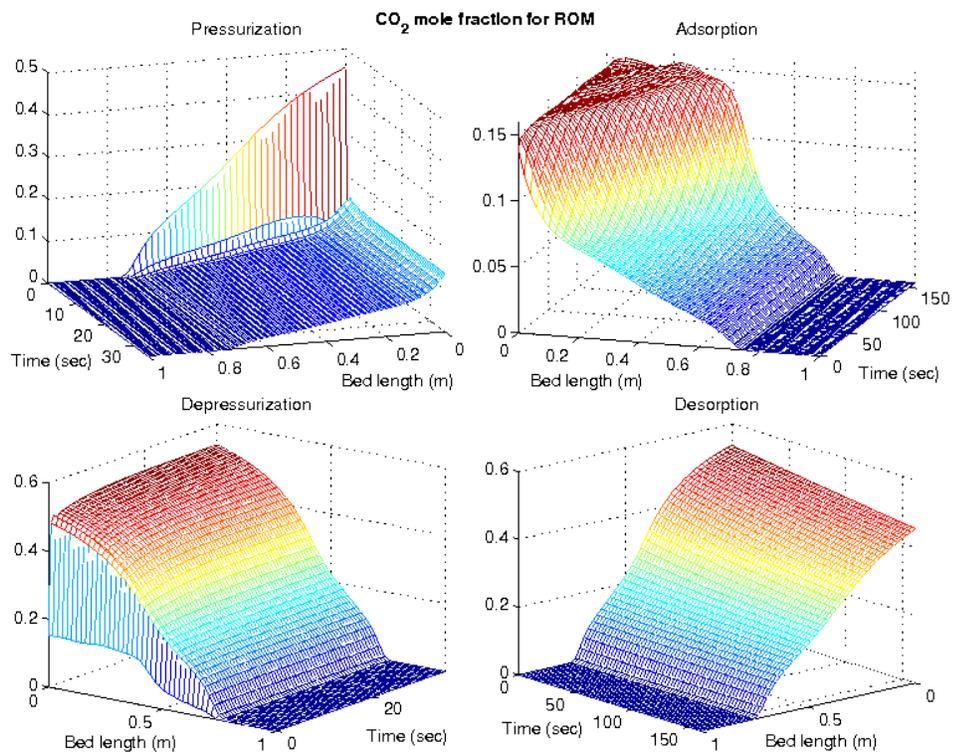
Final optimization statistics are listed in Table 7.9. Because of 92 trust-region iterations, the algorithm takes 1.88 hrs. of optimization CPU time to converge. Unlike previous case, three decision variables (P_h , P_l , and t_p) are at their bounds at the optimum, while the bounds of t_a , and u_a are not active. Purities and recoveries of the components obtained after final trust-region iteration in AMPL, are quite close to the ones obtained from MATLAB simulation of the rigorous model at the optimum.

We also perform perturbation-based analysis, as done in the previous case, in order to ensure if the algorithm converged at an optimal point. Giving a positive perturbation to P_h at its upper bound and negative perturbation to t_p at its lower bound improves both CO₂ purity and recovery, while a negative perturbation for P_l at its lower bound improves recovery but we lose feasibility. Moreover, perturbing t_a and u_a in both directions leads to a dichotomous behavior of either improvement in CO₂ recovery with loss in its purity, or vice-versa. Therefore, we conclude that the termination point is optimal.

To complete the analysis and to ensure ROM is predicting physically correct dynamic behavior of the system, we also compare gas-phase CO₂ mole fraction profiles at the optimum,



(a) Rigorous model simulation in MATLAB



(b) ROM optimization in AMPL

Figure 7.4: Comparison of CO₂ mole fraction for penalty TR algorithm with FOC

Table 7.10: Perturbation results with FOC

	Optimal value	Perturbed value	CO ₂ purity*	CO ₂ recovery*
P_h (kPa) [†]	300	303	50.16%	97.26%
P_l (kPa) [†]	40	39	49.83%	97.72%
t_p (sec) [†]	35	33	50.04%	97.31%
t_a (sec)	160.97	163.97	50.6%	96.69%
t_a (sec)	160.97	157.97	48.44%	97.56%
u_a (cm/s)	14.87	15.87	52.21%	95.12%
u_a (cm/s)	14.87	13.87	47.94%	98.26%

*Optimal CO₂ recovery: 97.19%, CO₂ purity: 50.01%, [†]at bound

obtained from ROM-based optimization in AMPL, and rigorous model simulation in MATLAB. Such a comparison in Figure 7.4 reveals convincing behavior of the reduced-order model as the profiles are nearly identical.

7.5 Hybrid Filter Trust-region Algorithm

7.5.1 Motivation

In this section we develop a hybrid filter-based trust-region algorithm for optimization involving ROMs. The algorithm is hybrid in the sense that it utilizes both ZOC and FOC at different times during its course.

The motivation to develop this algorithm comes from two key observations in the aforementioned PSA case study. First, we notice that although ROM-based trust-region subproblems with ZOC cannot ensure convergence to an optimal point, they are at least capable of generating descent directions and even attain feasibility even without the actual gradient information. Moreover, constructing FOC for every trust-region iteration is expensive as it involves computation of gradients from the original problem. However, FOC is necessary to converge to an optimal point. Therefore, we propose a hybrid strategy in which the trust-region algorithm begins with subproblems constructed with ZOC. This is continued until no further improvement is observed, after which the algorithm switches to subproblems with FOC until convergence.

Second, we observe in the optimization case study with FOC that the penalty-based trust-

region algorithm takes tiny steps and marches considerably slowly once feasibility is attained. The penalty parameter doesn't allow infeasible moves as $ared_k$, and thus ρ_k in this case becomes negative which entails sharp reduction in the trust-region radius and therefore, short steps. Hence, instead of developing a hybrid algorithm with the exact penalty algorithm, we develop a filter-based approach. Such an approach is desirable since it can allow taking a step which reduces objective while increasing infeasibility, and therefore can march faster towards the optimum. Moreover, the motivation for developing a filter-based algorithm comes from the difficulty of determining a suitable penalty parameter μ or its updation strategy in the exact penalty function.

Filter methods have been extensively studied and applied for nonlinear programming problems. Fletcher et al. [143] provide a brief survey of the literature on filter methods. Filter approach was first proposed by Fletcher in 1996; later described in [74]. The first global convergence proof for these methods was given for an SLP method [75], which was later generalized to SQP methods [76]. Fletcher et al. [72] analyze a trust-region SQP filter method that decomposes the SQP step into a normal step to attain feasibility, and a tangential step which reduced objective function. Nie et al. [135], on the other hand, combine the normal and tangential problem with a penalty parameter and solve them simultaneously in Fletcher's trust-region SQP filter method. Other studies with filter method include a bundle method for non-smooth optimization [73], and a pattern-search algorithm for derivative-free optimization [16]. Benson et al. [24] and Ulbrich et al. [187] have studied filter methods in the context of interior-point methods for solving NLPs. Wächter and Biegler [193, 194] have successfully incorporated filter mechanism in the NLP solver IPOPT [195]. They develop a line-search filter method that avoids convergence to arbitrary stagnation points, as illustrated by the example in [192].

In this work, we use Fletcher's trust-region filter method [72] with additional modifications for POD-based ROMs. The proposed modifications still enjoy the globally convergent properties of Fletcher's algorithm (see section 7.5.9).

7.5.2 Filter

There are two objectives in a general nonlinear programming problem, minimization of the objective function $f(x)$, and minimization of the constraint violation $\theta(x)$, where

$$\theta(x) = \|c_{\mathcal{E}}(x)\| + \|\max(0, c_{\mathcal{I}}(x))\|$$

A penalty function combines both these goals in one single measure and minimizes $\psi(x) = f(x) + \mu\theta(x)$. In contrast, a filter method considers both of these as separate goals, and interprets the NLP as a bi-objective optimization problem. There is a special emphasis on the second goal since a point has to be feasible in order to be an optimal solution, and thus $\theta(x^*)$ should be zero at the optimum x^* . Filter methods borrow the concept of domination from multiobjective optimization and state that a point x_k *dominates* x_l whenever

$$\theta(x_k) \leq \theta(x_l) \quad \text{and} \quad f(x_k) \leq f(x_l)$$

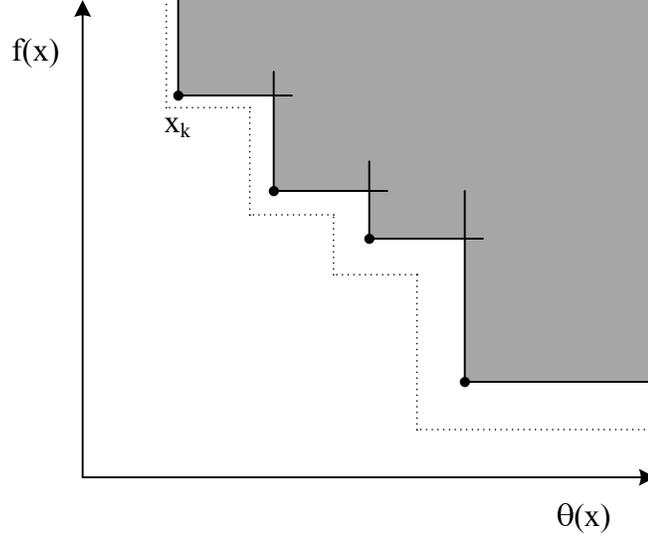
As a result, x_l is of no use as x_k is better in terms of both feasibility and optimality. A filter method involves storing iterates x_k that are not dominated by any other iterates. More precisely, a *filter* is a list \mathcal{F} of pairs (θ_i, f_i) such that

$$\text{either } \theta_i \leq \theta_j \quad \text{or} \quad f_i \leq f_j \quad \forall j, i \neq j$$

During optimization, we aim to accept a new iterate x_i only if it is not dominated by any other iterate in the filter. Figure 7.5 illustrates the concept by showing (θ_k, f_k) at x_k as black dots in the (θ, f) -space. The lines emanating from each (θ, f) -pair (forming filter envelope) indicate that any iterate whose associated (θ, f) -pair occurs in the shaded region in Figure 7.5 is dominated by at least one of these black dots. Iterates which do not lie in the shaded region are acceptable. The contours of the ℓ_1 exact penalty function will be straight lines with slope $-\mu$ in this plot, indicating that the filter is generally less restrictive than penalty methods.

We do not accept new iterate $x_k + s_k$ if its (θ, f) -pair is quite close to the filter envelope, and thus set a small “margin” around this envelope. Formally, we say that a point x is *acceptable for the filter* if and only if

$$\theta(x) \leq (1 - \gamma_\theta)\theta_j \quad \text{or} \quad f(x) \leq f_j - \gamma_f\theta_j \quad \forall (\theta_j, f_j) \in \mathcal{F} \quad (7.33)$$

Figure 7.5: A filter with four (θ, f) -pairs

for some $\gamma_f, \gamma_\theta \in (0, 1)$. In Figure 7.5, this margin corresponds to the thin dotted line. In practical implementation, γ_f, γ_θ are typically chosen to be small. As the algorithm proceeds, we keep adding (θ_k, f_k) -pairs to the filter for the acceptable iterates x_k . However, it is important to note that (θ, f) -pairs are not added to a filter for all the acceptable iterates. We observe that $\theta(x)$ in (7.33) dominates to a certain point, especially when infeasibility is large. However, as $\theta(x) \rightarrow 0$, the method must focus on descent of $f(x)$. For this case, no filter point is added (see further details in section 7.5.4). Since x_k may not be in the filter, we move from x_k to $x_k + s_k$ only if $x_k + s_k$ is “acceptable for the filter and x_k ”, i.e., if the following condition holds

$$\theta(x_k + s_k) \leq (1 - \gamma_\theta)\theta_j \text{ or } f(x_k + s_k) \leq f_j - \gamma_f\theta_j \quad \forall (\theta_j, f_j) \in \mathcal{F} \cup (\theta_k, f_k) \quad (7.34)$$

Maintaining a list of (θ, f) -pairs in a filter avoids what is known as cycling. Cycling results between two points that alternately improve one of the measures, θ or f , but worsen the other one at the same time. The filter avoids cycling because for a movement from x_k to x_{k+1} that improves θ but not f , the filter is augmented with x_k , and thus x_k becomes unreachable during optimization. Hence, cycling cannot occur with filter methods [74, 193].

7.5.3 Trust-region Step Computation

Following the strategy proposed by Fletcher et al. [72], we decompose the trust-region subproblem (7.7) into a normal subproblem and a tangential subproblem. In [72], the normal subproblem computes a step v_k which reduces infeasibility of (7.7), while the tangential subproblem evaluates a step p_k which improves the objective and lies in the null space of equality and inequality constraints, thus maintaining feasibility achieved by the normal problem. The overall step $s_k = v_k + p_k$. In our case, we compute a normal step v_k to determine a level δ up to which the infeasibility can be reduced in the given trust-region. Tangential subproblem, then, computes the overall step s_k which reduces the objective while maintaining this infeasibility level δ . Formally, to minimize the following maximum constraint violation

$$\tilde{\theta}^{\mathcal{R}}(x) = \delta = \max \left[0, \max_{i \in \mathcal{E}} |\tilde{c}_i^{\mathcal{R}}(x)|, \max_{i \in \mathcal{I}} \tilde{c}_i^{\mathcal{R}}(x) \right] \quad (7.35)$$

we write the following normal subproblem

$$\begin{aligned} \min_{v, \delta} \quad & \delta \\ \text{s.t.} \quad & -\delta \leq \tilde{c}_{\mathcal{E},k}^{\mathcal{R}}(x_k + v) \leq \delta \\ & \tilde{c}_{\mathcal{I},k}^{\mathcal{R}}(x_k + v) \leq \delta \\ & \delta \geq 0 \\ & x_L \leq x_k + v \leq x_U \\ & \|v\|_{\infty} \leq \Delta_c \end{aligned} \quad (7.36)$$

In order to ensure a non-zero tangential step, we choose $\Delta_c = 0.6\Delta_k$. Once the optimum infeasibility level δ is obtained, it is fixed to $\bar{\delta}$ and we solve the following tangential subproblem

$$\begin{aligned} \min_s \quad & \tilde{f}_k^{\mathcal{R}}(x_k + s) \\ \text{s.t.} \quad & -\bar{\delta} \leq \tilde{c}_{\mathcal{E},k}^{\mathcal{R}}(x_k + s) \leq \bar{\delta} \\ & \tilde{c}_{\mathcal{I},k}^{\mathcal{R}}(x_k + s) \leq \bar{\delta} \\ & x_L \leq x_k + s \leq x_U \\ & \|s\|_{\infty} \leq \Delta_k \end{aligned} \quad (7.37)$$

These subproblems are similar to the ones proposed by Alexandrov et al. [10] in their MAESTRO-AMMO algorithm. However, unlike our case, MAESTRO-AMMO takes the normal step v_k and solves the tangential subproblem at $x_k + v_k$ to obtain a tangential step p_k which reduced objective in the null space of the constraints. Hence, their overall step is a

composite step defined as $s_k = v_k + p_k$.

Unlike MAESTRO-AMMO, we do not take the normal step v_k and compute the overall step s_k in the tangential subproblem in order to eschew computational expense involved in constructing corrections (ZOC or FOC) for the objective and the constraints at $x_k + v_k$. If the normal step is taken, corrections will have to be constructed once for the normal subproblem at x_k and again for the tangential subproblem at $x_k + v_k$. This involves evaluating the values and derivatives of the objective and the constraints of the original problem twice. To circumvent this, we construct ZOC or FOC only once at x_k and solve both subproblems with x_k as the center of the trust-region.

We note that although Fletcher et al. [72] allow obtaining approximate solutions for normal and tangential problem, we solve both problems using IPOPT and compute exact local solutions in this work.

7.5.4 Switching Condition

Relying solely on the condition (7.34) can produce a situation when the sequence of iterates $\{x_k\}$ always provide *sufficient reduction* of the constraint violation alone, and not the objective function. This could result in convergence to a feasible but non-optimal point. In order to prevent this, we check the following *switching condition* during optimization

$$\tilde{f}_k^{\mathcal{R}}(x_k) - \tilde{f}_k^{\mathcal{R}}(x_k + s_k) \geq \kappa_\theta \theta_k^\psi \quad (7.38)$$

where $\tilde{f}_k^{\mathcal{R}}(x)$ is the ROM-based objective function of the tangential subproblem, and $\theta_k \stackrel{\text{def}}{=} \theta(x_k)$ is the current actual constraint violation, different from $\tilde{\theta}_k^{\mathcal{R}}$ in (7.35), and is defined as

$$\theta(x) = \max \left[0, \max_{i \in \mathcal{E}} |c_i(x)|, \max_{i \in \mathcal{I}} c_i(x) \right] \quad (7.39)$$

Note that θ_k is used for the filter margin (7.34) instead of $\tilde{\theta}_k^{\mathcal{R}}$ defined in (7.35). Current iterate x_k is added to the filter if condition (7.38) fails. The role of this switching condition can be interpreted as follows. If it fails, then the current constraint violation θ_k is significant and we aim to improve on this in the future by inserting x_k into the filter. On the other hand, if it

holds, then the reduction in the objective function predicted by ROM is more significant than the current θ_k , and the algorithm should promote descent in the objective. In this case, it is important that a sufficient decrease is also realized in the objective function of the original optimization problem. In other words,

$$\rho_k = \frac{ared_k}{pred_k} = \frac{f(x_k) - f(x_k + s_k)}{\tilde{f}_k^{\mathcal{R}}(x_k) - \tilde{f}_k^{\mathcal{R}}(x_k + s_k)} \geq \eta_1 \quad (7.40)$$

should hold together with condition (7.38). In the parlance of filter methods, a step generated in such a case is called an “f-type step”. With an f-type step, x_k is not added to the filter.

If an iterate x_k is feasible ($\theta_k = 0$), equation (7.38) becomes $\tilde{f}_k^{\mathcal{R}}(x_k) - \tilde{f}_k^{\mathcal{R}}(x_k + s_k) \geq 0$. Consequently, the filter mechanism is irrelevant if all iterates are feasible, and the algorithm reduces to a classical unconstrained trust-region method. Moreover, condition (7.38) ensures that no feasible iterate is ever included in the filter. This is vital to avoid convergence to a feasible but suboptimal point, and crucial in allowing finite termination of the feasibility restoration phase discussed further in section 7.5.7.

7.5.5 Algorithm Section I: Filter with ZOC

As mentioned before, our interest here is to develop a hybrid filter algorithm which utilizes benefits of both ZOC and FOC. Since ZOC is cheap to construct and can predict descent in objective or infeasibility even without accurate gradients, the algorithm begins in Section I with normal subproblem (7.36) and tangential subproblem (7.37) defined with ZOC, and proceeds until no further improvement in the objective or the infeasibility measure is obtained. After this, the algorithm moves to Section II where subproblems are constructed with FOC (discussed in the next section).

In Section I, instead of haphazardly solving normal and tangential subproblems, we verify if the ROM-based objective function and constraints constructed with ZOC can indeed provide a descent in either objective or infeasibility or both. Normal and tangential subproblems are solved only if we can guarantee descent. Since we have no information about accurate gradients

in this section, we rely on the gradients of the objective and the constraints obtained from the reduced-order model to promote such descent. Formally, we evaluate Cauchy steps (steepest descent direction) for both objective function and infeasibility as below

$$s_f^C = \frac{\alpha_f \nabla \tilde{f}_k^{\mathcal{R}}}{\|\nabla \tilde{f}_k^{\mathcal{R}}\|} \quad \tau_f = \begin{cases} 0 & \text{if } s_f^C = 0 \\ \frac{\tilde{f}_k^{\mathcal{R}}(x_k) - \tilde{f}_k^{\mathcal{R}}(x_k + s_f^C)}{\alpha_f \|\nabla \tilde{f}_k^{\mathcal{R}}\|} & \text{otherwise} \end{cases} \quad (7.41)$$

$$s_\theta^C = \frac{\alpha_\theta \nabla \tilde{\theta}_k^{\mathcal{R}}}{\|\nabla \tilde{\theta}_k^{\mathcal{R}}\|} \quad \tau_\theta = \begin{cases} 0 & \text{if } s_\theta^C = 0 \\ \frac{\tilde{\theta}_k^{\mathcal{R}}(x_k) - \tilde{\theta}_k^{\mathcal{R}}(x_k + s_\theta^C)}{\alpha_\theta \|\nabla \tilde{\theta}_k^{\mathcal{R}}\|} & \text{otherwise} \end{cases} \quad (7.42)$$

for some $\alpha_f, \alpha_\theta \in (0, 1)$. Here $\tilde{f}_k^{\mathcal{R}}$ is constructed with ZOC in equation (7.8), and $\tilde{\theta}_k^{\mathcal{R}}$ is given by equation (7.35) with $\tilde{c}_\varepsilon^{\mathcal{R}}(x)$, and $\tilde{c}_T^{\mathcal{R}}(x)$ given by equation (7.8). We note that since $\tilde{f}_k^{\mathcal{R}}$ and $\tilde{\theta}_k^{\mathcal{R}}$ are based on the ROM for k^{th} iteration, their gradients, and thus s_f^C and s_θ^C can be cheaply evaluated for each trust-region iteration. Hence, with a little computational expense, we can determine if the reduced-order model can predict descent for $\tilde{f}_k^{\mathcal{R}}$ or $\tilde{\theta}_k^{\mathcal{R}}$ or both. If $\tau_f > 0$ or $\tau_\theta > 0$, it can guarantee descent for the reduced objective function $\tilde{f}_k^{\mathcal{R}}$ or reduced infeasibility $\tilde{\theta}_k^{\mathcal{R}}$, respectively. Therefore, the normal subproblem with ZOC is solved only if $\tau_\theta > 0$, and similarly, the tangential subproblem with ZOC is solved only when $\tau_f > 0$. If both cannot be ensured, algorithm moves to Section II where FOC with exact gradients is used.

In Section I, we also incorporate POD subspace augmentation which involves adding more POD basis functions to improve accuracy of the existing ROM for the k^{th} iteration, and thus enhancing its ability to accurately predict the descent direction for $\tilde{f}_k^{\mathcal{R}}$ or $\tilde{\theta}_k^{\mathcal{R}}$ or both. To construct a reduced-order model, we choose the number of basis functions M by deciding on an error tolerance level λ^* . During the course of the algorithm, if we encounter a situation when both $\tau_f \leq 0$ and $\tau_\theta \leq 0$, tolerance level λ^* is reduced to increase the POD subspace dimension M and more basis functions are added to the ROM. This is repeated until either one of the τ_f or τ_θ or both become positive, or we hit the maximum allowable limit for POD subspace dimension M^{max} . Once we reach M^{max} , algorithm switches to Section II.

Since Section II involves expensive gradient computation, we desire to achieve larger reduction in the objective and infeasibility in Section I itself and delay switching to Section II. POD subspace augmentation not only allows such a delay, but also improves the performance of normal and tangential subproblems with ZOC by producing more accurate ROM. Note that if M^{max} is same as the number of spatial discretization nodes N_x , ROM is essentially as accurate as the original DAE system and Section I itself can be used to attain convergence to an optimum. However, we avoid choosing M^{max} as high as N_x as we lose all the computational advantage offered by ROMs. Usually an M^{max} is chosen which is reasonably high compared to current M but considerably low compared to N_x . In this sense, it is beneficial to resort to Section II with FOC and accurate gradients instead of utilizing more expensive ROMs.

Finally, we note that unless $M^{max} = N_x$, we cannot guarantee convergence to an optimal point within Section I itself. Therefore, our filter algorithm never terminates in Section I and at least one iteration of Section II is always executed.

7.5.6 Algorithm Section II: Filter with FOC

In Section II of our hybrid filter algorithm, normal subproblem (7.36) and tangential subproblem (7.37) are constructed with First-order Corrections for the objective and the constraints. This involves computing exact gradients for each trust-region iteration. Because FOC and the exact gradients can ensure proper descent direction, we do not calculate Cauchy steps for the objective and infeasibility as done in Section I. Moreover, we do not utilize POD basis augmentation strategy for this section as the ROMs, even with few basis functions M , can generate accurate steps with accurate original gradients. As a result, we can work with smaller ROMs which leads to computationally cheaper trust-region iterations compared to Section I. We note that once the algorithm proceeds from Section I to Section II, it never resorts back to Section I in the future course of optimization.

Section II is essentially the SQP-filter algorithm proposed by Fletcher et al. [72]. The difference lies in the fact that Fletcher et al. use a quadratic model approximation for their

trust-region (tangential) subproblems, while we use a POD-based ROM as a surrogate model. However, we ensure all the assumptions made by Fletcher et al. are satisfied, which renders our algorithm similar to the SQP-filter algorithm from convergence theory point of view (see further details in section 7.5.9).

7.5.7 Feasibility Restoration Phase

The algorithm switches to a *feasibility restoration phase* when it is not able to obtain an “admissible” step size. The step is not admissible:

- when the next iterate $x_k + s_k$ obtained after solving the normal and the tangential subproblem is not acceptable for the filter and x_k , or
- when $x_k + s_k$ is acceptable for the filter and x_k , and satisfies the switching condition (7.38) as well, but fails to provide sufficient reduction in the original objective function $f(x)$, and generates $\rho_k < \eta_1$.

Restoration phase is invoked when either the current trust-region radius Δ_k goes below Δ_{min} , when the current infeasibility level $\tilde{\theta}_k^{\mathcal{R}}$ goes beyond a maximum limit θ_{max} , or when in Section I, $\tau_f \leq 0$ together with $\tau_\theta > 0$, i.e., when ROM-based subproblems with ZOC can only decrease infeasibility but not the objective function.

The purpose of the restoration phase is to decrease the current constraint violation and generate a new iterate $x_k + s_k$ which is acceptable for the current filter and x_k . In our algorithm, restoration phase involves solving the normal subproblem using the basic trust-region algorithm [54] until such an iterate is obtained. Consequently, restoration phase can follow its own trust-region update rules separate from the ones used in the filter trust-region algorithm. In the terminology of filter methods, such an infeasibility reducing step is known as “ θ -type step” or “h-type step”. We define distinct restoration phases for Section I and Section II. Restoration phase in Section I solves the normal subproblem with ZOC, while that in Section II solves it with FOC. Moreover, restoration phase in Section I also involves POD subspace augmentation

and evaluation of the Cauchy step τ_θ to determine if a descent direction exists for infeasibility with current ROM incorporating M_k basis functions for the current iterate k .

We note that the iterates generated during the restoration phase are never added to the filter as it can also lead to the addition of a feasible point to the filter which is detrimental for the algorithm. But we also note that whenever restoration phase is invoked at an iterate x_k , it is augmented to the filter to avoid a visit to this point again in the future. Since a feasible iterate is never included in the filter, feasibility restoration phase always either generates a successful iterate, or converges to a local minimizer with some measure of infeasibility, indicating that the original optimization problem might be infeasible.

7.5.8 Trust-region Algorithm

Algorithm II: Hybrid filter trust-region algorithm

Choose $0 < \eta_1 \leq \eta_2 < 1 \leq \eta_3$, $0 < \gamma_1 \leq \gamma_2 < 1 < \gamma_3$, $\gamma_f, \gamma_\theta \in (0, 1)$, $\kappa_\theta \in (0, 1)$, $\beta \in (0, 1)$, $\psi > 1/(1 + \beta)$, $\alpha_f, \alpha_\theta \in (0, 1)$, an initial trust-region radius Δ_0 , minimum radius Δ_{min} , and an initial iterate x_0 . Set $k = 0$.

Section I: Filter with ZOC

1. *ROM construction*: Compute POD basis functions using the snapshots obtained at x_k .
Choose λ^* , obtain corresponding M_k and construct a reduced-order model.
2. *Cauchy step computation*
 - (a) Compute $s_f^C, s_\theta^C, \tau_f$, and τ_θ .
 - (b) If $\tau_f > 0$, go to Step 3.
 - (c) If $\tau_\theta > 0$, add x_k to the filter and go to the restoration phase in Step 4.
 - (d) If $M_k < M^{max}$, decrease λ^* to update M_k and ROM, and repeat Step 2, else, go to Step R.
3. *Step computation*

- (a) Solve normal subproblem (7.36) and tangential subproblem (7.37) with ZOC.
- (b) If $\theta_k \geq \theta_{max}$, or if $\Delta_k \leq \Delta_{min}$, add x_k to the filter and go to the restoration phase in Step 4.
- (c) If $x_k + s_k$ is not acceptable for the filter and x_k , i.e., (7.34) fails, set $x_{k+1} = x_k$ and $\Delta_{k+1} = \gamma_1 \Delta_k$. Increment k by 1 and repeat Step 3.
- (d) Compute ρ_k from (7.40).
- (e) If $pred_k < 0$ and $ared_k > 0$, go to 3(g).
- (f) If (7.38) holds and $\rho_k < \eta_1$, set $x_{k+1} = x_k$ and $\Delta_{k+1} = \gamma_1 \Delta_k$. Increment k by 1 and repeat Step 3.
- (g) If (7.38) fails, add x_k to the filter.
- (h) Set $x_{k+1} = x_k + s_k$. If (7.38) fails, $\Delta_{k+1} = \Delta_k$, else

$$\Delta_{k+1} = \begin{cases} \gamma_2 \Delta_k & \text{if } \rho_k \in [\eta_1, \eta_2), \\ \Delta_k & \text{if } \rho_k \in [\eta_2, \eta_3), \\ \gamma_3 \Delta_k & \text{if } \rho_k \geq \eta_3 \end{cases}$$

Increment k by 1 and go to Step 1.

4. Restoration with ZOC

- (a) With M_k , solve normal subproblem (7.36) with ZOC using a basic trust-region algorithm until a point acceptable for the filter and x_k is found. If found, increment k by 1 and go to Step 1, else continue.
- (b) If $M_k \geq M^{max}$, go to Step R.
- (c) Decrease λ^* to update M_k and ROM. Compute τ_θ .
 - i. If $\tau_\theta > 0$, repeat Step 4.
 - ii. If $M_k \geq M^{max}$, go to Step R, else repeat 4(c).

Section II: Filter with FOC

- R. Reinitialize Δ_k , λ^* and corresponding M_k . Construct ROM with updated M_k and go to Step 6.
5. *ROM construction*: Compute POD basis functions using the snapshots obtained at x_k . Choose λ^* , obtain corresponding M_k and construct a reduced-order model.
6. *Step computation*
- (a) Solve normal subproblem (7.36) and tangential subproblem (7.37) with FOC.
 - (b) If $\theta_k = 0$, $v_k = 0$, and $p_k = 0$, or $\theta_k = 0$ and $\Delta_k \leq \Delta_{min}$, STOP.
 - (c) If $\theta_k \geq \theta_{max}$, or if $\Delta_k \leq \Delta_{min}$, add x_k to the filter and go to the restoration phase in Step 7.
 - (d) If $x_k + s_k$ is not acceptable for the filter and x_k , i.e., (7.34) fails, set $x_{k+1} = x_k$ and $\Delta_{k+1} = \gamma_1 \Delta_k$. Increment k by 1 and repeat Step 6.
 - (e) Compute ρ_k from (7.40).
 - (f) If (7.38) holds and $\rho_k < \eta_1$, set $x_{k+1} = x_k$ and $\Delta_{k+1} = \gamma_1 \Delta_k$. Increment k by 1 and repeat Step 6.
 - (g) If (7.38) fails, add x_k to the filter.
 - (h) Set $x_{k+1} = x_k + s_k$. If (7.38) fails, $\Delta_{k+1} = \Delta_k$, else update Δ_{k+1} as in 3(h). Increment k by 1 and go to Step 5.
7. *Restoration with FOC*: Solve normal subproblem (7.36) with FOC using a basic trust-region algorithm until a point acceptable for the filter and x_k is found. If found, increment k by 1 and go to Step 5, else STOP at an infeasible point.

Clearly, algorithm begins in Section I and always terminates in Section II. The choice of the constants in the algorithm depends on the optimization problem and the scaling mechanism used for the decision variables. In this work, we choose the following values:

$$\begin{aligned} \eta_1 &= 0.05, \eta_2 = 0.5, \eta_3 = 1, \gamma_1 = 0.25, \gamma_2 = 0.5, \gamma_3 = 1.5, \\ \gamma_f &= \gamma_\theta = 0.01, \kappa_\theta = 0.01, \psi = 0.09, \alpha_f = \alpha_\theta = 0.02 \end{aligned}$$

As in the case of exact penalty trust-region algorithm, we choose a small value for η_1 to allow taking a step even if the reduction in $f(x)$ is quite small. Since computation of $f(x_k + s_k)$ in ρ_k involves evaluation of new snapshots from the original DAEs, which can be used to update ROM at $x_k + s_k$, it is always beneficial to move to $x_k + s_k$ and expect the new ROM to predict a better descent step. Also, as in the penalty trust-region case, we choose $\eta_2 = 0.5$ and $\eta_3 = 1$ to maintain the trust-region for longer duration because of the oscillatory behavior of the ROM for large trust-regions.

One peculiar feature of the algorithm is step 3(e) in Section I. Even though $pred_k < 0$, this step allows us to move from x_k to $x_k + s_k$ because of the $ared_k$ being positive. Such a scenario is possible especially with ROM-based trust-region subproblems without exact gradient information. In particular, we encounter this situation when the normal and the tangential subproblems focus more on reducing infeasibility, leading to an increase in the objective function $\tilde{f}_k^{\mathcal{R}}$ which causes $pred_k$ to become negative. However, such an iterate can actually decrease both infeasibility and objective for the original optimization problem, leading to a positive $ared_k$. Inaccurate gradients in the ROM-based tangential subproblem with ZOC entails $pred_k$ to become negative. Consequently, ρ_k becomes negative ($< \eta_1$). If we move from 3(e) to 3(f), a step will be denied which is undesirable as $ared_k > 0$. Hence, we jump from 3(e) to 3(g). Note that a counterpart of 3(e) is missing in Section II since such a scenario cannot occur as a consequence of the availability of exact gradient information in this section.

Another important feature of the algorithm is that in both sections, trust-region radius is updated only when (7.38) holds. If (7.38) fails, the main effect of the current iteration is not to reduce objective (which makes ρ_k essentially irrelevant), but rather to reduce constraint violation (which is taken care of by inserting x_k to the filter in steps 3(g) and 6(g)). In this case, we impose no further restriction on Δ_{k+1} and keep it same as Δ_k because reducing Δ_{k+1} might cause steps towards infeasibility that are too small, or an unnecessary call for the restoration phase. If, on the other hand, (7.38) holds, iteration's emphasis is on reducing the objective and Δ_{k+1} is updated in the conventional way.

Finally, the performance of the algorithm depends significantly on the quality of the reduced-order model constructed. Highly accurate ROMs with sufficiently accurate gradients can quickly approach close to the optimum within Section I itself. On the other hand, ROMs which poorly predict the actual dynamic behavior and portray inaccurate gradients can end up landing in Section II quite early during optimization, making the whole process computationally demanding. Moreover, size of the trust-region and as a result, total number of iterations also rely significantly on ROM accuracy. POD-based ROMs can be made more accurate by adding more basis functions. However, it can also lead to ill-conditioned ROMs due to addition of those basis functions which do not affect the dynamics much. Hence, ROM-construction is the most important part of this algorithm.

7.5.9 Convergence and Optimality

Section II of the Algorithm II is essentially the SQP-filter algorithm proposed by Fletcher et al. [72]. Therefore, if we make the following assumptions

(AD) The sequence of iterates $\{x_k\}$ produced by Algorithm II lies within a closed, bounded domain Ω .

(AR) If $\{x_{k_i}\}$ is any subsequence of iterates for which $\lim_{i \rightarrow \infty} \theta_{k_i} = 0$, then a normal step v_{k_i} exists for i sufficiently large, and $\|v_{k_i}\| \leq \kappa_v \theta_{k_i}$ for some $\kappa_v > 0$.

the following convergence property holds.

Theorem 7.5.1. (See Theorem 15.5.13 in [54]) Suppose that **(AF1)**–**(AF3)**, **(A1)**–**(A4)**, **(AD)**, and **(AR)** hold and the fraction of Cauchy decrease (FCD) condition

$$\tilde{f}_k^{\mathcal{R}}(x_k + v_k) - \tilde{f}_k^{\mathcal{R}}(x_k + s_k) \geq \kappa_f \chi_k \min \left[\frac{\chi_k}{\beta_k}, \Delta_k \right] \quad (7.43)$$

is satisfied for some $\kappa_f > 0$, and a bounded sequence of $\beta_k > 1$. Then either the restoration procedure terminates unsuccessfully by converging to an infeasible first-order critical point of the normal subproblem (7.36), or there is a subsequence $\{k_j\}$ for which

$$\lim_{j \rightarrow \infty} x_{k_j} = x^* \quad (7.44)$$

and x^* is a first-order critical point for problem (7.1).

As discussed before in the case of Algorithm I, assumptions **(AF1)**–**(AF3)**, **(A1)**, and **(A4)** are assumed to be true in this work. Moreover, with the First-order Correction (FOC), we ensure that assumptions **(A2)** and **(A3)** are satisfied. Also, since $x_L \leq x \leq x_U$, **(AD)** is also guaranteed. Assumption **(AR)** requires existence of a normal step especially when the current constraint violation $\theta_{k_i} \stackrel{\text{def}}{=} \theta(x_{k_i})$, defined by (7.39), is sufficiently small. For Algorithm II, **(AR)** is satisfied by the construction of the normal subproblem (7.36) and by assuming that the gradients of the constraints are linearly independent. Since it is solved using a basic trust-region algorithm with exact gradients due to FOC, existence of a nonnegative step v_k together with a fraction of Cauchy decrease can always be ensured unless $\theta_{k_i} = 0$.

For FCD condition, χ_k is a first-order criticality measure. Based on the tangential problem (7.37), we define $\chi_k \stackrel{\text{def}}{=} \chi(x_k)$ in the following manner

$$\begin{aligned} \chi_k = & \min_d \quad |\nabla \tilde{f}_k^{\mathcal{R}}(x_k)^T d| \\ \text{s.t.} \quad & -\bar{\delta} \leq \tilde{c}_{i,k}^{\mathcal{R}}(x_k) + \nabla \tilde{c}_{i,k}^{\mathcal{R}}(x_k)^T d \leq \bar{\delta} \quad i \in \{\mathcal{E}\} \\ & \tilde{c}_{i,k}^{\mathcal{R}}(x_k) + \nabla \tilde{c}_{i,k}^{\mathcal{R}}(x_k)^T d \leq \bar{\delta} \quad i \in \{\mathcal{I}\} \\ & \|d\| \leq 1 \end{aligned} \quad (7.45)$$

where $\bar{\delta}$ is the optimum infeasibility level obtained from the following normal subproblem.

$$\begin{aligned} \min_{q, \delta} \quad & \delta \\ \text{s.t.} \quad & -\delta \leq \tilde{c}_{i,k}^{\mathcal{R}}(x_k) + \nabla \tilde{c}_{i,k}^{\mathcal{R}}(x_k)^T q \leq \delta \quad i \in \{\mathcal{E}\} \\ & \tilde{c}_{i,k}^{\mathcal{R}}(x_k) + \nabla \tilde{c}_{i,k}^{\mathcal{R}}(x_k)^T q \leq \delta \quad i \in \{\mathcal{I}\} \\ & \|q\| \leq 1 - \vartheta, \quad \delta \geq 0 \end{aligned} \quad (7.46)$$

Here $\vartheta > 0$ ensures (7.45) remains feasible. We note that χ_k can be defined in terms of $\tilde{f}_k^{\mathcal{R}}(x_k)$, $\tilde{c}_{\mathcal{E},k}^{\mathcal{R}}(x_k)$, and $\tilde{c}_{\mathcal{I},k}^{\mathcal{R}}(x_k)$, and their gradients since they match the original objective function and constraints, and their gradients at x_k because of the FOC. In order to use χ_k in (7.45) for the FCD condition, we need to show that it is a first-order criticality measure. Since the constraint set of (7.45) is linear, and thus convex, the following theorem ensures that χ_k is a first-order

criticality measure.

Theorem 7.5.2. (See Theorem 12.1.6 in [54]) Suppose that **(AF1)**, **(A2)**, and **(A3)** hold and x_k belongs to a nonempty, closed and convex feasible region. Then $\chi(x_k)$ defined by (7.45) is a first-order criticality measure, in the sense that it is a nonnegative, continuous function of x_k , and $\lim_{k \rightarrow \infty} \chi_k = 0$ if and only if $x_k \rightarrow x^*$.

In other words, we can always compute a Cauchy descent direction if $\chi(x_k) > 0$, and $\chi(x_k)$ vanishes only when x_k is a first-order critical point. Therefore, as the trust-region gets smaller, the linear part of the objective and the constraints dominate and thus, a Cauchy step can always be taken to ensure FCD condition (7.43) is satisfied. Also, because of FOC in the Section II of Algorithm II, the Cauchy step of the tangential problem (7.37) coincides with that of problem (7.37) with the original objective and constraints. Hence, Section II satisfies Theorem 7.5.1, and thus converges to the first-order critical point if the restoration procedure terminates successfully. Moreover, Algorithm II never terminates in Section I and always reaches Section II. Hence, Algorithm II is globally convergent and always converges to the exact local optimum of the original optimization problem. In order to verify optimality of the termination point of Algorithm II, we conduct a perturbation analysis as done with the exact penalty trust-region algorithm.

7.6 PSA Case Study Revisited

We demonstrate Algorithm II for the 2-bed 4-step PSA case study for post combustion CO₂ capture, and utilize it to solve the optimization problem (7.31) with same five decision variables. The algorithm begins at the same initial guess as shown in Table 7.5. At this initial guess, ROM is constructed with a threshold error tolerance λ^* of 0.05, similar to the exact penalty function case study, which yields $M = 2, 4, 1,$ and 3 for pressurization, adsorption, depressurization, and desorption steps, respectively. For Section II, gradients are evaluated using perturbation.

Table B.3 in Appendix B lists the trust-region iterations for the tangential subproblem,

Table 7.11: Optimization results with hybrid filter trust-region algorithm

Problem size and computational time		
No. of variables	52247	
Total trust-region iterations	51	
Optimization CPU time	1.36 hrs.	
Optimal parameters		
Adsorption pressure (P_h)	300 kPa	
Desorption pressure (P_l)	40 kPa	
Pressurization step time (t_p)	35 sec	
Adsorption step time (t_a)	187.91 sec	
Adsorption feed flow (u_a)	12.77 cm/s	
Comparison of performance variables		
	ROM (AMPL)	Rigorous model (MATLAB)
N ₂ purity	99.99%	99.99%
N ₂ recovery	83.16%	82.97%
CO ₂ purity	50.29%	50.01%
CO ₂ recovery	98.43%	97.26%

while Table B.4 lists those for the normal subproblem. Table B.3 shows decision variables at $x_k + p_k = x_k + s_k$, while Table B.4 shows decisions at $x_k + v_k$ corresponding to the iteration k . However, CO₂ purity and recovery is listed at $x_k + s_k$ for both tables, since it is observed that for all k , CO₂ purity at $x_k + s_k$ remains same as that at $x_k + v_k$. The algorithm begins with the restoration phase of Section I since the inaccurate gradients of ROM yield negative τ_f . Restoration phase is invoked for every iteration until $k = 3$, after which feasibility is attained and CO₂ purity goes beyond 50%. After $k = 3$, since $\tau_\theta = 0$, and ROM yield a $\tau_f < 0$, POD basis augmentation is used. When λ^* is reduced up to 0.1%, we obtain $\tau_f > 0$ and thus, proceed for step computation in Section I for 5th iterate ($k = 4$). Algorithm continues in Section I after this and improves objective until 35th iterate ($k = 34$). At $k = 34$, ROM is not able to predict a descent in the objective function even after increasing POD subspace dimension. Hence, algorithm switches to Section II. Eventually, the algorithm terminates after 51st iteration when Δ_{50} shrinks from 0.031 to 0.008 and goes below Δ_{min} of 0.02. One of the key observations in Table B.3 is the value of t_p , which increases steadily in Section I but starts decreasing and hits its lower bound after $k = 34$, when the algorithm switches to

Section II. This implies that $\frac{\partial \tilde{f}_k^R}{\partial t_p}$ has an opposite sign in Section I during optimization which gets corrected in Section II. Moreover, since P_h hit its upper bound and Section I terminates during same iteration ($k = 34$), we conclude that CO₂ recovery improves in Section I even with incorrect $\frac{\partial \tilde{f}_k^R}{\partial t_p}$ due to the increment obtained in P_h . Another key observation is the 31st iteration ($k = 30$) when we take a step despite ρ_k being negative. This is a consequence of the step 3(e) in the algorithm. In this iteration, although $pred_k < 0$, we observe $ared_k > 0$.

As highlighted before, once feasibility is attained in the exact penalty trust-region algorithm, we are not allowed a move which increases infeasibility. This eventually causes Algorithm I to pursue tiny steps towards optimum. In contrast, Algorithm II allows such a step which can increase infeasibility when it tries to improve objective function value. For instance, in Table B.4, we notice iterations $k=28$, or 30, or iterations after $k = 33$, when algorithm sacrifices feasibility in order to achieve greater improvement in CO₂ recovery. Hence, Algorithm II takes fewer iterations to optimum than Algorithm I.

Table 7.11 lists the optimal values of the decision variables together with the optimization CPU time. With 52,247 algebraic variables, Algorithm II terminated within a reasonable CPU time of 1.36 hrs. As observed in the case of exact penalty trust-region algorithm with FOC, P_h , P_l , and t_p are at their bounds at the optimum. However, the local optimum obtained in this case is slightly different from the one obtained in section 7.4.3.3 in the sense that the values of t_a and u_a are marginally different. As a consequence, optimal CO₂ recovery obtained in this case is marginally better than the one obtained in section 7.4.3.3. We also report the purities and recoveries of nitrogen and CO₂ obtained from AMPL after final optimization iteration, and from the rigorous model MATLAB simulation at the optimum. The values are fairly close with no appreciable difference.

Table 7.12 lists the results for the perturbation analysis performed in order to validate if the algorithm terminated at an optimal point. A positive perturbation for P_h at its upper bound and a negative perturbation for t_p at its lower bound improves both CO₂ purity and recovery, while a negative perturbation for P_l at its lower bound improves recovery but deteriorates CO₂

Table 7.12: Perturbation results for Algorithm II

	Optimal value	Perturbed value	CO ₂ purity*	CO ₂ recovery*
P_h (kPa) [†]	300	303	50.13%	97.31%
P_l (kPa) [†]	40	39	49.82%	97.63%
t_p (sec) [†]	35	33	50.03%	97.34%
t_a (sec)	187.91	190.91	50.49%	96.77%
t_a (sec)	187.91	184.91	49.51%	97.57%
u_a (cm/s)	12.77	13.77	52.53%	94.94%
u_a (cm/s)	12.77	11.77	47.61%	98.48%

*Optimal CO₂ recovery: 97.26%, CO₂ purity: 50.01%, [†]at bound

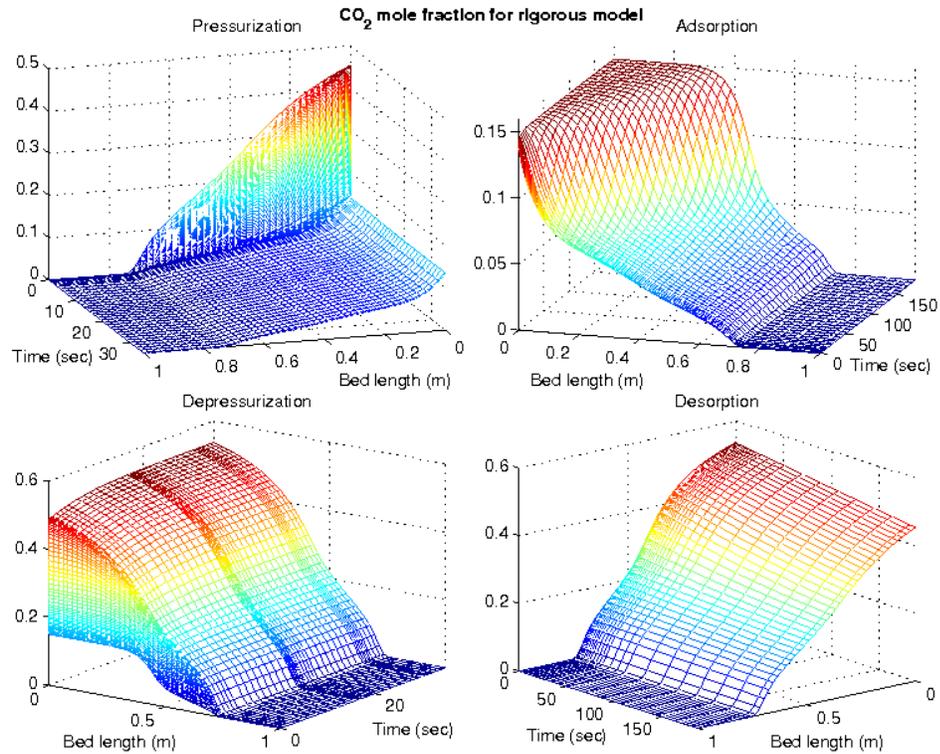
purity. Moreover, perturbing t_a and u_a in both directions either improves CO₂ recovery while diminishing its purity, or vice-versa. Therefore, we can safely conclude that the algorithm converged to a local optimum.

Finally, in Figure 7.6, we also present a comparison between the gas-phase CO₂ mole fraction profiles at the optimum, obtained from the final valid tangential subproblem iteration, and from the rigorous model simulation in MATLAB. Profiles are nearly identical which confirms that ROM is predicting physically correct dynamic behavior of the system during optimization.

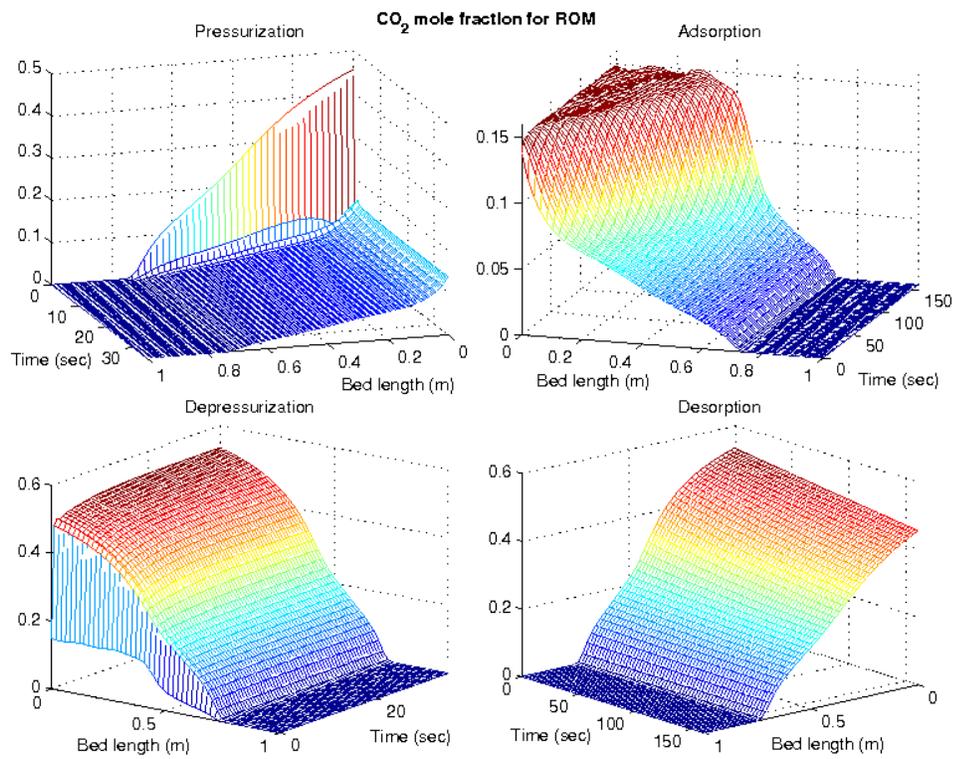
7.7 Conclusions

Trust-region based methodology provides an excellent adaptive framework to systematically utilize reduced-order models for optimization since it not only restricts the validity zone of the reduced-order model, but also provides a robust and globally convergent algorithm. Therefore, we develop trust-region based algorithms and explore both exact penalty-based and filter-based approaches to handle general equality and inequality constraints in the original optimization problem.

First, exact penalty trust-region algorithm is demonstrated for a 2-bed 4-step isothermal PSA process. We illustrate that executing the algorithm with only Zero-order Correction cannot ensure convergence to the local optimum of the original optimization problem, and thus conclude that FOC with exact gradient information is necessary for convergence. With



(a) Rigorous model simulation in MATLAB



(b) ROM optimization in AMPL

Figure 7.6: Comparison of CO₂ mole fraction for hybrid filter TR algorithm

FOC, exact penalty algorithm converges to a local optimum after 92 trust-region iterations and 1.88 hrs of optimization CPU time. Although not so encouraging, these results and success with this case study enables us to conclude that we can indeed perform optimization using reduced-order models with the help of a systematic trust-region based adaptive strategy.

We find that the reason for this high iteration count is early attainment of feasibility which further doesn't allow infeasible moves and thus, tightens the step size. One reason for this might be the choice of high penalty for constraint violation ($\mu = 1000$). This reflects one of the main issues with penalty functions, i.e., to find a reasonable value for μ . An updating scheme can be developed which penalizes constraint violation based on its magnitude; however, it may not be straight-forward. Instead, we develop a filter-based trust-region framework since it not only avoids such difficult decisions of choosing μ , but also allows steps which can achieve greater reduction in the objective by increasing infeasibility in a controlled manner. When applied to the PSA case study, filter trust-region algorithm converges to a local optimum within 51 trust-region iterations consuming 1.36 hrs of CPU time., which is significantly less compared to the penalty algorithm.

Since trust-region subproblems with ZOC can also generate descent due to accurate ROMs, we follow a hybrid strategy for filter-based algorithm. Moreover, we also incorporate POD basis augmentation in Section I to improve ROM's accuracy. For the PSA case study, we observe that 35 iterations out of the total 51 are indeed carried out in Section I of the algorithm, which is quite encouraging as it delays expensive gradient evaluations for FOC. Thus, we infer that a hybrid strategy and POD subspace augmentation are potentially useful tools for optimization with ROMs.

Finally, success of the idea of using ROMs for computationally efficient optimization ultimately depends on the quality of the ROM and its ability to accurately predict the descent direction. Although we obtain promising results in this chapter, iteration count for both penalty and filter approaches can further be reduced by improving the quality of ROMs. In future, alternative methodologies can be explored to construct better and more efficient ROMs.

Chapter 8

Conclusions

Synopsis

With growing demands for efficient PSA cycles, and increasing needs for computationally cheap modeling techniques, especially for flowsheet simulation and optimization, it has become essential to develop novel systematic optimization-based strategies for design and operation of PSA systems. In this dissertation, we not only introduce a novel idea of synthesizing PSA cycles using a superstructure, but also successfully demonstrate it for practical applications. Moreover, we address the challenging nature of PSA optimization problems by developing a new optimization framework using reduced-order modeling, which when applied to PSA optimization problems yields promising results. All these developments and our contributions are summarized in the next section, followed by directions for future work.

8.1 Thesis Summary and Contributions

This dissertation primarily focuses on introducing and developing two new ideas to address research challenges presented by PSA processes in terms of cycle synthesis and computational complexity of the PDAEs governing its dynamics, and presents a successful proof of principle analysis for both ideas. Beginning with an overview of the PSA processes and adsorption fundamentals in the first two chapters, we describe that a practical PSA/VPSA process can be fairly complex with a multicolumn design executing a wide variety of non-steady-state operating steps in a non-trivial sequence, and motivate the need for a systematic methodology to synthesize PSA cycles. Therefore, we first explore the idea of development of a unique PSA superstructure to design optimal PSA processes. Secondly, we show that PSA processes are governed by highly nonlinear PDAEs with solution profiles characterized by steep adsorption fronts. As a result, PSA optimization problems present a significant computational challenge to current optimization techniques. Consequently, we explore the idea of using POD to generate computationally-efficient ROMs and actualize novel trust-region algorithms to solve PSA optimization problems using these ROMs. We provide a summary of the work done and discuss our contributions separately in the subsequent sections.

PSA Superstructure

In Chapter 3, we present a new and original PSA superstructure to simultaneously determine new cycle configurations and design parameters. Interconnections between the two beds of the superstructure are governed by time-dependent control variables, which are manipulated to accomplish a wide variety of different PSA operating steps. An optimal cycle is eventually obtained by solving an optimal control problem for the superstructure. To solve it, we adopt a complete discretization approach, and alleviate its singular nature by using coarse discretization for controls.

The superstructure approach is illustrated for a post-combustion CO₂ capture case study. Superstructure is optimized to maximize CO₂ recovery. With the optimal 2-bed 6-step VSA

cycle, we are able to recover about 80% of CO₂ at a substantially high purity of 95%, and at a significantly high feed flux of 80 kgmol m⁻² hr⁻¹. Next, we develop an optimal configuration which yields high-purity separation with minimal power requirements. Optimal profiles translate in a 2-bed 8-step VSA configuration which, at 90% purity and 85% recovery, extracts CO₂ with a substantially low power consumption of 465 kWh/tonne CO₂ captured. We also apply the superstructure methodology for pre-combustion CO₂ capture in Chapter 5. When CO₂ recovery is maximized, superstructure optimization results in a 2-bed 8-step VSA cycle which can produce both H₂ and CO₂ at a substantially high purity of 98% and 90%, respectively. Changing the objective to minimizing power consumption yields an entirely different 2-bed 10-step VSA cycle which can produce CO₂ at a purity of 90% and a recovery of 92% with a significantly low power consumption of 46.82 kWh/tonne CO₂ captured. Our contributions for this part of the dissertation are as below:

- **First systematic methodology for cycle design**

All the studies in the literature so far only suggest simplistic formulations to determine minimum number of beds required in a PSA process with a given fixed sequence of operating steps. To the best of our knowledge, this is the first instance when a systematic methodology is proposed to design, evaluate and optimize PSA processes, and the first instance when a PSA superstructure is successfully developed and demonstrated to determine an optimal sequence of operating steps for a given number of beds.

- **PSA as a potential technology for CO₂ capture**

By developing cycles that can extract CO₂ at a purity of over 95% for post-combustion capture, and with a power consumption as low as 46.82 kWh/tonne CO₂ captured for pre-combustion capture, we successfully project PSA as a promising and viable technology for both post-combustion and pre-combustion carbon capture. We not only synthesize cycles which are practically feasible, but also suggest operating steps which should be incorporated in a PSA process for high-purity CO₂ capture. More importantly, we discover novel operating steps such as the total reflux step which have never been seen before in

the PSA literature.

- **Generic framework**

The key accomplishment is that the proposed superstructure framework is quite generic and can be extended to many other PSA applications. We do not make any assumption on the adsorbent or the feedstock, the kinds of operating steps that can be predicted, or details of the bed model. Moreover, we do not impose any upper bound on the number of operating steps eventually included in the optimal PSA cycle. This makes the approach fairly general. Also, besides developing optimal cycles, the framework can be used to evaluate different kinds of adsorbents for the same feedstock and process conditions.

ROM-based Optimization

In Chapter 6, with the help of the method of snapshots and Galerkin projection, we utilize proper orthogonal decomposition (POD) to successfully construct ROMs which are orders of magnitude smaller than the original problem and also, significantly accurate. Methodology to construct ROMs is illustrated for a Skarstrom PSA process to separate H_2 and CH_4 . With a model reduction of 93% in size, the resulting ROM accurately mimics the actual dynamic behavior. ROM is also used to maximize hydrogen recovery within a trust-region around the point where it is constructed. ROM-based optimization is not only fast and cheap, but an accurate prediction of the descent direction together with an improvement in the objective is also obtained by ROM.

With such encouraging results, we devise a systematic adaptive trust-region based framework for optimization with ROMs. First, an exact penalty-based trust-region algorithm is developed and illustrated for a two-bed four-step PSA process for post-combustion capture. We conclude that a First-order Correction with exact gradient information is necessary for convergence to an optimum. For a CO_2 recovery maximization problem, the exact penalty algorithm with FOC converges to a local optimum after 92 TR iterations and 1.88 hrs of optimization CPU time. To circumvent the difficulty of choosing a penalty parameter, we

also devise a filter-based trust-region framework. This hybrid framework utilizes both ZOC and FOC to save on the computational effort of computing exact gradients. When applied to the PSA case study, filter TR algorithm converges to a local optimum within 51 trust-region iterations consuming 1.36 hrs of CPU time, which is significantly smaller than the penalty algorithm. Our major contributions for this part of the thesis are given below:

- **First use of POD-based ROMs for large-scale application**

Although POD-based reduced order modeling technique has been used for a variety of disciplines, its utilization has remained limited to small-scale optimal control or dynamic optimization problems. This is the first instance when the use of POD-based ROM is successfully demonstrated for a large-scale challenging application, which contains multiple sets of PDAEs, state variables, and boundary conditions. Moreover, this is the first instance when POD-based ROMs are used for adsorption systems.

- **Unique model reduction technique for PSA**

Although studies in the literature have attempted model simplification for PSA processes, this is the first successful study which reports the use of a POD-based technique to develop low-order approximations for PSA models. Also, we present a unique construction technique for ROMs for PSA which is different from other dynamic processes in the sense that we develop separate ROMs for each state variable and each operating step.

- **Trust-region algorithms for ROM-based constrained optimization**

Trust-region algorithms have been developed to handle approximate models for unconstrained optimization. In this work, we extend the use of approximate models to constrained optimization problems. In particular, our accomplishments are novel trust-region frameworks based on the exact penalty and filter approaches, and their successful demonstration on a challenging PSA process. More importantly, these frameworks are quite generic, do not make any assumption on the optimization problem, and can be extended for a wide variety of applications.

8.2 Directions for Future Work

Development of the proposed ideas in this dissertation together with the fruitful analysis of the case studies have also helped us identify many potential areas for improvement and a number of outstanding issues that need to be investigated. Some recommendations for future work in both areas are as follows.

PSA Superstructure

- The superstructure can be updated by incorporating flow valves for the inlet and exit streams of CoB and CnB. To realize operating steps, valve constants can be manipulated instead of pressures at the ends of CoB and CnB, which is more practical, and can lead to more stable and nonoscillatory solutions compared to the current optimal control framework. Moreover, such valves can ensure a proper flow control during steps like co-current pressurization and pressure equalization, which cannot be ensured neatly with bed pressures as control variables.
- Product tanks can also be incorporated in the superstructure with additional mass balance equations for them. This can help obtain operating steps that involve a pure product purge, which is not possible with the current superstructure.
- The complete discretization approach used in this work to solve the optimal control problem requires an additional accuracy verification step. Such a step can be completely avoided, and accuracy of the results can be enhanced by using a sensitivity-based sequential approach, similar to [100]. Partially discretized PDAEs together with the sensitivity equations can be integrated outside the NLP problem using a sophisticated dynamic simulator. NLP problem can then be solved using these sensitivities.
- In this work, computational limitations allow us to do the analysis with only binary feed mixtures. In future, the approach can be extended to applications that involve multi-component feed mixtures. Moreover, multiple layers of adsorbents can also be

incorporated in CoB and CnB for higher selectivity, efficient separation, and enhanced purity and recovery for one or more components.

- Although analyzed for CO₂ capture in this work, the superstructure framework is fairly general and can be applied for many other PSA applications in future.

ROM-based Optimization

- Although the trust-region algorithms developed are globally convergent, more detailed convergence analysis can be carried out with a focus on improved performance.
- In this work, we do not consider CSS conditions as a part of the original optimization problem, and achieve CSS for each trust-region iteration in our case studies because we assume that the number of decision variables, and equality and inequality constraints remain same for both original problem and ROM-based trust-region subproblem. However, CSS conditions for the original problem get reduced in dimension for the ROM-based trust-region subproblem after applying Galerkin projection onto the POD subspace, thus violating our assumption. In future, a different trust-region framework, such as the recursive multilevel algorithm proposed by Gratton et al. [87, 88] can be devised to handle CSS constraints.
- Although we demonstrate the trust-region algorithms for a two-bed four-step PSA process, in future, the proposed framework can easily be extended to optimize large-scale PSA applications involving multiple adsorbent layers, complex flow patterns and more challenging multi-component feed mixtures.
- The trust-region framework architected in this work is fairly generic and can be utilized for applications other than PSA as well.
- Finally, success of the trust-region framework depends heavily on the quality of the reduced-order models and their ability to correctly predict the descent directions for the

objective and infeasibility measure. Although, POD-based ROMs produce promising results, quality of the ROMs can be further enhanced with alternative ROM-construction techniques such as a PCA-based methodology suggested by Lang et al. [121], where ROMs are constructed using neural networks, or a Kriging approximation based methodology proposed by Caballero et al. [38], in which ROMs are essentially metamodels which replace black-box models of the unit operations in a process flowsheet.

Bibliography

- [1] *MATLAB User's Guide*, The Mathworks, Inc., 1994-2005.
- [2] *gPROMS User's Manual*, PSE Limited, London, UK, 2000.
- [3] D. Aaron and C. Tsouris, *Separation of CO₂ from Flue Gas: A Review*, *Separ. Sci. Technol.* **40** (2005), no. 1, 321–348.
- [4] N. M. Alexandrov, *Multilevel Methods for MDO*, In *Multidisciplinary Design Optimization: State of the Art*; N. M. Alexandrov and M. Y. Hussaini, Ed., Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997.
- [5] ———, *A Trust-region Framework for Managing Approximations in Constrained Optimization Problems*, Proceedings of the First ISSMO/NASA Internet Conference on Approximation and Fast Real Analysis Techniques in Engineering Optimization, June 14–27, 1998.
- [6] ———, *On Managing the Use of Surrogates in General Nonlinear Optimization and MDO*, Proceedings of 7th AIAA/USAF/NASA/ISSMO Symposium on MA&O, St.Louis, September 2–4, AIAA paper 98-4798., 1998.
- [7] N. M. Alexandrov and J. E. Dennis Jr., *Multilevel Algorithms for Nonlinear Optimization*, In *Computational Methods for Optimal Design and Control*; J. Borggaard, J. Burns, E. Cliff, and S. Schreck, Eds., Birkhäuser, 1998.
- [8] N. M. Alexandrov, J. E. Dennis Jr., R. M. Lewis, and V. Torczon, *A Trust-region Framework for Managing the Use of Approximate Models in Optimization*, *Struct. Optim.* **15** (1998), 16–23.
- [9] N. M. Alexandrov and R. M. Lewis, *An Overview of First-Order Model Management for Engineering Optimization*, *Optim. Eng.* **2** (2001), no. 4, 413–430.
- [10] N. M. Alexandrov, R. M. Lewis, C. R. Gumbert, L. L. Green, and P. A. Newman, *Approximation and Model Management in Aerodynamic Optimization with Variable-Fidelity Models*, *J. Aircraft* **38** (2001), no. 6, 1093–1101.
- [11] A. C. Antoulas and D. C. Sorensen, *Approximation of Large-Scale Dynamical Systems: An Overview*, *Int. J. Appl. Math. Comput. Sci.* **11** (2001), no. 5, 1093–1121.
- [12] E. Arian, M. Fahl, and E. Sachs, *Trust-region Proper Orthogonal Decomposition for Optimal Flow Control*, Tech. report, Institute for Computer Applications in Science and Engineering, ICASE 2000-25, NASA Langley Research Center, Hampton, 2000.

-
- [13] A. Armaou and P. D. Christofides, *Finite-dimensional Control of Nonlinear Parabolic PDE Systems with Time-dependent Spatial Domains using Empirical Eigenfunctions*, Int. J. Appl. Math. Comput. Sci. **11** (2001), no. 2, 287–317.
- [14] ———, *Dynamic Optimization of Dissipative PDE Systems using Nonlinear Order Reduction*, Chem. Eng. Sci. **7** (2002), 5083–5114.
- [15] P. Astrid, S. Weiland, K. Willcox, and T. Backx, *Missing Point Estimation in Models Described by Proper Orthogonal Decomposition*, IEEE T. Automat. Contr. **53** (2008), no. 10, 2237–2251.
- [16] A. Audet and J. E. Dennis Jr., *A Pattern-Search Filter Method for Nonlinear Programming without Derivatives*, SIAM J. Optim. **14** (2004), no. 4, 980–1010.
- [17] G. Bader and U. M. Ascher, *A New Basis Implementation for a Mixed Order Boundary Value ODE Solver*, SIAM J. Sci. and Stat. Comput. **8** (1987), no. 4, 483–500.
- [18] M. S. A. Baksh and M. W. Ackley, *Pressure Swing Adsorption Process for the Production of Hydrogen*, US Patent 6340382, 2002.
- [19] M. S. A. Baksh and C. E. Terbot, *Pressure Swing Adsorption Process for the Production of Hydrogen*, US Patent 6503299, 2003.
- [20] S. Balakrishna and L. T. Biegler, *Targeting Strategies for the Synthesis and Energy Integration of Nonisothermal Reactor Networks*, Ind. Eng. Chem. Res. **31** (1992), no. 9, 2152–2164.
- [21] E. Balsa-Canto, J. R. Banga, and A. A. Alonso, *A Novel, Efficient and Reliable Method for Thermal Process Design and Optimization. Part II: Applications*, J. Food Eng. **52** (2002), no. 3, 235–247.
- [22] E. Bendersky and P. D. Christofides, *Optimization of Transport-reaction Processes using Nonlinear Model Reduction*, Chem. Eng. Sci. **55** (2000), 4349–4366.
- [23] C. Benkmann, *System for Treatment of Plural Crude Gases in Single Adsorption Plant*, US Patent 4402712, 1983.
- [24] H. Y. Benson, D. F. Shanno, and R. J. Vanderbei, *Interior-point Methods for Nonconvex Nonlinear Programming: Filter-methods and Merit Functions*, Comput. Optim. Appl. **23** (2002), no. 2, 257–272.
- [25] M. Bergmann, L. Cordier, and J.-P. Brancher, *Control of the Cylinder Wake in the Laminar Regime by Trust-region methods and POD Reduced Order Models*, Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005, Seville, Spain, Dec 12-15, 2005.
- [26] ———, *Optimal Rotary Control of the Cylinder Wake using Proper Orthogonal Decomposition Reduced-order Model*, Phys. Fluids **17** (2005), 97–101.
- [27] G. Berkooz, P. Holmes, and J. L. Lumley, *The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows*, Annu. Rev. Fluid Mech. **25** (1993), 539–575.
- [28] L. T. Biegler, A. M. Cervantes, and A. Wächter, *Advances in Simultaneous Strategies for Dynamic Process Optimization*, Chem. Eng. Sci. **57** (2002), no. 4, 575–593.

-
- [29] L. T. Biegler, I. E. Grossmann, and A. W. Westerberg, *A Note on Approximation Techniques used for Process Optimization*, *Comput. Chem. Eng.* **9** (1985), no. 2, 201–206.
- [30] ———, *Systematic Methods for Chemical Process Design*, Prentice Hall PTR: Upper Saddle River, NJ, 1997.
- [31] L. T. Biegler, L. Jiang, and V. G. Fox, *Recent Advances in Simulation and Optimal Design of Pressure Swing Adsorption Systems*, *Sep. Purif. Rev.* **33** (2005), no. 1, 1–39.
- [32] J. P. Boris and D. L. Book, *Flux-Corrected Transport. I. SHASTA, A Fluid Transport Algorithm that Works*, *J. Comput. Phys.* **11** (1973), 38–69.
- [33] K. E. Brenan, S. L. Campbell, and L. R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, SIAM: North-Holland, NY, 1989.
- [34] T. Bui-Thanh, M. Damodaran, and K. Willcox, *Aerodynamic Data Reconstruction and Inverse Design using Proper Orthogonal Decomposition*, *AIAA J.* **42** (2004), no. 8, 1505–1516.
- [35] T. Bui-Thanh, K. Willcox, and O. Ghattas, *Model Reduction for Large-Scale Systems with High-Dimensional Parametric Input Space*, *SIAM J. Sci. Comput.* **30** (2008), no. 6, 3270–3288.
- [36] ———, *Parametric Reduced-Order Models for Probabilistic Analysis of Unsteady Aerodynamic Applications*, *AIAA J.* **46** (2008), no. 10, 2520–2529.
- [37] T. Bui-Thanh, K. Willcox, O. Ghattas, and B. Bloemen Waanders, *Goal-Oriented, Model-Constrained Optimization for Reduction of Large-Scale Systems*, *J. Comput. Phys.* **224** (2007), no. 2, 880–896.
- [38] J. A. Caballero and I. E. Grossmann, *An Algorithm for the Use of Surrogate Models in Modular Flowsheet Optimization*, *AIChE J.* **54** (2008), no. 10, 2633–2650.
- [39] Y. Cao, J. Zhu, Z. Luo, and I. M. Navon, *Reduced-order Modeling of the Upper Tropical Pacific Ocean Model using Proper Orthogonal Decomposition*, *Comput. Math. Appl.* **52** (2006), 1373–1386.
- [40] G. F. Carey and B. A. Finlayson, *Orthogonal Collocation on Finite Elements*, *Chem. Eng. Sci.* **30** (1975), 587–596.
- [41] R. G. Carter, *On the Global Convergence of Trust Region Algorithms using Inexact Gradient Information*, *SIAM J. Numer. Anal.* **28** (1991), no. 1, 251–265.
- [42] P. Cen and R. T. Yang, *Separation of a Five-Component Gas Mixture by Pressure Swing Adsorption*, *Separ. Sci. Technol.* **20** (1985), no. 9, 725–747.
- [43] P. L. Cen, W. N. Chen, and R. T. Yang, *Ternary Gas Mixture Separation by Pressure Swing Adsorption: a Combined Hydrogen-Methane Separation and Acid Gas Removal Process*, *Ind. Eng. Chem. Process Des. Dev.* **24** (1985), no. 4, 1201–1208.
- [44] A. L. Chaffee, G. P. Knowles, Z. Liang, J. Zhang, P. Xiao, and P. A. Webley, *CO₂ Capture by Adsorption: Materials and Process Development*, *Int. J. Greenh. Gas Con.* **1** (2007), no. 1, 11–18.

-
- [45] A. Chatterjee, *An Introduction to Proper Orthogonal Decomposition*, Curr. Sci. **78** (2000), no. 7, 808–817.
- [46] A. S. T. Chiang, *Arithmetic of PSA Process Scheduling*, AIChE J. **34** (1988), no. 11, 1910–1912.
- [47] K. Chihara and M. Suzuki, *Air Drying by Pressure Swing Adsorption*, J. Chem. Eng. Jpn. **16** (1983), no. 4, 293–299.
- [48] ———, *Simulation of Nonisothermal Pressure Swing Adsorption*, J. Chem. Eng. Jpn. **16** (1983), no. 1, 53–60.
- [49] M. Chlendi, D. Tondeur, and F. Rolland, *A Method to Obtain a Compact Representation of Process Performances from a Numerical Simulator: Example of Pressure Swing Adsorption for Pure Hydrogen Production*, Gas Sep. Purif. **9** (1995), no. 2, 125–135.
- [50] W.-K. Choi, T.-I. Kwon, Y.-K. Yeo, H. Lee, H. Song, and B.-K. Na, *Optimal Operation of the Pressure Swing Adsorption (PSA) Process for CO₂ Recovery*, Korean J. Chem. Eng. **20** (2003), no. 4, 617–623.
- [51] C.-T. Chou and C.-Y. Chen, *Carbon Dioxide Recovery by Vacuum Swing Adsorption*, Sep. Purif. Technol. **39** (2004), no. 1-2, 51–65.
- [52] K. T. Chue, J. N. Kim, Y. J. Yoo, S. H. Cho, and R. T. Yang, *Comparison of Activated Carbon and Zeolite 13X for CO₂ Recovery from Flue Gas by Pressure Swing Adsorption*, Ind. Eng. Chem. Res. **34** (1995), no. 2, 591–598.
- [53] Y. Chung, B.-K. Na, and H. K. Song, *Short-cut Evaluation of Pressure Swing Adsorption Systems*, Comput. Chem. Eng. **22**, **Suppl.** (1998), S637–S640.
- [54] A. R. Conn, N. M. Gould, and P. L. Toint, *Trust-Region Methods*, MPS-SIAM Series on Optimization, SIAM: Philadelphia, PA, 2000.
- [55] J. M. Coulson and J. F. Richardson, *Chemical Engineering*, vol. 3, Pergamon Press: Oxford, UK, 2nd. edition, 1979.
- [56] M. Couplet, C. Basdevant, and P. Sagaut, *Calibrated Reduced-order POD-Galerkin System for Fluid Flow Modelling*, J. Comput. Phys. **207** (2005), 192–220.
- [57] P. Cruz, J. C. Santos, F. D. Magalhães, and A. Mendes, *Cyclic Adsorption Separation Processes: Analysis Strategy and Optimization Procedure*, Chem. Eng. Sci. **58** (2003), no. 14, 3143–3158.
- [58] ———, *Simulation of Separation Processes using Finite Volume Method*, Comput. Chem. Eng. **30** (2005), 83–98.
- [59] M. S. Darwish and F. Moukalled, *TVD Schemes for Unstructured Grids*, Int. J. Heat Mass Tran. **46** (2003), no. 4, 599–611.
- [60] P. G. de Montgareuil and D. Domine, *Process for Separating a Binary Gaseous Mixture by Adsorption*, US Patent 3155468, 1964.
- [61] J. E. Dennis Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice Hall: Englewood Cliffs, NJ, 1983.

-
- [62] S. J. Doong and R. T. Yang, *Bidisperse Pore Diffusion Model for Zeolite Pressure Swing Adsorption*, *AIChE J.* **33** (1987), no. 6, 1045–1049.
- [63] M. S. Eldred and D. M. Dunlavy, *Formulations for Surrogate-based Optimization with Data Fit, Multifidelity, and Reduced-order Models*, 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Sept. 6 - 8, Portsmouth, VA. AIAA Paper 2006-7117, 2006.
- [64] M. S. Eldred, A. A. Giunta, and S. S. Collis, *Second-Order Corrections for Surrogate-Based Optimization with Model Hierarchies*, Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, NY, Aug 30-Sept 1, AIAA Paper 2004-4457, 2004.
- [65] M. Fahl, *Trust-region Methods for Flow Control based on Reduced Order Modelling*, Ph.D. thesis, Trier University, 2000.
- [66] M. Fahl and E. W. Sachs, *Reduced-order Modelling Approaches to PDE-constrained Optimization based on Proper Orthogonal Decomposition*, In *Large-scale PDE-Constrained Optimization*; L. T. Biegler, O. Ghattas, M. Heinkenschloss, and B. van Bloemen Waanders, Eds., Springer Verlag, Berlin, Germany, 2003.
- [67] S. Farooq, M. M. Hassan, and D. M. Ruthven, *Heat Effects in Pressure Swing Adsorption Systems*, *Chem. Eng. Sci.* **43** (1988), no. 5, 1017–1031.
- [68] J. Favier, L. Cordier, A. Kourta, and A. Iollo, *Calibrated POD Reduced-Order Models of Massively Separated Flows in the Perspective of their Control*, Proceedings of FEDSM2006, 2006 ASME Joint U.S. - European Fluids Engineering Summer Meeting, July 17-20, Miami, FL, 2006.
- [69] B. A. Finlayson, *Non-Linear Analysis in Chemical Engineering*, McGraw-Hill: New York, NY, 1980.
- [70] _____, *Numerical Methods for Problems with Moving Fronts*, Ravenna Park Publishing, Inc.: Seattle, WA, 1992.
- [71] R. Fletcher, *Practical Methods of Optimization: Vol 2. Constrained Optimization*, John Wiley & Sons: Somerset, NJ, 1981.
- [72] R. Fletcher, N. I. M. Gould, S. Leyffer, P. L. Toint, and A. Wächter, *Global Convergence of a Trust-region SQP-Filter Algorithm for General Nonlinear Programming*, *SIAM J. Optim.* **13** (2002), no. 3, 635–659.
- [73] R. Fletcher and S. Leyffer, *A Bundle Filter Method for Nonsmooth Nonlinear Optimization*, Tech. report, NA/195, Department of Mathematics, University of Dundee, Scotland, UK, 1999.
- [74] _____, *Nonlinear Programming without a Penalty Function*, *Math. Program.* **91** (2002), no. 2, 239–269.
- [75] R. Fletcher, S. Leyffer, and P. L. Toint, *On the Global Convergence of an SLP-filter Algorithm*, Tech. report, NA/183, Department of Mathematics, University of Dundee, Scotland, UK, 1998.

-
- [76] ———, *On the Global Convergence of a Filter-SQP Algorithm*, SIAM J. Optim. **13** (2002), no. 1, 44–59.
- [77] J. F. Forbes, T. E. Marlin, and J. F. MacGregor, *Model Adequacy Requirements for Optimizing Plant Operations*, Comput. Chem. Eng. **18** (1994), no. 6, 497–510.
- [78] R. Fourer, D. M. Gay, and B. W. Kernighan, *A Modeling Language for Mathematical Programming*, Manage. Sci. **36** (1990), no. 5, 519–554.
- [79] A. Fuderer, *Pressure Swing Adsorption Process and System*, US Patent 4381189, 1983.
- [80] ———, *Pressure Swing Adsorption with Intermediate Product Recovery*, US Patent 4512780, 1985.
- [81] A. Fuderer and E. Rudelstorfer, *Selective Adsorption Process*, US Patent 3986849, 1976.
- [82] B. Galletti, C. H. Bruneau, L. Zannetti, and A. Iollo, *Low-order Modelling of Laminar Flow Regimes past a Confined Square Cylinder*, J. Fluid Mech. **503** (2004), 161–170.
- [83] A. A. Giunta and M. S. Eldred, *Implementation of a Trust Region Model Management Strategy in the DAKOTA Optimization Toolkit*, Proceedings of the 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, CA, September 6-8, AIAA Paper 2000-4935., 2000.
- [84] E. Glueckauf and J. I. Coates, *Theory of Chromatography IV: The Influence of Incomplete Equilibrium on the Front Boundary of Chromatograms and the Effectiveness of Separation*, J. Chem. Soc. (1947), 1315–1321.
- [85] V. G. Gomes and K. W. K. Yee, *Pressure Swing Adsorption for Carbon Dioxide Sequestration from Exhaust Gases*, Sep. Purif. Technol. **28** (2002), no. 2, 161–171.
- [86] C. A. Grande, S. Cavenati, and A. E. Rodrigues, *Pressure Swing Adsorption for Carbon Dioxide Sequestration*, 2nd Mercosur Congress on Chemical Engineering and 4th Mercosur Congress on Process Systems Engineering, 2005.
- [87] S. Gratton, A. Sartenaer, and P. L. Toint, *Recursive Trust-region Methods for Multiscale Nonlinear Optimization*, Tech. report, Department of Mathematics, University of Namur, Namur, Belgium,, 2004.
- [88] ———, *Numerical Experience with a Recursive Trust-region Method for Multilevel Nonlinear Optimization*, Tech. report, Department of Mathematics, University of Namur, Namur, Belgium,, 2006.
- [89] M. D. Gunzburger, J. S. Peterson, and J. N. Shadid, *Reduced-order Modeling of Time-dependent PDEs with Multiple Parameters in the Boundary Data*, Comput. Methods Appl. Mech. Engrg. **196** (2007), 1030–1047.
- [90] M. Hirose, I. Omori, M. Oba, and T. Kawai, *Carbon Dioxide Separation and Recovery System*, Japanese Patent 2005262001, 2005.
- [91] C. Hirsch, *Numerical Computation of Internal and External Flows, Volume 1, Fundamentals of Numerical Discretization*, John Wiley-Interscience: New York, NY, 1988.
- [92] J. L. Humphrey and G. E. Keller, *in Separation Process Technology*, ch. 4, McGraw-Hill: New York, NY, 1997.

-
- [93] IEA/WEO, *World Energy Outlook 2006*, Tech. report, International Energy Agency, Paris, France, 2006.
- [94] IPCC, *Carbon Dioxide Capture and Storage*, Tech. report, Intergovernmental Panel on Climate Change, Geneva, Switzerland, 2005.
- [95] M. Ishibashi, H. Ota, N. Akutsu, S. Umeda, M. Tajika, J. Izumi, A. Yasutake, T. Kabata, and Y. Kageyama, *Technology for Removing Carbon Dioxide from Power Plant Flue Gas by the Physical Adsorption Method*, *Energ. Convers. Manage.* **37** (1996), no. 5, 929–933.
- [96] K. Ito, K. Otake, and M. Itoi, *Carbon Dioxide Desorption Method*, Japanese Patent 2004202393, 2004.
- [97] D. H. Jacobson, S. B. Gershwin, and M. M. Lele, *Computation of Optimal Singular Controls*, *IEEE T. Automat. Contr.* **15** (1970), 67–73.
- [98] J.-G. Jee, M.-B. Kim, and C.-H. Lee, *Adsorption Characteristics of Hydrogen Mixtures in a Layered Bed: Binary, Ternary, and Five-Component Mixtures*, *Ind. Eng. Chem. Res.* **40** (2001), no. 3, 868–878.
- [99] L. Jiang, *Optimization of Partial Differential Equation-based Systems - Application to Pressure Swing Adsorption Processes*, Ph.D. thesis, Carnegie Mellon University, Pittsburgh, PA, 2004.
- [100] L. Jiang, V. G. Fox, and L. T. Biegler, *Simulation and Optimal Design of Multiple-Bed Pressure Swing Adsorption Systems*, *AIChE J.* **50** (2004), no. 11, 2904–2917.
- [101] S. Kameswaran, *Analysis and Formulation of a Class of Complex Dynamic Optimization Problems*, Ph.D. thesis, Carnegie Mellon University, Pittsburgh, PA, 2006.
- [102] S. Kameswaran and L. T. Biegler, *Simultaneous Dynamic Optimization Strategies: Recent Advances and Challenges*, *Comput. Chem. Eng.* **30** (2006), no. 10-12, 1560–1575.
- [103] ———, *Convergence Rates for Direct Transcription of Optimal Control Problems using Collocation at Radau points*, *Comput. Optim. Appl.* **41** (2008), no. 1, 81–126.
- [104] A. Kapoor and R. T. Yang, *Optimization of a Pressure Swing Adsorption Cycle*, *Ind. Eng. Chem. Res.* **27** (1988), no. 1, 204–206.
- [105] J. Karger and D. M. Ruthven, *Diffusion in Zeolites and Other Porous Solids*, John Wiley-Interscience: New York, NY, 1992.
- [106] G. E. Keller, *Gas Adsorption Processes: State of the Art*, In *Industrial Gas Separations*; T. E. Whyte, Ed., American Chemical Society: Washington, DC, ACS Symposium Series 223, Vol. 145, 1983.
- [107] E. S. Kikkinides, R. T. Yang, and S. H. Cho, *Concentration and Recovery of Carbon Dioxide from Flue Gas by Pressure Swing Adsorption*, *Ind. Eng. Chem. Res.* **32** (1993), no. 11, 2714–2720.
- [108] S. P. Knaebel, D. Ko, and L. T. Biegler, *Simulation and Optimization of a Pressure Swing Adsorption System: Recovering Hydrogen from Methane*, *Adsorption* **11** (2005), 615.

-
- [109] D. Ko and I. L. Moon, *Multiobjective Optimization of Cyclic Adsorption Processes*, Ind. Eng. Chem. Res. **41** (2002), no. 1, 93–104.
- [110] D. Ko, R. Siriwardane, and L. T. Biegler, *Optimization of Pressure Swing Adsorption Process using Zeolite 13X for CO₂ Sequestration*, Ind. Eng. Chem. Res. **42** (2003), no. 2, 339–348.
- [111] ———, *Optimization of Pressure Swing Adsorption and Fractionated Vacuum Pressure Swing Adsorption Processes for CO₂ Capture*, Ind. Eng. Chem. Res. **44** (2005), no. 21, 8084–8094.
- [112] B. Kragel, *Streamline Diffusion POD Models in Optimization*, Ph.D. thesis, Trier University, 2005.
- [113] R. Kumar, *Removal of Water and Carbon Dioxide from Atmospheric Air*, US Patent 4711645, 1987.
- [114] R. Kumar, V. G. Fox, D. Hartzog, R. E. Larson, Chen Y. C., P.A. Houghton, and T. Naheiri, *A Versatile Process Simulator for Adsorptive Separations*, Chem. Eng. Sci. **49** (1994), 3115–3125.
- [115] K. Kunisch and S. Volkwein, *Control of the Burgers Equation by a Reduced-Order Approach Using Proper Orthogonal Decomposition*, J. Opt. Theory Appl. **102** (1999), no. 2, 345–371.
- [116] ———, *Galerkin Proper Orthogonal Decomposition Methods for Parabolic Problems*, Numer. Math. **90** (2001), 117–148.
- [117] ———, *Galerkin Proper Orthogonal Decomposition Methods for a General Equation in Fluid Dynamics*, SIAM J. Numer. Anal. **40** (2002), no. 2, 492–515.
- [118] H. M. Kvamsdal and T. Hertzberg, *Optimization of Pressure Swing Adsorption Systems—The Effect of Mass Transfer during the Blowdown Step*, Chem. Eng. Sci. **50** (1995), no. 7, 1203–1212.
- [119] ———, *Pressure Swing Adsorption - Optimization of a Trace Separation System*, Comput. Chem. Eng. **19** (1995), no. 11, 339–344.
- [120] ———, *Optimization of PSA Systems—Studies on Cyclic Steady State Convergence*, Comput. Chem. Eng. **21** (1997), no. 8, 819–832.
- [121] Y.-d. Lang, A. Malacina, L. T. Biegler, S. Munteanu, J. I. Madsen, and S. E. Zitney, *Reduced Order Model based on Principal Component Analysis for Process Simulation and Optimization*, Energ. Fuel. **23** (2009), no. 3, 1695–1706.
- [122] P. A. LeGresley and J. J. Alonso, *Airfoil Design Optimization using Reduced Order Models based on Proper Orthogonal Decomposition*, Fluids 2000 Conference and Exhibit, June 19–22, Denver, CO. AIAA Paper 2000-2545, 2000.
- [123] R. J. LeVeque, *CLAWPACK Software*, <http://www.amath.washington.edu/~claw>.
- [124] ———, *Numerical Methods for Conservation Laws*, Birkhäuser, 1992.
- [125] ———, *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press: Cambridge, UK, 2002.

-
- [126] J. L. Lumley, *Coherent Structures in Turbulence*, pp. 215–242, in: R.E. Meyer (ed.), *Transition and Turbulence*, Academic Press: New York, NY, 1981.
- [127] E. Luna-Ortiz and C. Theodoropoulos, *An Input-Output Model Reduction-Based Optimization Scheme for Large-Scale Systems*, *Multiscale Model. Simul.* **4** (2005), no. 2, 691–708.
- [128] H. V. Ly and H. T. Tran, *Modeling and Control of Physical Processes Using Proper Orthogonal Decomposition*, *Math. Comput. Model.* **33** (2001), no. 1-3, 223–236.
- [129] A. Malek and S. Farooq, *Study of a Six-Bed Pressure Swing Adsorption Process*, *AIChE J.* **43** (1997), no. 10, 2509–2523.
- [130] G. Q. Miller and J. Stöcker, *Selection of a Hydrogen Separation Process*, NPRA Annual Meeting: San Francisco, California, 1989.
- [131] J. Moehlis, T. R. Smith, P. Holmes, and H. Faisst, *Models for Turbulent Plane Couette Flow using the Proper Orthogonal Decomposition*, *Phys. Fluids* **14** (2002), no. 7, 2493–2507.
- [132] B.-K. Na, K.-K. Koo, H.-M. Eum, H. Lee, and H. Song, *CO₂ Recovery from Flue Gas by PSA Process using Activated Carbon*, *Korean J. Chem. Eng.* **18** (2001), no. 2, 220–227.
- [133] B.-K. Na, H. Lee, K.-K. Koo, and H. K. Song, *Effect of Rinse and Recycle Methods on the Pressure Swing Adsorption Process To Recover CO₂ from Power Plant Flue Gas Using Activated Carbon*, *Ind. Eng. Chem. Res.* **41** (2002), no. 22, 5498–5503.
- [134] NETL/DOE, *The Cost and Performance Baseline for Fossil Energy Power Plants study, Volume 1: Bituminous Coal and Natural Gas to Electricity*, Tech. report, National Energy Technology Laboratory, Department of Energy, USA, May, 2007.
- [135] P.-y. Nie and C.-f. Ma, *A Trust-region Filter Method for General Non-linear Programming*, *Appl. Math. Comput.* **172** (2006), 1000–1017.
- [136] D. Nikolić, A. Giovanoglou, M. C. Georgiadis, and E. S. Kikkinides, *Generic Modeling Framework for Gas Separations Using Multibed Pressure Swing Adsorption Processes*, *Ind. Eng. Chem. Res.* **47** (2008), no. 9, 3156–3169.
- [137] D. Nikolić, E. S. Kikkinides, and M. C. Georgiadis, *Optimization of Multibed Pressure Swing Adsorption Processes*, *Ind. Eng. Chem. Res.* **48** (2009), no. 11, 5388–5398.
- [138] S. Nilchan, *The Optimisation of Periodic Adsorption Processes*, Ph.D. thesis, Imperial College of Science, Technology, and Medicine, London, UK, 1997.
- [139] S. Nilchan and C. C. Pantelides, *On the Optimisation of Periodic Adsorption Processes*, *Adsorption* **4** (1998), no. 2, 113–147.
- [140] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer-Verlag: New York, NY, 1999.
- [141] H. M. Park and D. H. Cho, *The Use of the Karhunen-Loève Decomposition for the Modeling of Distributed Parameter Systems*, *Chem. Eng. Sci.* **51** (1996), no. 1, 81–98.

-
- [142] J.-H. Park, H.-T. Beum, J.-N. Kim, and S.-H. Cho, *Numerical Analysis on the Power Consumption of the PSA Process for Recovering CO₂ from Flue Gas*, *Ind. Eng. Chem. Res.* **41** (2002), no. 16, 4122–4131.
- [143] Fletcher. R., S. Leyffer, and P. Toint, *A Brief History of Filter Methods*, Tech. report, ANL/MCS-P1372-0906, Argonne National Laboratory, Mathematics and Computer Science Division., 2006.
- [144] N. S. Raghavan, M. M. Hassan, and D. M. Ruthven, *Numerical Simulation of a PSA System - Part I*, *AIChE J.* **31** (1985), no. 3, 385–392.
- [145] R. Rajasree and A. S. Moharir, *Simulation based Synthesis, Design and Optimization of Pressure Swing Adsorption (PSA) Processes*, *Comput. Chem. Eng.* **24** (2000), no. 11, 2493–2505.
- [146] J. Rambo and Y. Joshi, *Reduced-order Modeling of Turbulent Forced Convection with Parametric Conditions*, *Int. J. Heat Mass Tran.* **50** (2007), 539–551.
- [147] S. S. Ravindran, *Proper Orthogonal Decomposition in Optimal Control of Fluids*, Tech. report, NASA/TM-1999-209113, 1999.
- [148] S. Reynolds, A. Mehrotra, A. Ebner, and J. Ritter, *Heavy Reflux PSA Cycles for CO₂ Recovery from Flue Gas: Part I. Performance Evaluation*, *Adsorption* **14** (2008), no. 2, 399–413.
- [149] S. P. Reynolds, A. D. Ebner, and J. A. Ritter, *New Pressure Swing Adsorption Cycles for Carbon Dioxide Sequestration*, *Adsorption* **11** (2005), no. 0, 531–536.
- [150] ———, *Carbon Dioxide Capture from Flue Gas by Pressure Swing Adsorption at High Temperature using a K-promoted HTlc: Effects of Mass Transfer on the Process Performance*, *Environ. Prog.* **25** (2006), no. 4, 334–342.
- [151] ———, *Stripping PSA Cycles for CO₂ Recovery from Flue Gas at High Temperature Using a Hydrotalcite-Like Adsorbent*, *Ind. Eng. Chem. Res.* **45** (2006), no. 12, 4278–4294.
- [152] J. A. Ritter and A. D. Ebner, *State-of-the-Art Adsorption and Membrane Separation Processes for Hydrogen Production in the Chemical and Petrochemical Industries*, *Separ. Sci. Technol.* **42** (2007), no. 6, 1123–1193.
- [153] C. W. Rowley, T. Colonius, and R. M. Murray, *Model Reduction for Compressible Flows using POD and Galerkin Projection*, *Physica D* **189** (2004), no. 1–2, 115–129.
- [154] D. M. Ruthven, *Principles of Adsorption and Adsorption Processes*, John Wiley-Interscience: New York, NY, 1984.
- [155] ———, *Past Progress and Future Challenges in Adsorption Research: Commentaries*, *Ind. Eng. Chem. Res.* **39** (2000), no. 7, 2127–2131.
- [156] D. M. Ruthven, S. Farooq, and K. S. Knaebel, *Pressure Swing Adsorption*, VCH Publishers: New York, NY, 1994.
- [157] B. Sankararao and S. K. Gupta, *Multi-Objective Optimization of Pressure Swing Adsorbers for Air Separation*, *Ind. Eng. Chem. Res.* **46** (2007), no. 11, 3751–3765.

-
- [158] A. Sasaki, S. Matsumoto, M. Fujitsuka, T. Shinoki, T. Tanaka, and J. Ohtsuki, *CO₂ Recovery in Molten Carbonate Fuel Cell System by Pressure Swing Adsorption*, IEEE T. Energy Conver. **8** (1993), no. 1, 26–32.
- [159] J. Schell, N. Casas, and M. Mazzotti, *Pre-combustion CO₂ Capture for IGCC Plants by an Adsorption Process*, Energ. Procedia **1** (2009), 655–660.
- [160] W. E. Schiesser, *The Numerical Method of Lines Integration of Partial Differential Equations*, Academic Press: San Diego, CA, 1991.
- [161] M. Schlegel, K. Stockmann, T. Binder, and W. Marquardt, *Dynamic Optimization using Adaptive Control Vector Parameterization*, Comput. Chem. Eng. **29** (2005), no. 8, 1731–1751.
- [162] S. Y. Shvartsman, C. Theodoropoulos, R. Rico-Martinez, I. G. Kevrekidis, E. S. Titi, and T. J. Mountziaris, *Order Reduction for Nonlinear Dynamic Models of Distributed Reacting Systems*, J. Process Contr. **10** (2000), 177–184.
- [163] R. E. H. Sims, H.-H. Rogner, and K. Gregory, *Carbon emission and mitigation cost comparisons between fossil fuel, nuclear and renewable energy resources for electricity generation*, Energ. Policy **31** (2003), no. 13, 1315–1326.
- [164] S. Sircar, *Separation of Multicomponent Gas Mixtures*, US Patent 4171206, 1979.
- [165] ———, *Separation of Methane and Carbon Dioxide Gas Mixtures by Pressure Swing Adsorption*, Separ. Sci. Technol. **23** (1988), no. 6, 519–529.
- [166] ———, *Pressure Swing Adsorption: Research Needs in Industry*, Proceedings of Third International Conference on Fundamentals of Adsorption: Sonthofen, Germany; Mersmann, A. B., Scholl, S. E., Eds.; AIChE: New York, NY; pp 815-843., 1989.
- [167] ———, *Publications on Adsorption Science and Technology*, Adsorption **6** (2000), 359–365.
- [168] ———, *Applications of Gas separation by Adsorption for the Future*, Adsorpt. Sci. Technol. **19** (2001), 347–366.
- [169] ———, *Pressure Swing Adsorption: Commentaries*, Ind. Eng. Chem. Res. **41** (2002), no. 6, 1389–1392.
- [170] ———, *Basic Research Needs for Design of Adsorptive Gas Separation Processes*, Ind. Eng. Chem. Res. **45** (2006), no. 16, 5435–5448.
- [171] S. Sircar and T. C. Golden, *Purification of Hydrogen by Pressure Swing Adsorption*, Separ. Sci. Technol. **35** (2000), no. 5, 667–687.
- [172] S. Sircar and J.R. Hufton, *Why Does the Linear Driving Force Model for Adsorption Kinetics Work?*, Adsorption **6** (2000), no. 2, 137–147.
- [173] S. Sircar and W. C. Kratz, *Simultaneous Production of Hydrogen and Carbon Dioxide from Steam Reformer Off-Gas by Pressure Swing Adsorption*, Separ. Sci. Technol. **23** (1988), no. 14, 2397–2415.
- [174] S. Sircar and J. W. Zondlo, *Hydrogen Purification by Selective Adsorption*, US Patent 4077779, 1978.

-
- [175] L. Sirovich, *Turbulence and the Dynamics of Coherent Structures*, Q. Appl. Math. **45** (1987), no. 3, 561–571.
- [176] C. W. Skarstrom, *Method and Apparatus for Fractionating Gaseous Mixtures by Adsorption*, US Patent 2944627, 1960.
- [177] O. J. Smith and A. W. Westerberg, *Mixed-integer Programming for Pressure Swing Adsorption Cycle Scheduling*, Chem. Eng. Sci. **45** (1990), no. 9, 2833–2842.
- [178] ———, *The Optimal Design of Pressure Swing Adsorption Systems*, Chem. Eng. Sci. **46** (1991), no. 12, 2967–2976.
- [179] ———, *The Optimal Design of Pressure Swing Adsorption Systems–II*, Chem. Eng. Sci. **47** (1992), no. 15-16, 4213–4217.
- [180] B. Srinivasan, S. Palanki, and D. Bonvin, *Dynamic Optimization of Batch Processes: I. Characterization of the Nominal Solution*, Comput. Chem. Eng. **15** (2003), no. 1, 1–26.
- [181] M. Suzuki, *Adsorption Engineering*, Kondasha Ltd.: Tokyo, and Elsevier Science Publishers: Amsterdam, 1990.
- [182] Y. Takamura, S. Narita, J. Aoki, S. Hironaka, and S. Uchida, *Evaluation of Dual-bed Pressure Swing Adsorption for CO₂ Recovery from Boiler Exhaust Gas*, Sep. Purif. Technol. **24** (2001), no. 3, 519–528.
- [183] T. Tamura, *Absorption Process for Gas Separation*, US Patent 3797201, 1974.
- [184] M. Tańczyk and K. Warmuziński, *Multicomponent Pressure Swing Adsorption. Part II. Experimental Verification of the Model*, Chem. Eng. Process. **37** (1998), no. 4, 301–315.
- [185] A. Theodoropoulou, R. A. Adomaitis, and E. Zafriou, *Model Reduction for Optimization of Rapid Thermal Chemical Vapor Deposition Systems*, IEEE Trans. Semicond. Manuf. **11** (1998), no. 1, 85–98.
- [186] P. L. Toint, *Global Convergence of a Class of Trust-region Methods for Nonconvex Minimization in Hilbert Space*, IMA J. Numer. Anal. **8** (1988), 231–252.
- [187] M. Ulbrich, S. Ulbrich, and L. N. Vicente, *A Globally Convergent Primal-dual Interior-Point Filter Method for Nonconvex Nonlinear Programming*, Math. Program. **100** (2003), no. 2, 379–410.
- [188] D. P. Valenzuela and A. L. Myers, *Adsorption Equilibrium Data Handbook*, Prentice Hall: Englewood Cliffs, NJ, 1989.
- [189] A. Varshney and A. Armaou, *Reduced Order Modeling and Dynamic Optimization of Multiscale PDE/kMC Process Systems*, Comput. Chem. Eng. **32** (2008), no. 9, 2136–2143.
- [190] T. Vermuelen, G. Klein, and N. K. Hiester, in *Chemical Engineers' Handbook*, ch. 16, R. H. Perry and C. H. Chilton, Ed., 5th ed, McGraw-Hill: New York, NY, 1974.
- [191] C. Voss, *Applications of Pressure Swing Adsorption Technology*, Adsorption **11** (2005), no. 0, 527–529.

-
- [192] A. Wächter and L. T. Biegler, *Failure of Global Convergence for a Class of Interior-Point Methods for Nonlinear Programming*, Math. Program. **88** (2000), no. 3, 565–574.
- [193] ———, *Line Search Filter Methods for Nonlinear Programming: Local Convergence*, SIAM J. Optim. **16** (2005), no. 1, 32–48.
- [194] ———, *Line Search Filter Methods for Nonlinear Programming: Motivation and Global Convergence*, SIAM J. Optim. **16** (2005), no. 1, 1–31.
- [195] ———, *On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-scale Nonlinear Programming*, Math. Program. **106** (2006), no. 1, 25–57.
- [196] K. Warmuźiński and M. Tańczyk, *Multicomponent Pressure Swing Adsorption Part I. Modeling of Large-scale PSA Installations*, Chem. Eng. Process. **36** (1997), 89–99.
- [197] P. A. Webley and J. He, *Fast Solution-adaptive Finite Volume Method for PSA/VSA Cycle Simulation; 1 Single Step Simulation*, Comput. Chem. Eng. **23** (2000), no. 11-12, 1701 – 1712.
- [198] G. Weickum, M. S. Eldred, and K. Maute, *Multi-point Extended Reduced Order Modeling For Design Optimization and Uncertainty Analysis*, 2nd AIAA Multidisciplinary Design Optimization Specialist Conference, May 1 - 4, Newport, RI. AIAA Paper 2006-2145., 2006.
- [199] M. Whysall and L. J. M. Wagemans, *Very Large-scale Pressure Swing Adsorption Processes*, US Patent 6210466, 2001.
- [200] K. Willcox, *Reduced-Order Aerodynamic Models for Aeroelastic Control of Turbomachines*, Ph.D. thesis, Massachusetts Institute of Technology, Boston, MA, 2000.
- [201] K. Willcox and Peraire J., *Balanced Model Reduction via the Proper Orthogonal Decomposition*, AIAA J. **40** (2002), no. 11, 2323–2330.
- [202] P. Xiao, S. Wilson, G. Xiao, R. Singh, and P. Webley, *Novel Adsorption Processes for Carbon Dioxide Capture within an IGCC Process*, Energy Procedia **1** (2009), 631–638.
- [203] P. Xiao, J. Zhang, P. A. Webley, G. Li, R. Singh, and R. Todd, *Capture of CO₂ from Flue gas Streams with Zeolite 13X by a Vacuum-Pressure Swing Adsorption*, Adsorption **14** (2008), no. 4, 575–582.
- [204] J. Xu and E. L. Weist, *Six Bed Pressure Swing Adsorption Process with Four Steps of Pressure Equalization*, US Patent 6454838, 2002.
- [205] T. Yamaguchi and K. Yasushi, *Gas Separation Process*, US Patent 5250088, 1993.
- [206] R. T. Yang, *Gas Separation by Adsorption Processes*, Butterworths: Boston, MA, 1997.
- [207] R. T. Yang and S. J. Doong, *Gas Separation by Pressure Swing Adsorption: a Pore-Diffusion Model for Bulk Separation*, AIChE J. **31** (1985), 1829–1841.
- [208] S.-Il Yang, D.-Y. Choi, S.-C. Jang, S.-H. Kim, and D.-K. Choi, *Hydrogen Separation by Multi-bed Pressure Swing Adsorption of Synthesis Gas*, Adsorption **14** (2008), no. 4, 583–590.

- [209] T. Yokoyama, *Japanese R&D on Large-Scale CO₂ Capture*, ECI Symposium Series on Separations Technology VI: New Perspectives on Very Large-Scale Operations, vol. RP3, 2004.
- [210] T. Yuan, P. G. Cizmas, and T. Ó Brien, *A Reduced-order Model for a Bubbling Fluidized Bed based on Proper Orthogonal Decomposition*, *Comput. Chem. Eng.* **30** (2005), 243–259.
- [211] J. Zhang and P. A. Webley, *Cycle Development and Design for CO₂ Capture from Flue Gas by Vacuum Swing Adsorption*, *Environ. Sci. Technol.* **42** (2008), no. 2, 563–569.
- [212] J. Zhang, P. A. Webley, and P. Xiao, *Effect of Process Parameters on Power Requirements of Vacuum Swing Adsorption Technology for CO₂ Capture from Flue Gas*, *Energ. Convers. Manage.* **49** (2008), no. 2, 346–356.
- [213] L. Zhou, C.-Z. L, S.-J. Bian, and Y.-P. Zhou, *Pure Hydrogen from the Dry Gas of Refineries via a Novel Pressure Swing Adsorption Process*, *Ind. Eng. Chem. Res.* **41** (2002), no. 21, 5290–5297.
- [214] S. E. Zitney and M. Syamlal, *Integrated Process Simulation and CFD for Improved Process Engineering*, Proceedings of the European Symposium on Computer Aided Process Engineering–12, ESCAPE–12, May 26–29: The Hague, The Netherlands; Grievink, J., and van Schijndel, J., Eds.; pp 397–402., 2002.

Appendix A

Nomenclature

BR_i	flux of i^{th} component in the bottom reflux stream ($\text{gmol m}^{-2} \text{sec}^{-1}$)
C_i	gas-phase concentration of i^{th} component (gmol m^{-3})
C_{pg}^i	heat capacity of i^{th} component ($\text{J gmol}^{-1} \text{K}^{-1}$)
C_{ps}	heat capacity of the adsorbent ($\text{J kg}^{-1} \text{K}^{-1}$)
D	Adsorbent bed diameter (m)
D_K	Knudsen diffusivity of i^{th} component (m^2/s)
D_L	Axial dispersion (m^2/s)
$D_{m,i}$	Bulk diffusivity of i^{th} component (m^2/s)
$D_{p,i}$	Macropore diffusivity of i^{th} component (m^2/s)
d_p	particle diameter (m)
F_i	input flux of i^{th} component to the co-current bed ($\text{gmol m}^{-2} \text{sec}^{-1}$)
h	total gas-phase enthalpy (J m^{-3})
h_w	fluid-to-wall heat transfer coefficient ($\text{J m}^{-2} \text{sec}^{-1} \text{K}^{-1}$)
ΔH_i^{ads}	isosteric heat of adsorption (J gmol^{-1})
HP_i	flux of i^{th} component in the heavy product stream ($\text{gmol m}^{-2} \text{sec}^{-1}$)
K_L	effective axial thermal conductivity ($\text{J m}^{-1} \text{sec}^{-1} \text{K}^{-1}$)
k_i	lumped mass transfer coefficient for i^{th} component (sec^{-1})
k_i^H	Henry's constant ($\text{gmol kg}^{-1} \text{kPa}^{-1}$)
L	bed length (m)
LP_i	flux of i^{th} component in the light product stream ($\text{gmol m}^{-2} \text{sec}^{-1}$)

M_w^i	molecular weight of i^{th} component (kg gmol ⁻¹)
M_w	average molecular weight of the gas mixture (kg gmol ⁻¹)
N_c	number of components in the mixture
P	total bed pressure (kPa)
P_i	partial pressure of i^{th} component (kPa)
P_{ads}	pressure at the light end of the co-current bed (kPa)
P_{des}	pressure at the heavy end of the counter-current bed (kPa)
P_H	adsorption pressure for Skarstrom cycle (same as P_h , P_{high}) (kPa)
P_L	desorption pressure for Skarstrom cycle (same as P_l , P_{low}) (kPa)
$Q_{feed,i}$	flux of i^{th} component in the feed stream (gmol m ⁻² sec ⁻¹)
q_i	solid-phase concentration of i^{th} component (gmol kg ⁻¹)
q_i^*	equilibrium solid-phase concentration of i^{th} component (gmol kg ⁻¹)
R	universal gas constant (J gmol ⁻¹ K ⁻¹)
R_p	adsorbent particle radius (m)
T	gas-phase temperature in the bed (K)
T_w	wall/ambient temperature (K)
TR_i	flux of i^{th} component in the top reflux stream (gmol m ⁻² sec ⁻¹)
t_a	adsorption step time (same as desorption time) (sec)
t_p	pressurization step time (same as depressurization time) (sec)
U_A	effective heat transfer coefficient (J m ⁻³ sec ⁻¹ K ⁻¹)
u_0	bed inlet superficial velocity at $x = 0$ (m sec ⁻¹)
u_L	bed exit superficial velocity at $x = L$ (m sec ⁻¹)
u_a	feed velocity for the adsorption step (m sec ⁻¹)
u_{feed}	feed velocity (m sec ⁻¹)
u_{reg}	exhaust velocity (m sec ⁻¹)
v	gas superficial velocity (m sec ⁻¹)
W_{total}	total work done by the compressors and the vacuum generator (J m ⁻² sec ⁻¹)
y_i	mole fraction of i^{th} component
$y_{f,i}$	feed mole fraction of i^{th} component

Greek Letters

α	fraction of the heavy product going as a bottom reflux
β	fraction of the light product going as a top reflux
ϵ_b	bulk void fraction
ϵ_p	adsorbent particle void fraction
γ	heat capacity ratio (=1.4)
μ	gas viscosity ($\text{kg m}^{-1} \text{sec}^{-1}$)
η_c	efficiency of the feed compressor
η_{fg}	efficiency of the flue gas compressor
η_h	efficiency of the heavy product compressor
η_v	efficiency of the vacuum generator
ϕ	fraction of the feed going to the co-current bed
ρ_b	bulk density (kg m^{-3})
ρ_g	average gas density (kg m^{-3})
ρ_s	adsorbent density (kg m^{-3})
τ	Tortuosity factor

Subscripts

<i>des</i>	desorption step
<i>pres</i>	pressurization step

Appendix B

Optimization Iterations

B.1 Exact Penalty Algorithm with ZOC

Table B.1: Iteration sequence for Algorithm I with ZOC for Problem (7.31)

k	Δ_k	P_h (kPa)	P_l (kPa)	t_p (sec)	t_a (sec)	u_a (m/s)	p_{co2} (%)	r_{co2} (%)	$\tilde{\psi}_k^{\mathcal{R}}(0)$	$\tilde{\psi}_k^{\mathcal{R}}(s_k)$	$\psi(s_k)$	ρ_k	step?
		150	50	50	150	0.2	37.76	66.27			12173.7		
0	8	158	48	54	154	0.22	40.29	67.55	12170	9990	9642.45	1.16	yes
1	12	148.67	45.02	57.05	159.76	0.198	40.34	69.93	9641	5311.5	9590.07	0.01	no
2	3	161	47.25	55.5	155.5	0.228	41.25	68.04	9641	8923.7	8681.96	1.34	yes
3	4.5	165.5	46.13	57.75	157.75	0.239	42.68	68.7	8684	7517.8	7251.3	1.23	yes
4	6.75	172.25	44.44	61.13	161.13	0.256	44.83	69.73	7252	5248.7	5100.27	1.07	yes
5	10.13	182.38	41.91	66.19	166.19	0.281	48.05	71.25	5100	1598.6	1878.76	0.92	yes
6	10.13	192.5	40	61.13	170.57	0.256	50.3	75.48	1878	486.41	-75.3967	1.4	yes
7	15.19	207.08	40	53.53	162.98	0.218	50.79	80.98	-75.4	-84	-80.9484	0.65	yes
8	15.19	210.02	40	61.13	155.39	0.18	48.76	86.15	-80.95	-93.12	1153.87	-101	no
9	3.797	203.28	40	55.43	161.08	0.208	50.01	81.74	-80.95	-84.29	-79.6689	0.23	yes
10	0.949	202.33	40	55.9	160.61	0.206	49.81	81.94	-78.99	-79.86	108.191	-217	no
11	0.237	203.04	40	55.55	160.96	0.208	49.96	81.79	-78.99	-79.21	-41.1745	-178	no
12	0.059	203.22	40	55.5	161.05	0.208	49.99	81.75	-78.99	-79.05	-69.6789	-185	no

B.2 Exact Penalty Algorithm with FOC

Table B.2: Iteration sequence for Algorithm I with FOC for Problem (7.31)

k	Δ_k	P_h	P_l	t_p	t_a	u_a	p_{co2}	r_{co2}	$\tilde{\psi}_k^{\mathcal{R}}(0)$	$\tilde{\psi}_k^{\mathcal{R}}(s_k)$	$\psi(s_k)$	ρ_k	step?
		(kPa)	(kPa)	(sec)	(sec)	(m/s)	(%)	(%)					
		150	50	50	150	0.2	37.76	66.27			12173.7		
0	8	158	48	46	154	0.22	40.39	67.69	12170	8823.4	9542.31	0.79	yes
1	8	166	46	42	158	0.24	43.05	69.09	9544	6745.1	6880.91	0.95	yes
2	8	174	44	38	162	0.26	45.695	70.41	6885	3369.2	4234.59	0.75	yes
3	8	182	42	35	166	0.28	48.37	71.78	4234	1726.9	1558.22	1.07	yes
4	12	194	40.1	35	172	0.25	50.77	76.48	1560	-2058	-76.48	0.45	yes
5	6	200	40	35	169	0.235	51.101	78.58	-76.49	-80.35	-78.58	0.54	yes
6	6	206	40	35	166	0.22	51.21	80.86	-78.58	-80.54	-80.86	1.16	yes
7	9	215	40	35	161.5	0.1975	51.06	84.47	-80.85	-88.76	-84.465	0.46	yes
8	4.5	219.5	40	35	159.25	0.1863	50.79	86.3	-84.45	-86.13	-86.3	1.1	yes
9	6.75	226.25	40	35	155.88	0.1694	49.03	91.51	-86.27	-89.53	878.49	-295	no
10	1.688	221.19	40	35	158.41	0.182	50.64	86.99	-86.29	-87.04	-86.99	0.93	yes
11	1.688	222.88	40	35	157.56	0.1778	50.36	87.71	-86.98	-87.72	-87.71	0.97	yes
12	1.688	224.56	40	35	156.72	0.1741	49.76	89.72	-87.71	-88.64	150.28	-255	no
13	0.422	223.3	40	35	157.35	0.1768	50.19	88.38	-87.71	-87.95	-88.38	2.75	yes
14	0.64	223.94	40	35	157.03	0.1758	50.02	88.95	-88.38	-89.19	-88.95	0.7	yes
15	0.64	224.58	40	35	157.35	0.1749	50	89.18	-88.95	-89.17	-89.059	0.5	yes
16	0.32	224.9	40	35	157.51	0.1746	50	89.27	-89.18	-89.29	-89.149	0.83	yes
17	0.32	225.22	40	35	157.35	0.1747	50.002	89.31	-89.27	-89.37	-89.309	1.75	yes
18	0.48	225.7	40	35	157.59	0.1742	50.002	89.4	-89.31	-89.41	-89.401	1.01	yes
19	0.72	226.42	40	35	157.23	0.1742	50.007	89.57	-89.4	-89.56	-89.57	1.08	yes
20	1.08	227.5	40	35	157.77	0.1732	50	89.76	-89.57	-89.81	-89.637	0.27	yes
21	0.54	228.04	40	35	157.5	0.1731	50	89.98	-89.76	-90.02	-89.859	0.84	yes
22	0.54	228.58	40	35	157.23	0.1733	50.009	90.01	-89.98	-90.05	-90.007	1.99	yes
23	0.81	229.39	40	35	156.83	0.1734	50.012	90.16	-90.01	-90.19	-90.155	0.79	yes
24	0.81	230.2	40	35	157.23	0.1725	50	90.35	-90.16	-90.41	-90.229	0.3	yes
25	0.405	230.61	40	35	157.03	0.1729	50.05	90.32	-90.35	-90.46	-90.32	0.81	yes
26	0.405	231.01	40	35	157.23	0.1721	50	90.51	-90.32	-90.58	-90.455	0.52	yes
27	0.405	231.42	40	35	157.43	0.1718	50.001	90.48	-90.51	-90.6	-90.479	0.26	yes

continued on next page

continued from previous page

k	Δ_k	P_h	P_l	t_p	t_a	u_a	p_{co2}	r_{co2}	$\tilde{\psi}_k^R(0)$	$\tilde{\psi}_k^R(s_k)$	$\psi(s_k)$	ρ_k	step?
		(kPa)	(kPa)	(sec)	(sec)	(m/s)	(%)	(%)					
28	0.2	231.62	40	35	157.53	0.1716	50	90.54	-90.48	-90.5	-90.489	0.55	yes
29	0.2	231.82	40	35	157.63	0.1714	50.001	90.55	-90.54	-90.57	-90.551	2.48	yes
30	0.3	232.12	40	35	157.78	0.1711	50	90.72	-90.55	-90.6	-90.599	1.06	yes
31	0.45	232.57	40	35	158.01	0.1707	50.004	90.78	-90.72	-90.78	-90.779	3.05	yes
32	0.8	233.37	40	35	158.41	0.1697	50	90.92	-90.78	-91.11	-90.864	0.26	yes
33	0.4	233.77	40	35	158.61	0.1694	50	90.96	-90.92	-91	-90.909	0.51	yes
34	0.4	234.17	40	35	158.81	0.1692	50.01	90.99	-90.96	-91.01	-90.986	1.48	yes
35	0.6	234.77	40	35	159.11	0.1687	50.009	91.08	-90.99	-91.14	-91.083	0.63	yes
36	0.6	235.37	40	35	159.41	0.1682	50.02	91.24	-91.08	-91.23	-91.24	1.07	yes
37	0.9	236.27	40	35	158.96	0.1683	50.012	91.34	-91.24	-91.43	-91.336	0.5	yes
38	0.9	237.17	40	35	158.51	0.1684	50.01	91.56	-91.34	-91.51	-91.56	1.29	yes
39	1.35	238.52	40	35	157.83	0.1685	50	91.72	-91.56	-91.88	-91.667	0.34	yes
40	0.675	239.19	40	35	158.17	0.1678	50	91.92	-91.72	-91.89	-91.865	1.15	yes
41	1	240.19	40	35	157.67	0.1676	50	92.16	-91.92	-92.6	-92.039	0.26	yes
42	0.5	240.69	40	35	157.42	0.1679	50	92.17	-92.17	-92.32	-92.12	0.51	yes
43	0.5	241.19	40	35	157.17	0.1681	50	92.2	-92.17	-92.22	-92.15	0.56	yes
44	0.5	241.69	40	35	156.92	0.1682	50	92.22	-92.21	-92.27	-92.168	0.3	yes
45	0.25	241.94	40	35	157.05	0.1681	50.007	92.27	-92.22	-92.24	-92.266	4.83	yes
46	0.375	242.32	40	35	156.86	0.168	50	92.37	-92.27	-92.42	-92.315	0.32	yes
47	0.2	242.52	40	35	156.76	0.1681	50	92.39	-92.37	-92.38	-92.319	0.53	yes
48	0.2	242.72	40	35	156.66	0.1682	50.003	92.34	-92.39	-92.41	-92.335	0.78	yes
49	0.2	242.92	40	35	156.76	0.168	50	92.36	-92.33	-92.37	-92.304	0.57	yes
50	0.2	243.12	40	35	156.86	0.1679	50.008	92.38	-92.36	-92.39	-92.378	2.84	yes
51	0.4	243.52	40	35	157.06	0.1675	50.004	92.44	-92.38	-92.44	-92.439	1.01	yes
52	0.8	244.32	40	35	157.46	0.1668	50.007	92.69	-92.44	-92.54	-92.692	2.44	yes
53	1.2	245.52	40	35	158.06	0.1658	50.008	92.7	-92.69	-92.71	-92.703	0.58	yes
54	1.2	246.72	40	35	158.66	0.1648	50.009	92.87	-92.7	-92.87	-92.868	0.99	yes
55	1.8	248.52	40	35	159.56	0.1634	50.041	93.05	-92.87	-93.02	-93.053	1.22	yes
56	2.7	251.22	40	35	160.91	0.1609	50	93.52	-93.05	-93.88	-93.399	0.42	yes
57	1.35	252.57	40	35	161.58	0.1599	50	93.79	-93.52	-93.85	-93.731	1	yes
58	2.025	254.59	40	35	160.57	0.1608	50.088	93.85	-93.79	-94.01	-93.848	0.52	yes
59	2	256.59	40	35	159.57	0.1611	50.075	94.07	-93.85	-94.26	-94.067	0.53	yes
60	2	258.59	40	35	160.57	0.1592	50.018	94.27	-94.07	-94.45	-94.269	0.52	yes
61	2	260.59	40	35	161.57	0.1578	50.043	94.39	-94.27	-94.49	-94.386	0.52	yes

continued on next page

continued from previous page

k	Δ_k	P_h	P_l	t_p	t_a	u_a	p_{co2}	r_{co2}	$\tilde{\psi}_k^R(0)$	$\tilde{\psi}_k^R(s_k)$	$\psi(s_k)$	ρ_k	step?
		(kPa)	(kPa)	(sec)	(sec)	(m/s)	(%)	(%)					
62	2	262.59	40	35	162.57	0.156	50	94.64	-94.39	-94.86	-94.52	0.29	yes
63	1	263.59	40	35	163.07	0.1552	50	94.84	-94.64	-94.88	-94.719	0.83	yes
64	1	264.59	40	35	163.57	0.1548	50.045	94.89	-94.84	-94.91	-94.889	2.65	yes
65	1.5	266.09	40	35	164.32	0.1538	50.067	94.98	-94.89	-95.06	-94.976	0.52	yes
66	1.5	267.59	40	35	165.07	0.1521	50	95.26	-94.98	-95.3	-95.139	0.5	yes
67	1.5	269.09	40	35	165.82	0.1516	50.055	95.25	-95.26	-95.31	-95.25	2.46	yes
68	2.25	271.34	40	35	166.95	0.1501	50.073	95.33	-95.25	-95.55	-95.332	0.27	yes
69	1.2	272.54	40	35	167.55	0.1491	50.038	95.55	-95.33	-95.56	-95.55	0.96	yes
70	1.2	273.74	40	35	168.15	0.1483	50.037	95.63	-95.55	-95.72	-95.634	0.5	yes
71	1.2	274.94	40	35	168.75	0.1475	50.043	95.63	-95.63	-95.63	-95.634	0.52	yes
72	1.2	276.14	40	35	169.35	0.1467	50.035	95.82	-95.63	-95.8	-95.821	1.1	yes
73	1.8	277.94	40	35	170.25	0.1454	50.011	95.97	-95.82	-96.11	-95.969	0.5	yes
74	1.8	279.74	40	35	171.15	0.1447	50.098	96	-95.97	-96.04	-96.002	0.51	yes
75	1.8	281.54	40	35	170.25	0.1444	50	96.16	-96	-96.21	-96.11	0.5	yes
76	1.8	283.34	40	35	169.35	0.1448	50	96.24	-96.16	-96.3	-96.19	0.57	yes
77	1.8	285.14	40	35	168.45	0.1452	50	96.38	-96.24	-96.39	-96.325	0.93	yes
78	1.8	286.94	40	35	167.55	0.1455	50	96.51	-96.38	-96.52	-96.453	0.9	yes
79	1.8	288.74	40	35	166.65	0.146	50.002	96.51	-96.51	-96.61	-96.509	0.52	yes
80	1.8	290.54	40	35	165.75	0.1463	50	96.79	-96.51	-96.64	-96.578	0.52	yes
81	1.8	292.34	40	35	164.85	0.1467	50	96.7	-96.79	-96.92	-96.645	0.5	yes
82	1.8	294.14	40	35	163.95	0.1471	50	96.78	-96.7	-96.84	-96.694	0.35	yes
83	0.9	295.04	40	35	163.5	0.1474	50	96.81	-96.77	-96.83	-96.784	1.63	yes
84	1.35	296.39	40	35	162.82	0.1478	50.014	96.93	-96.81	-96.97	-96.93	0.95	yes
85	1.35	297.74	40	35	162.15	0.1481	50.023	96.95	-96.93	-96.97	-96.951	0.51	yes
86	1.35	299.09	40	35	161.47	0.1483	50.004	96.99	-96.95	-97.08	-96.984	0.25	yes
87	0.675	299.77	40	35	161.13	0.1485	50.007	97.01	-96.98	-97.04	-97.009	0.43	yes
88	0.338	300	40	35	160.97	0.1487	50.009	97.2	-97.01	-97.2	-97.197	0.99	yes
89	0.338	300	40	35	160.8	0.1487	50.001	97.03	-97.2	-97.98	-97.027	-0.2	no
90	0.084	300	40	35	160.92	0.1487	50.007	97.03	-97.2	-97.29	-97.028	-1.9	no
91	0.021	300	40	35	160.96	0.1486	50.006	97.16	-97.2	-97.28	-97.16	-0.4	no

B.3 Hybrid Filter Trust-region Algorithm: Tangential Subproblem

Table B.3: Iteration sequence for tangential subproblems of Algorithm II for Problem (7.31)

k	Δ_k	P_h	P_l	t_p	t_a	u_a	r_{co2}	$\tilde{f}_k^{\mathcal{R}}(0)$	$\tilde{f}_k^{\mathcal{R}}(s_k)$	$f(s_k)$	ρ_k	Eq. (7.38)	step?
		(kPa)	(kPa)	(sec)	(sec)	(m/s)	(%)						
		150	50	50	150	0.2	66.27			-66.268			
0r		158	48	54	154	0.22	67.55	NA	NA	-67.554		NA	yes
1r		166	46	58	158	0.24	68.79	NA	NA	-68.794		NA	yes
2r		178	43	64	164	0.27	70.61	NA	NA	-70.61		NA	yes
3r		193.33	40.33	64.3	168.68	0.289	73.12	NA	NA	-73.121		NA	yes
4	4	197.33	40	66.3	170.68	0.279	74.27	-73.12	-75.4	-74.265	0.5	holds	yes
5	4	193.33	40	68.3	168.68	0.269	74.69	-74.27	-77.36	-74.694	0.14	holds	yes
6	2	191.33	40	69.3	167.68	0.264	74.89	-74.69	-76.25	-74.889	0.13	holds	yes
7	1	191.79	40	68.8	167.18	0.2615	75.14	-74.89	-75.39	-75.143	0.51	holds	yes
8	1	192.28	40	68.3	166.68	0.259	75.48	-75.14	-75.71	-75.476	0.59	holds	yes
9	1	192.69	40	67.8	166.18	0.2565	75.75	-75.48	-75.99	-75.746	0.52	holds	yes
10	1	193.09	40	67.3	165.68	0.254	75.99	-75.75	-76.23	-75.989	0.5	holds	yes
11	1	193.91	40	66.8	165.18	0.2515	76.27	-75.99	-76.52	-76.271	0.54	holds	yes
12	1	193.94	40	66.3	164.68	0.249	76.59	-76.27	-76.9	-76.586	0.5	holds	yes
13	1	194.94	40	65.8	164.45	0.2465	76.86	-76.59	-77.07	-76.855	0.56	holds	yes
14	1	194.83	40	65.3	163.95	0.244	77.1	-76.86	-77.53	-77.103	0.37	holds	yes
15	0.5	195.2	40	65.04	163.69	0.2427	77.27	-77.1	-77.38	-77.267	0.59	holds	yes
16	0.5	195.35	40	64.79	163.44	0.2415	77.41	-77.27	-77.49	-77.408	0.64	holds	yes
17	0.5	195.66	40	64.54	163.19	0.2402	77.55	-77.41	-77.6	-77.552	0.74	holds	yes
18	0.5	196.14	40	64.29	162.94	0.239	77.75	-77.55	-77.73	-77.752	1.1	holds	yes
19	1	197.14	40	63.79	163.26	0.2365	78.03	-77.75	-78.02	-78.028	1.03	holds	yes
20	2	199.14	40	62.79	163.56	0.2315	78.63	-78.03	-78.96	-78.626	0.64	holds	yes
21	2	199.74	40	61.79	162.56	0.2264	79.35	-78.63	-79.63	-79.351	0.72	holds	yes
22	2	201.74	40	60.79	162.77	0.2215	79.98	-79.35	-80.35	-79.979	0.63	holds	yes
23	2	203.1	40	59.79	161.77	0.2165	80.71	-79.98	-80.99	-80.706	0.72	holds	yes
24	2	205.1	40	58.79	161.47	0.2115	81.42	-80.71	-81.42	-81.423	1.01	holds	yes
25	4	209.1	40	56.79	159.47	0.2054	82.67	-81.42	-82.66	-82.671	1.01	holds	yes

continued on next page

continued from previous page

k	Δ_k	P_h	P_l	t_p	t_a	u_a	r_{co2}	$\tilde{f}_k^R(0)$	$\tilde{f}_k^R(s_k)$	$f(s_k)$	ρ_k	Eq. (7.38)	step?
		(kPa)	(kPa)	(sec)	(sec)	(m/s)	(%)						
26	8	217.1	40	52.79	155.47	0.1971	84.8	-82.67	-84.58	-84.802	1.12	holds	yes
27	16	233.1	40	60.79	157.85	0.1571	93.26	-84.8	-96.69	-93.26	0.71	holds	yes
28	16	249.1	40	68.79	165.85	0.1639	89.66	-96.23	-89.01	-89.662	0.5	fails	yes
29	8	257.1	40	64.79	169.85	0.1516	92.85	-89.66	-92.92	-92.848	0.98	holds	yes
30	8	265.1	40	68.79	173.85	0.1484	92.89	-92.85	-92.62	-92.887	-0.2	fails	yes
31	8	273.1	40	72.79	177.85	0.1425	93.66	-92.89	-93.24	-93.662	2.22	holds	yes
32	16	289.1	40	80.79	185.85	0.1328	94.52	-93.66	-93.93	-94.519	3.16	holds	yes
33	32	299.44	40.06	86.08	200.34	0.1208	94.98	-94.52	-97.03	-94.978	0.18	holds	yes
34	16	300	40	94.08	208.34	0.1201	93.83	-94.98	-94.31	-93.832	1.72	fails	yes
35	16	300	40	86.08	200.34	0.1235	94.47	-93.85	-95.06	-94.466	0.52	holds	yes
36	16	300	40	78.08	192.34	0.1273	94.87	-94.44	-96.21	-94.869	0.23	holds	yes
37	8	300	40	74.08	188.34	0.1295	95.28	-94.86	-95.62	-95.28	0.54	holds	yes
38	8	300	40	70.08	184.34	0.1326	95.18	-95.28	-95.2	-95.182	1.18	fails	yes
39	8	300	40	66.08	180.34	0.1346	95.59	-95.19	-95.46	-95.594	1.53	holds	yes
40	16	300	40	58.08	188.34	0.129	96.05	-95.59	-95.92	-96.047	1.38	holds	yes
41	32	300	40	42.08	172.34	0.139	96.95	-96.07	-96.98	-96.946	0.99	holds	yes
42	32	300	40	35	188.34	0.1277	97.24	-96.94	-99.06	-97.237	0.14	holds	yes
43	16	300	40	35	180.34	0.1335	97.18	-97.24	-99.61	-97.176	-0	holds	no
44	4	300	40	35	186.34	0.1289	97.1	-97.24	-97.86	-97.099	-0.2	holds	no
45	1	300	40	35	187.84	0.1278	97.25	-97.24	-97.42	-97.253	0.08	holds	yes
46	0.5	300	40	35	188.09	0.1276	97.24	-97.25	-97.28	-97.244	-0.3	holds	no
47	0.125	300	40	35	187.9	0.1277	97.25	-97.25	-97.27	-97.254	0.05	holds	yes
48	0.063	300	40	35	187.93	0.1276	97.25	-97.25	-97.26	-97.248	-2.6	holds	no
49	0.016	300	40	35	187.91	0.1277	97.26	-97.25	-97.25	-97.256	6.28	holds	yes
50	0.031	300	40	35	187.93	0.1277	97.25	-97.26	-97.26	-97.248	-6	holds	no

B.4 Hybrid Filter Trust-region Algorithm: Normal Subproblem

Table B.4: Iteration sequence for normal subproblems of Algorithm II for Problem (7.31)

k	P_h (kPa)	P_l (kPa)	t_p (sec)	t_a (sec)	u_a (m/s)	p_{co2} (%)	$\widehat{\theta}_k^{\mathcal{R}}(s_k)$	$\widehat{f}_k^{\mathcal{R}}(s_k)$
	150	50	50	150	0.2	37.764	12.236	
0r	158	48	54	154	0.22	40.291	9.709	-67.554
1r	166	46	58	158	0.24	42.838	7.162	-68.794
2r	178	43	64	164	0.27	46.662	3.338	-70.6097
3r	193.33	40.33	64.3	168.68	0.289	50.58	0	-73.121
4	193.33	40.33	64.3	168.68	0.289	51.134	0	-74.265
5	197.33	40	66.3	170.68	0.279	50.435	0	-74.694
6	193.33	40	68.3	168.68	0.269	50.078	0	-74.889
7	191.33	40	69.3	167.68	0.264	50.079	0	-75.143
8	191.79	40	68.8	167.18	0.2615	50.087	0	-75.476
9	192.28	40	68.3	166.68	0.259	50.082	0	-75.746
10	192.69	40	67.8	166.18	0.2565	50.072	0	-75.989
11	193.09	40	67.3	165.68	0.254	50.113	0	-76.271
12	193.91	40	66.8	165.18	0.2515	50.06	0	-76.5858
13	193.94	40	66.3	164.68	0.249	50.12	0	-76.855
14	194.94	40	65.8	164.45	0.2465	50.047	0	-77.1026
15	194.83	40	65.3	163.95	0.244	50.056	0	-77.2668
16	195.2	40	65.04	163.69	0.2427	50.039	0	-77.408
17	195.35	40	64.79	163.44	0.2415	50.035	0	-77.552
18	195.66	40	64.54	163.19	0.2402	50.059	0	-77.752
19	196.14	40	64.29	162.94	0.239	50.128	0	-78.028
20	197.14	40	63.79	163.26	0.2365	50.245	0	-78.626
21	199.14	40	62.79	163.56	0.2315	50.157	0	-79.351
22	199.74	40	61.79	162.56	0.2264	50.245	0	-79.979
23	201.74	40	60.79	162.77	0.2215	50.205	0	-80.706
24	203.1	40	59.79	161.77	0.2165	50.241	0	-81.423
25	205.1	40	58.79	161.47	0.2115	50.335	0	-82.671
<i>continued on next page</i>								

continued from previous page

k	P_h	P_l	t_p	t_a	u_a	p_{co2}	$\tilde{\theta}_k^R(s_k)$	$\tilde{f}_k^R(s_k)$
	(kPa)	(kPa)	(sec)	(sec)	(m/s)	(%)		
26	209.1	40	56.79	159.47	0.2054	50.537	0	-84.802
27	217.1	40	52.79	155.47	0.1971	47.469	2.531	-93.2604
28	239.2	41.23	60.77	159.88	0.1817	50.654	0	-89.662
29	249.1	40	68.79	165.85	0.1639	49.645	0.355	-92.848
30	258.36	40.71	64.79	170.28	0.1616	50.058	0	-92.8865
31	265.1	40	68.79	173.85	0.1484	49.883	0.117	-93.662
32	275.21	41.5	72.78	178.5	0.161	49.719	0.281	-94.519
33	292.47	41.13	80.86	186.17	0.1546	49.492	0.508	-94.978
34	296.93	40.53	86.11	200.93	0.1382	50.389	0	-93.8316
35	300	40	94.08	208.34	0.1201	50.192	0	-94.466
36	300	40	86.08	200.34	0.1235	50.01	0	-94.8686
37	300	40	78.08	192.34	0.1273	49.955	0.045	-95.2801
38	297.37	40.74	74.06	188.85	0.1395	50.113	0	-95.182
39	300	40	70.08	184.34	0.1326	49.966	0.034	-95.594
40	294.73	41.53	65.99	181.2	0.1542	50.07	0	-96.0471
41	300	40	58.08	188.34	0.129	49.906	0.094	-96.946
42	289.47	43.4	44.4	173.74	0.176	50.07	0	-97.2373
43	300	40	35	188.34	0.1277	50.134	0	-97.176
44	300	40	35	188.34	0.1277	50.048	0	-97.099
45	300	40	35	188.34	0.1277	50.019	0	-97.2528
46	300	40	35	187.84	0.1278	50.001	0	-97.2443
47	300	40	35	187.84	0.1278	50	2.2E-05	-97.2535
48	299.99	40	35.01	187.9	0.1277	49.999	7.2E-04	-97.24844
49	299.99	40	35	187.9	0.1277	50.001	0	-97.25646
50	300	40	35	187.91	0.1277	49.997	2.8E-03	-97.24844